# Trust in 5G Open RANs through Machine Learning: RF Fingerprinting on the POWDER PAWR Platform

Guillem Reus-Muns, Dheryta Jaisinghani, Kunal Sankhe, Kaushik R. Chowdhury
Institute for the Wireless Internet of Things, Northeastern University, Boston, USA
greusmuns@coe.neu.edu, dheryta@ieee.org, sankhe.ku@husky.neu.edu, krc@ece.neu.edu

***Abstract*—5G and open radio access networks (Open RANs) will result in vendor-neutral hardware deployment that will require additional diligence towards managing security risks. This new paradigm will allow the same network infrastructure to support virtual network slices for transmit different waveforms, such as 5G New Radio, LTE, WiFi, at different times. In this multi-vendor, multi-protocol/waveform setting, we propose an additional physical layer authentication method that detects a specific emitter through a technique called as RF fingerprinting. Our deep learning approach uses convolutional neural networks augmented with triplet loss, where examples of similar/dissimilar signal samples are shown to the classifier over the training duration. We demonstrate the feasibility of RF fingerprinting base stations over the large-scale over-the-air experimental POWDER platform in Salt Lake City, Utah, USA. Using real world datasets, we show how our approach overcomes the challenges posed by changing channel conditions and protocol choices with 99.86% detection accuracy for different training and testing days.**

## I. Introduction

The advent of 5G and mobile edge computing (MEC) paradigm have enabled network slicing where a number of different protocols and waveforms can be transmitted by the same base station (BS). An interesting case arises when the BS hardware is also shared among different vendors, as seen in Open RANs. With the inevitable opening up of hardware access and interfaces, it is necessary to carefully consider the need for additional forms of authentication that can be performed at the client without introducing any overhead of signaling and spectrum use. We propose trust metric based on a method called Radio Frequency (RF) fingerprinting, which learns discriminative features introduced by the transmitter's processing chain on the signals that pass through it. The overarching idea is for the client to continuously monitor the signals from the BS and match its known fingerprint with the broadcast ID.

• **Problem.** Protocols like 5G New Radio, LTE, and WiFi have different standards-defined authentication mechanisms. Yet, when network slicing is implemented in an Open RAN architecture, not only can the waveform change over time, but the BS functionality can also virtually migrate from one hardware/software defined radio (SDR) to another. This raises many new issues in the context of *trust* towards a BS that a node is associated with. This can arise from simple misconfigurations or intentional variations, for example, as shown in Fig. 1, where a new vendor accesses network infrastructure at BS 4 and 5, with false advertising of a different BS ID. Thus, with SDR-enabled open-source implementation of standards, there is need for vigilance to thwart attacks like forced cell-outage, false signaling, wrong identification, incorrect bidding, battery drain, advertising fake BSs among others [1, 2].

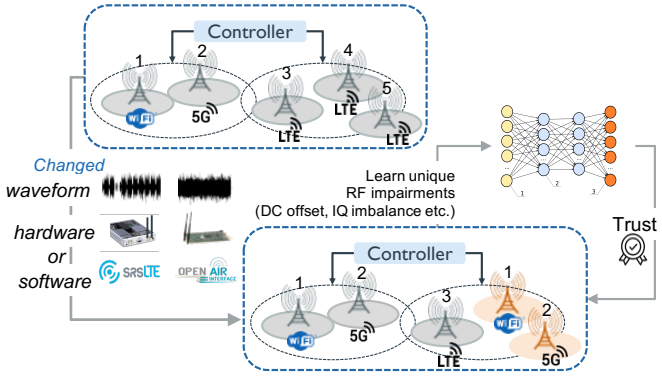



**Fig. 1:** Using RF fingerprinting to detect cases where certain BS (in orange) transmit incorrect IDs, i.e., spoofing BS 1 and 2. Such violations must be detected irrespective of the waveform being transmitted.

Although there exist several security schemes at upper-layers of the protocol stack to authenticate BSs, they involve operations at the central cloud. This may not be suitable for ultra-low latency networks, besides imposing additional resource requirements [3]. Instead, we propose a trust metric based on the probability of correctly classifying the BS using IQ samples of signals available at the physical layer, which can be directly undertaken at the associated clients. The trust metric builds upon traditional RF fingerprinting, where devices can be uniquely identified solely with raw IQ samples. A number of gain/phase/frequency offsets and non-linear distortions commonly known as fingerprints are induced by process variations during manufacturing. Given the inherent randomness of wireless channels, a single sequence of IQ samples may not be enough to establish the credibility or *trust* in identifying a BS. In contrast, in accordance with the law of large numbers, we empirically determine the low-trust ranges to request more data samples from the BS for the purpose of a trustworthy classification. In the process, we acquire enough data for a reliable classification while considering the randomness of wireless channels.

• **Challenges in RF Fingerprinting for Open RAN.** Our prior work explored demonstrating RF fingerprinting on datasets collected from WiFi devices when training and testing sets are carved out from a large shuffled collection of IQ samples [4, 5]. Our investigations uncovered the significant impact of the wireless channel in the classification accuracy, which results in

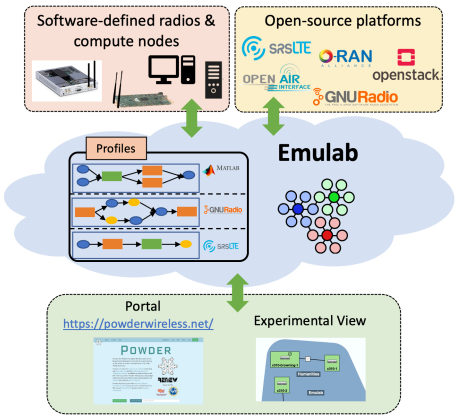



**Fig. 2:** POWDER Experimental Flow

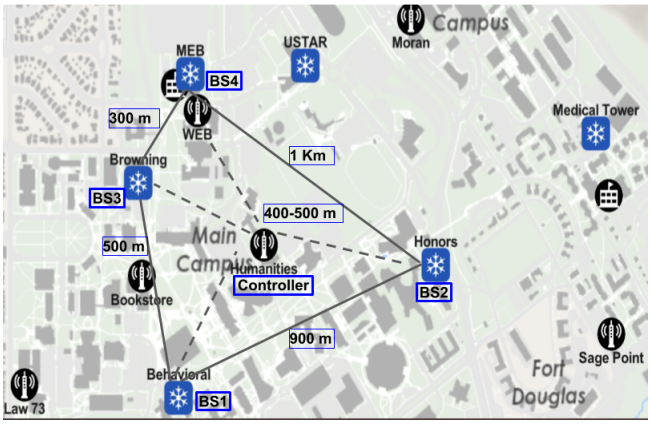



**Fig. 3:** Topological map of POWDER network. A central controller records raw IQ samples from each of the 4 BSs.

a massive drop (close to 50% in some cases) when the dataset for testing is collected on a different day from that of training, majorly due to unpredictable channel variations [6–8]. Thus, the issue of trust cannot be resolved unless we show that the classification accuracy is not susceptible to changing channel conditions spanning days. Furthermore, the RF fingerprinting method should not be impacted by the protocol chosen by the vendor, i.e., the classification outcome should be unchanged if a cellular waveform or WiFi waveform are used at any given BS. Finally, no RF Fingerprinting performance has ever been demonstrated on a large-scale experimental testbed that permits repeatable experiments. For this reason, we have chosen to utilize a dataset collected from the POWDER testbed in Salt Lake City [9], which is a joint NSF-industry funded and publicly accessible experimental platform spanning 6 square kilometers, to demonstrate the feasibility of deploying our trust-building concept at-scale.

• **Approach and Contributions.** To combat the adversarial impact of the wireless channel, we leverage neural networks with triplet-loss functions. This method has seen success in learning semantic similarity [10, 11] but, so far has not been applied in the RF domain. We collect datasets from the same set of BSs that emit standards-compliant WiFi, LTE, and 5G New Radio (NR) waveforms to show the protocol-agnostic nature of our approach. Our implementation on POWDER is also an early demonstrator of how the PAWR platform can be utilized by the research community for advanced 5G research. To support independent investigations beyond this work, we make the dataset publicly accessible [12].

Our results demonstrate an accuracy of 99.86% *irrespective of the training/testing time gap*, for the over-the-air datasets collected on POWDER, which not only significantly advances the state-of-the-art in RF Fingerprinting, but also demonstrates the potential benefit in building trust for future Open RAN networks.

## II. Experimental Setup on POWDER

### A. POWDER Platform

The POWDER platform supports wireless and mobility related experiments at a community-wide scale and is composed of SDRs and open-source software stacks. Our experimentation on POWDER utilizes the available hardware resources of BSs, user endpoints, and MEC.

A BS is composed of a USRP X310 SDR with MIMO capabilities and multi-band antennas that can operate from 1.6 GHz to 6 GHz. Although there are a total of 8 BSs deployed on the rooftop of campus buildings and connected to a central aggregation point via 10 Gigabit Ethernet links, we use 4 of them at a time in this work. POWDER employs USRP B210 SDRs for the fixed end points and they are capable of operating from 698 MHz to 6 GHz. Compute hosts for the SDRs are Intel Core i7-8559U based computers with 32 GB RAM and 250 GB storage. In addition, there are 19 edge-compute nodes with the number of cores ranging from 12-16, RAM 192-768 GB, storage up to TBs, all connected with high-speed backhaul links. Fig. 2 abstracts how the protocol/programming blocks are arranged in POWDER. As a user, we instantiate an experiment with a 'profile' that encapsulates the hardware resources and software modules that execute on the reserved SDR/compute resources. POWDER allows programming the profiles with commonly available tools like Python and MATLAB.

### B. Experiment Setup

We study the performance of our RF fingerprinting approach using IQ samples collected from the experimental setup shown in Fig. 3. A fixed end-point USRP B210 is the receiver that wishes to authenticate and associate with an available BS. Thus, it collects the IQ samples and runs the inference step for verifying the BS ID. All transmitter BSs are bit-similar USRP X310 radios (random payload but same address fields). The BS emits one of the following waveforms – WiFi (IEEE 802.11ac), 3GPP 4G LTE and 5G NR standards-compliant frames that are generated through MATLAB WLAN, LTE or 5G toolbox, respectively. We have a central controller equipped with a B210 SDR that captures transmitted frames from 4 surrounding BSs (Fig. 3). The incoming signals are sampled at 5 MS/s at center frequency of 2.685 GHz for WiFi and 7.68 MS/s sampling rate at center frequency of 2.685 GHz for LTE and 5G NR.

The distance between BSs and between the central controller varies from 300 meters to 1 kilometer. As this is a real-world deployment with human activity and terrain variations, there are several line-of-sight (LOS) and Non-LOS components

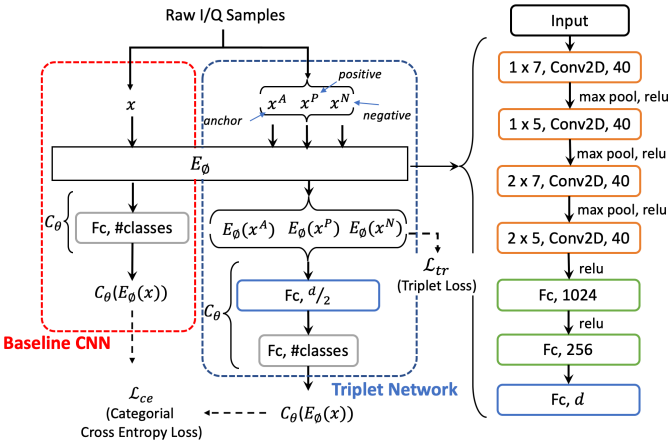



**Fig. 4:** Classification with baseline CNN vs triplet network. In the triplet network, raw IQ samples are fed into the model. First, an embedding $E_\phi(x)$ of these triplets is obtained, which are passed through the classification network with weights $\theta$.

for the transmission links that results in considerable channel variation over days. On any given day, we collect 5 sets of 3 million IQ samples per BS at the controller, each set separated in time by ≈10 seconds. We then repeat this entire data collection process on a different day. Each stream of IQ samples is split into sequences of 512 samples, which we refer as a "slice". The dataset collected on the first day is subsequently divided into 80% for training, 10% for validation and 10% for test. The data collected on a day different than training is fully used for testing purposes.

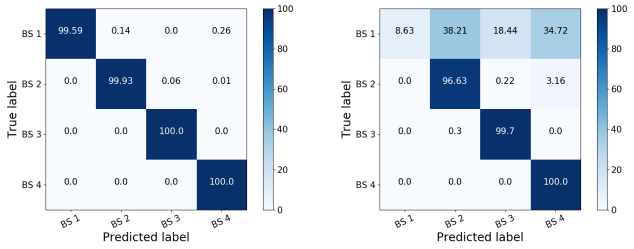
## III. Learning to Identify Base Station Fingerprints

In this section, we first analyze the detection accuracy under unseen channel conditions with the baseline CNN shown in Fig. 4. Using the outcomes from this study, we formulate a triplet loss approach that results in an improvement of 21.94%. The key idea here is that triplet network reinforces the embedding separation among classes prior to the final classification step.

### A. Results using baseline CNN

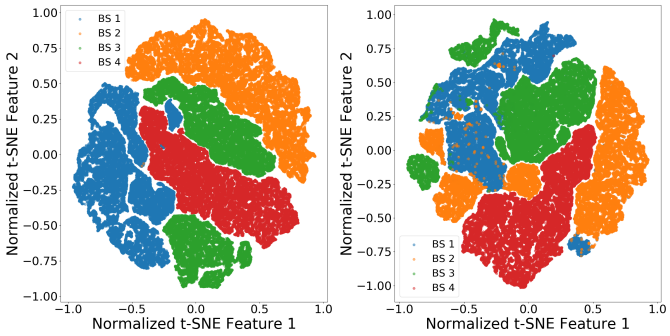
Our preliminary evaluation with only WiFi waveform aims to demonstrate the adverse effect of the wireless channel on the accuracy of RF fingerprinting. This occurs when there is a time gap between the training and testing phase, suggesting that the learned features are not purely that of the transmitter, but also the channel plays a discriminative role. These studies motivate our enhancements proposed later in Sec. III-B.

• ***Classifier architecture*:** Our baseline CNN architecture consists of eight layers, with four convolutional layers and four fully connected (or dense) layers. The input to our CNN is a windowed sequence of raw IQ samples with length 512, referred as a slice. Each complex value is represented as two-dimensional real values (i.e., I and Q are two real value streams), which results in the dimension of our input data growing to $2 \times 512$. This is then fed to the main CNN building block, which we refer to as $E_\phi$ as shown in Fig. 4. The first two convolutional layers have 40 filters with size $1 \times 7$ and




**(a)** **(b)**

**Fig. 5:** Accuracy of BS detection with baseline CNN when a) trained and tested on the same day gives overall accuracy of 99.98%; b) trained and tested on different days, the overall accuracy drops to 76.24%. BS1 is confused with the other three BSs when tested on another day, despite near-perfect accuracy when tested on the same day.




**(a)** **(b)**

**Fig. 6:** 2D feature visualization with t-SNE plots when baseline CNN is a) trained and tested on the same day; b) trained and tested on different days.

$1 \times 5$, respectively, which learn a 7-sample variation in time over the I or Q dimensions separately. On the other hand, the next two convolutional layers have 40 filters of size $2 \times 7$ and $2 \times 5$ that learn variations over both I and Q dimensions jointly. Each convolutional layer (except the last one) is followed by a Max Pooling layer that reduces the dimensionality of the output feature maps of the preceding convolutional layer, while retaining the most important information. The last convolutional layer is followed by a set of 3 Fully Connected (Fc) layers, composed of 1024, 256 and $d$ neurons, respectively, and a Softmax classifier layer. The use of parameter $d$ is justified later in Sec. III-B. We use PyTorch for our implementation and the choice of hyperparameters, such as filter size, number of filters in the convolutional layers and the depth of the CNN are chosen carefully through cross-validation. We chose Adam optimizer with a learning rate of $1e^{-4}$ and a weight decay of $1e^{-4}$.

• ***Classification Accuracy*:** Fig. 5a shows the near-perfect classification accuracy of 4 BSs, when the baseline CNN is trained and tested with data collected on the same day. In an at-scale deployment, however, we are likely to observe environmental changes on a daily basis. These changes have a considerable impact on received IQ samples, at times distorting the samples such that the classifier no longer correctly identifies the BS. As we show next, classification performance degrades severely when classifiers are trained on raw IQ samples on a given day and then tested on IQ samples obtained on a

different day. Fig. 5b shows the classification accuracy for the same setup on a different day, where we observe the overall classification accuracy drops to 76.24%. From the confusion matrix, we infer that BS1 is mainly confused with the remaining BSs.

• ***t-Distributed Stochastic Neighbor Embedding (t-SNE)*:** t-SNE is widely used to reduce high-dimensional data into a lower dimensional space. We use it to visualize the internal representation of input samples that neural networks have learned during the training phase, giving an idea of the feature space that hidden layers (in particular, the layer (Fc, $d$), which we refer to as an *embedding* layer) for mapping input samples.

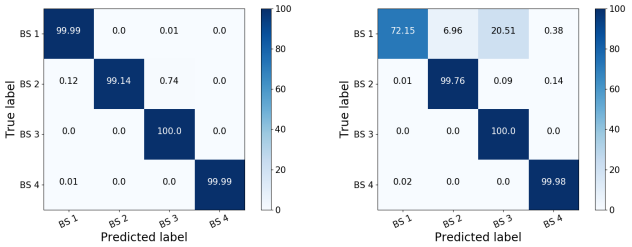
We leverage t-SNE to reveal the reasons behind the poor performance of the baseline CNN in detecting BSs on different days. The bi-dimensional similarity map helps to visualize the embedding features that are obtained from the last fully connected layer (embedding layer). We randomly select 2000 samples for each BS from the test set, pass them through the trained CNN while keeping track of each sample output at the embedding layer. After collecting these intermediate representations, we use t-SNE to visualize their similarities and project them onto a 2-D plane. First, we show the outputs for the above mentioned experiments for a scenario, when baseline CNN is trained and tested on the same day as shown in Fig. 6a. We infer that IQ samples collected on the same days for all 4 BSs are clearly separable and encoded successfully by the CNN, resulting in near-perfect classification accuracy. Similarly, we plot t-SNE representation for the second scenario when baseline CNN is trained and tested on different days, as shown in Fig. 6b. This plot confirms that IQ samples collected on different days get mapped into more similar, non-linear intermediate representations by the neural network, leading to incorrect classification of the transmitting BS. This effect explains the poor results seen in the confusion matrix shown in Fig. 5b.

### B. Triplet Network

Given a neural network with parameters $\phi$ ($E_\phi$), output size $d$, and an input $x$, we consider $E_\phi \in \mathbb{R}^d$ an embedding of input $x$ into a $d$-dimensional Euclidean space. The triplet loss [10] is designed to enforce class separation into the embedding space. Typically, the triplet loss is trained on series of triplets $x^A, x^P, x^N$, where $x^A$ is the *anchor*, $x^P$ is the *positive* and $x^N$ the *negative*. Then, the loss function is designed to minimize the distance between $x^A$ and $x^P$, which belong to the same class, while maximizing the distance from $x^A$ to $x^N$, which belong to different classes. The triplet loss is formulated as:

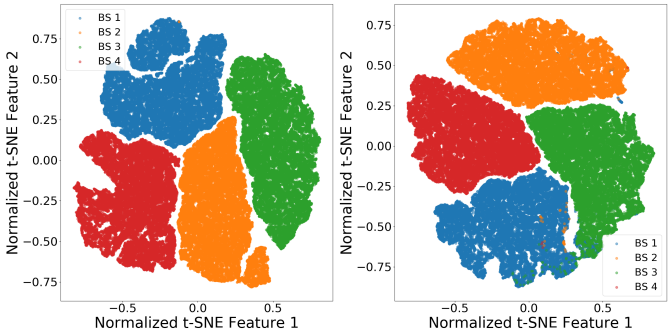
$$\mathcal{L}_{tr} = \max(\sum_{i}^{N} ||E_\phi(x_i^A) - E_\phi(x_i^P)||^2 - ||E_\phi(x_i^A) - E_\phi(x_i^N)||^2 + \alpha, 0) \quad (1)$$

where $\alpha$ represents a margin enforced between the positive and negative pairs. While this work generates the triplets randomly during training time, may other methods are proposed [10], which we will explore in future work.




**(a)** **(b)**

**Fig. 7:** Accuracy of BS detection with triplet network when a) trained and tested on the same day is 99.98%; b) trained and tested on different days is 92.97%. Notice the improvement in detection accuracy for BS, BS1, as shown in the confusion matrices.




**(a)** **(b)**

**Fig. 8:** 2D feature visualization with t-SNE plots when the triplet network is a) trained and tested on the same day; b) trained and tested on different days. We observe clear feature demarcation for all the four classes in the 2D feature visualization with t-SNE plots.

We aim to leverage the learned embeddings by minimizing (1) for the classification task. Intuitively, given a *large* set of IQ samples from different devices that have been effected by different channel conditions, we would expect the learned representation to isolate such channel distortions and group each class based on residual features that will facilitate the classification. For that purpose, we connect the output of the embedding network ($E_\phi$) with a classification network with parameters $\theta$ ($C_\theta$). Then, we define a combined loss function in order to ensure class separation while achieving successful classification.

$$\mathcal{L}_T = \mathcal{L}_{tr} + \mathcal{L}_{ce} \quad (2)$$

where $\mathcal{L}_{ce}$ is the categorical cross-entropy. In Fig. 4 we show a summary of the overall approach.

## IV. Accuracy Improvement with Triplet Network

In this section, we present the performance of triplet network showing that identification accuracy does not degrade despite varying wireless channel conditions.

• ***Classification Accuracy*:** Fig. 7a shows the confusion matrix for the triplet network when it is trained and tested with WiFi transmissions emitted by BSs on the same day. We observe an overall accuracy of 99.98%. In Fig. 7b, we plot the confusion matrix for of the classification accuracy obtained from the same trained model but tested on a different day. This time, although the overall accuracy reduces marginally to

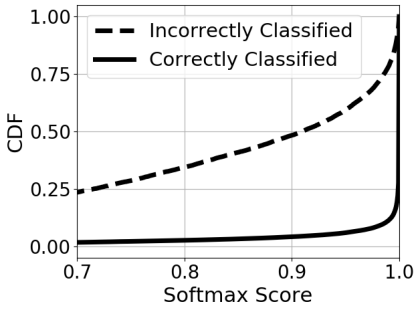



**Fig. 9:** Softmax score CDF for correct and incorrect classification.

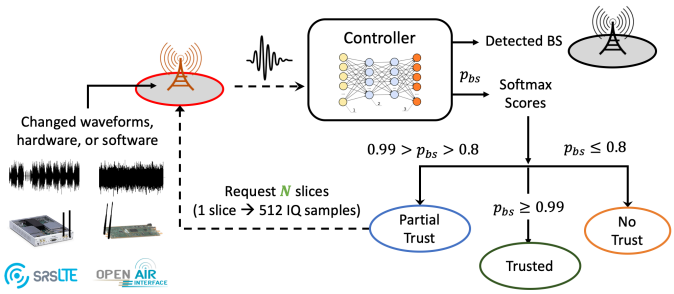



**Fig. 10:** Trust Algorithm. The controller leverages the triplet network in conjunction with majority votes for incoming slices to detect trust level of a BS. A BS with partial trust is requested for more slices to further establish trust.

92.97%, unlike baseline CNN, there is no significant degradation in the classification accuracy because results for BS1 are not impacted (row 1 of the confusion matrices).

• ***t-Distributed Stochastic Neighbor Embedding (t-SNE)*:** Similar to baseline CNN, we analyze the feature maps learned by the triplet network using t-SNE visualization plots. Fig. 8a plots the t-SNE representation for the triplet network when it is trained and tested on the same day; whereas Fig. 8b plots when the triplet network is trained and tested on different days. In both plots, we observe clear feature demarcation for all the four classes in the 2D feature visualization with t-SNE. This clearly indicates triplet network is able to learn more discriminating features than baseline CNN, resulting in better classification accuracy.

While the approach introduced in this section represents a meaningful improvement on the RF fingerprinting classification accuracy, a measure of trust is needed to quantify the certainty of each decision, which we present in the following section.

## V. Quantifying 'Trust' in a Base Station

We next formulate an algorithm that returns a quantitative measure of trust in a BS while it advertises its ID. For this purpose, we propose a three step approach based on the softmax score values. In classification tasks, neural networks typically use a softmax activation function in the final layer, which can be expressed as:

$$\pi_i(\boldsymbol{z}) = \frac{\exp z_i}{\sum_{i=1}^{K} \exp z_i} \tag{3}$$

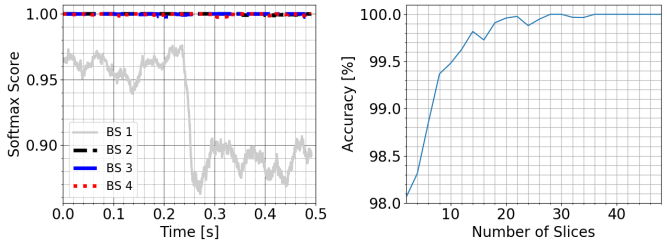
where $\boldsymbol{z}$ is an input of dimension $K$. The softmax function normalizes a series of inputs into a probability distribution $\boldsymbol{\pi}$, where each value $\pi_i$ is proportional to the exponential of the input $z_i$. Thus, given a classification task with $K$ classes, the softmax values reported at the last layer of the neural network represent the probability of the input to belong to any of the $K$ classes, where the class with the highest probability $\pi_{max}$ will be chosen.

In Fig. 9, we analyze the distribution of the $\pi$ values for each class ($\pi_{max}$) both for the correctly and incorrectly classified inputs. As expected, we see that the majority of the correctly classified cases obtained a $\pi_{max}$ value very close to 1. On the other hand, the distribution of scores in the incorrectly classified cases are scattered with 265% higher variance and 0.118 lower mean. Based on this analysis, we consider three $\pi_{max}$ ranges – (a) $\leq$ 80%, (b) 80% and 99%, and (c) $\geq$ 99%. As shown in Fig. 10, each range here corresponds to a trust category – (a) No Trust, (b) Partial Trust, (c) Trusted. In case (a), since classification confidence is very low, the BS is deemed to be not trustworthy the receiver should re-establish trust with fresh beacon exchanges and standards-defined authentication steps. In case (c), the classification confidence is close to 100% and thus, the end result from the classification is a trusted BS.

However, case (b), needs careful consideration – we present an approach to consider different slices of frames coming in from a transmitter, passing through different channel conditions. Then, a classification decision is obtained for each of the slices independently, and later combined into a majority voting approach. The class that has most slices voting for it is chosen as the final outcome. The number of slices will depend on factors pertaining to randomness of wireless channels.

We demonstrate in Fig. 11a that the Softmax score of one of the BSs drops over time due to various factors involved with Open RANs. As identified above, trust of a BS can be established if more frames from the BS are received in time for classification. As shown in Fig. 11b, an accuracy of near-100% is achieved as the number of slices increases while considering the majority voting approach.

• ***Multi-Waveform Identification while Establishing Trust*:** Fig. 12 shows the results in identifying BSs irrespective of the transmit waveforms (WiFi, LTE or 5G NR) emitted by those BSs. Ideally, we would like the trust establishment to perform equally well irrespective of the waveform choice. While our approach does maintain a minimum classification accuracy of 92.97% for a single waveform, it establishes trust for the BS identified in Fig. 11a with slices received from multiple consecutive transmissions. Considering such a case, the accuracy of BS1 increases from 74% to 99% for slices taken from 10 consecutive transmissions. In the multi-waveform scenario where any jumbled mix of IQ samples from WiFi/LTE/5G waveforms are fed to the classifier (with a separate mixed training dataset for these three waveforms), we see an increase in the accuracy, since a greater variety of channel conditions are involved in the training process. However, both single and multi-waveform scenarios converge to similar accuracy values using the trust algorithm from




**(a)** **(b)**

**Fig. 11:** (a) Softmax scores over time demonstrate a drop in trust due to channel variations for a particular BS, BS1 in this case, while the other 3 remain at highest score of 1 and (b) accuracy for different number of slices considered shows significant improvement in classification accuracy as the number of slices increase.

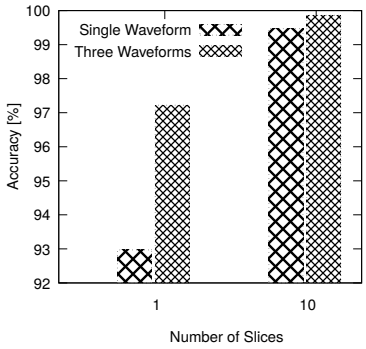



**Fig. 12:** Classification accuracy of single and multiple waveforms for different number of slices considered to establish trust. Accuracy increases with the number of slices considered, irrespective of training and testing day.

Fig. 10. We note that these results are from different datasets collected on separate training and testing days, which confirms the robustness of our proposed approach for unseen channels.

## VI. Related Work

RF fingerprinting has so far been limited to theory, simulations, and small-scale lab experiments. We briefly describe recent experimental results below. [13] uses RF fingerprinting on GSM waveforms with a probability-based thresholding scheme to differentiate between a rogue and a legitimate BS. The hardware used is prior generation USRP N210 SDRs with Flex900 daughter boards, GSM900 frequency range (935MHz to 960MHz for downlink). Unlike USRP X310s used in our study, the N210s have large intrinsic impairments (IQ imbalance, DC offsets) that make them easier to differentiate. Moreover, this work is majorly based on feature extraction from PHY transmissions (signal processing features) and thus, it requires an extensive domain knowledge. The work also uses a dataset collected in a controlled lab environment. For the outdoor evaluation, the authors deploy a single rogue BS and the distance between the receiver and the rogue BS varies from 20 m to 210 m. Another similar work that employs RF fingerprinting for physical layer authentication is presented in [14]. Again, the authors perform an SDR based evaluation in a laboratory environment, where the two radios were placed 20 feet apart within LOS. The authors in [3] propose a fingerprinting based authentication scheme for handover scenario in 5G heterogenous networks, which is evaluated with Matlab simulations. In contrast, our work establishes a trust metric for a BS irrespective of the protocol of transmission and the day of testing. Moreover, we perform real-world data collection on the POWDER PAWR platform, and release the dataset for public use.

## VII. Conclusion

We presented the first-of-its-kind RF fingerprinting evaluation on the POWDER PAWR platform to demonstrate how to build trust in future 5G networks. Our approach of incorporating the triplet loss with the deep CNN shows an improvement in detection accuracy to 92.97% for a single slice and achieve a total accuracy of 99.86% for 10 slices using majority voting. Importantly, our approach for establishing trust between BSs shows the RF fingerprinting works well for unseen channels/days, and this will result in mitigating deployment barriers for trusted Open RANs.

## Acknowledgement

This work is supported by the US National Science Foundation Award CNS-1923789.

## References

[1] H. Yang, S. Bae, M. Son, H. Kim, S. M. Kim, and Y. Kim, "Hiding in Plain Signal: Physical Signal Overshadowing Attack on LTE," *USENIX Security Symposium*, 2019.

[2] A. Shaik, R. Borgaonkar, S. Park, and J.-P. Seifert, "New Vulnerabilities in 4G and 5G Cellular Access Network Protocols: Exposing Device Capabilities," *ACM WiSec*, 2019.

[3] T. Ma, F. Hu, and M. Ma, "Securing 5g hetnets using mutual physical layer authentication," in *ICIT*, 2019.

[4] S. Riyaz, K. Sankhe, S. Ioannidis, and K. Chowdhury, "Deep learning convolutional neural networks for radio identification," *IEEE Communications Magazine*, 2018.

[5] K. Sankhe, M. Belgiovine, F. Zhou, S. Riyaz, S. Ioannidis, and K. Chowdhury, "Oracle: Optimized radio classification through convolutional neural networks," in *IEEE INFOCOM*, 2019.

[6] F. Restuccia, S. D'Oro, A. Al-Shawabka, M. Belgiovine, L. Angioloni, S. Ioannidis, K. Chowdhury, and T. Melodia, "Deepradioid: Real-time channel-resilient optimization of deep learning-based radio fingerprinting algorithms," in *ACM MobiHoc*, 2019.

[7] A. Al-Shawabka, F. Restuccia, S. D'Oro, T. Jian, B. C. Rendon, N. Soltani, J. Dy, K. Chowdhury, S. Ioannidis, and T. Melodia, "Exposing the fingerprint: Dissecting the impact of the wireless channel on radio fingerprinting," in *IEEE INFOCOM*, 2020.

[8] E. Goldoni, L. Prando, A. Vizziello, P. Savazzi, and P. Gamba, "Experimental data set analysis of rssi-based indoor and outdoor localization in lora networks," *Internet Technology Letters*, 2019.

[9] J. Breen, A. Buffmire, J. Duerig, K. Dutt, E. Eide, M. Hibler, D. Johnson, S. Kumar Kasera, E. Lewis, D. Maas, A. Orange, N. Patwari, D. Reading, R. Ricci, D. Schurig, L. B. Stoller, K. Van der Merwe, K. Webb, and G. Wong, "Powder: Platform for open wireless data-driven experimental research," in *Proceedings of the 14th International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH)*, 2020.

[10] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *IEEE CVPR*, 2015.

[11] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML deep learning workshop*, 2015.

[12] Genesys, "Genesys Lab ML datasets," [accessed 01-Sep-2020]. [Online]. Available: http://genesys-lab.org/mldatasets

[13] Z. Zhuang, X. Ji, T. Zhang, J. Zhang, W. Xu, Z. Li, and Y. Liu, "FBSleuth: Fake Base Station Forensics via Radio Frequency Fingerprinting," in *ACM ASIACCS*, 2018.

[14] G. Verma, P. Yu, and B. M. Sadler, "Physical layer authentication via fingerprint embedding using software-defined radios," *IEEE Access*, 2015.