ADA: Adversarial Dynamic Test Time Adaptation in Radio Frequency Machine Learning Systems

Shahriar Rifat[×], Michael De Lucia[‡], Ananthram Swami[‡],
Jonathan Ashdown[*], Kurt Turck[*] and Francesco Restuccia[×]
[‡] DEVCOM Army Research Laboratory, United States
[*] Air Force Research Laboratory, United States
[×] Institute for the Wireless Internet of Things, Northeastern University, United States

***Abstract*—Deep Neural Networks (DNNs) are an attractive solution to address several problems in Radio Frequency Machine Learning Systems (RFMLS). The key blocker that inhibits the widespread deployment of DNNs in real-world tactical scenarios is the performance degradation experienced under dynamic channel conditions. Test Time Adaptation (TTA) presents a promising solution to mitigate this issue by dynamically updating the DNN to adapt to the current channel conditions in an unsupervised manner. Although it offers superior performance and practical benefits, TTA introduces new security concerns and vulnerabilities, potentially exposing sensitive deployments to Adversarial Machine Learning (AML) activity. In this work, we introduce a novel attack strategy named *Adversarial Dynamic Adaptation* (ADA) that leverages the inherent vulnerabilities in TTA to compromise RFMLS tasks. We demonstrate that even under realistic assumptions and while perturbing only 20% of the samples in a test data batch, ADA degrades the performance of the unperturbed data by up to 20.3% compared to similar attacks designed for computer vision tasks. By assessing the robustness against the latest TTA methods, ADA serves as a valuable tool to identify and understand the security risks associated with adapting DNNs at the test time in mission-critical and sensitive deployments.**

I. Introduction

Despite the promising results shown by Radio Frequency Machine Learning Systems (RFMLS), previous work has revealed that the non-stationary, dynamic, and unpredictable nature of wireless channels can lead to a significant performance decline [1]. The key reason is that Deep Neural Networks (DNNs) used in RFMLS are trained using data collected under specific channel conditions and noise environments. Recent work in Test Time Adaptation (TTA) [2–4] has been proposed to address the problem of dynamic distribution shift at test time. In contrast with conventional RFMLS that rely on fixed source trained DNN to handle data in different channel conditions, TTA generates DNNs specialized for the current channel condition. In particular, TTA refines the base DNN whenever unlabeled data in unseen channel conditions becomes available. Subsequently, it conducts inference using the updated DNN customized for the specific channel condition. Empirical evidence has been shown to support the effectiveness of TTA primarily in computer vision tasks such as image classification [2], object detection [5], and document understanding [6]. Additionally, it has been shown [7] that TTA outperforms conventional RFMLS in dynamically changing channel conditions. In this paper, we demonstrate a security vulnerability of TTA for RFMLS, which is more severe than conventional Adversarial Machine Learning (AML) in wireless [8, 9].

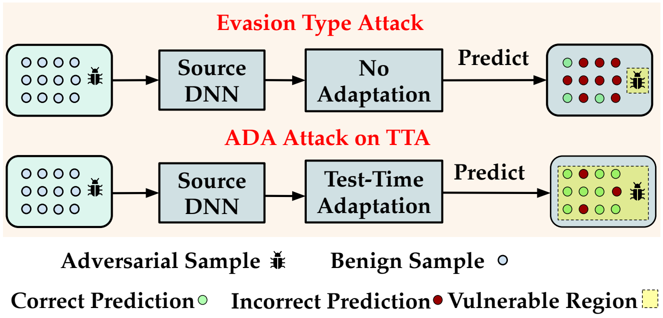


Fig. 1: Brief Overview of Conventional Evasion Type Attack against static DNN and ADA.

TTA constructs the final predictive DNN by considering the test batch rather than making predictions for a single test data point. The prediction for one entry within a batch is influenced by other entries in the same batch. Consequently, adversarial action on some data points can affect final predictions on samples that are unperturbed. Although such kind of security risks has been demonstrated in a recent work [10] for computer vision application, the proposed ADA attack is applicable in more general and practical scenarios. In [10], adversarial examples were crafted using cross-entropy loss that needs access to labeled samples. However, obtaining labeled samples in real time can be extremely challenging for wireless signals. However, our proposed ADA is completely agnostic to availability of labeled samples, and can generate attacks that achieve 20.3% increased error rate compared to a similar state of the art strategy [10], while assuming similar capabilities of the adversary. Our key contribution are summarized as follows:

- We analyze the security vulnerabilities of TTA for RFMLS through our proposed novel attack strategy *Adversarial Dynamic Adaptation* (ADA) which does not assume any access to labels, yet generates stronger attacks through our novel loss function (Section IV-A). Our attack can serve as a litmus test for deploying TTA in mission-critical tasks;

Approved for Public Release; Distribution Unlimited: AFRL-2024-2896.

- We evaluate our attack mechanism on three different state of the art TTA algorithms deployed for dynamic model adaptation and show that our proposed attack's efficacy is agnostic to different TTA algorithms and source DNN architecture.

The remainder of the paper is structured as follows. Section II provides some fundamental background to explain the motivation and distinctive properties of our approach as well as some related work on AML for wireless. In Section III, we describe the threat model of our adversarial attack and rationale for our assumptions on the adversary's capabilities. Section IV explains our approach in detail. In Section V, we describe our experimental setup and the related results. This is followed by a summary of the paper and concluding remarks in Section VI.

II. Background and Related Work

In this section, we introduce TTA and AML to better explain our ADA attack later and then review some related work on AML in the wireless context.

A. Test-Time Adaptation

Without loss of generality, we present TTA in the context of multi-class classification problem. We learn a prediction rule $f(\cdot;\theta_s)$ parameterized by the weights θ_s of a DNN. Given a training data set from the source domain $\mathcal{D}^s := \left\{(x_i^s, y_i^s)_{i=1}^{N_s}\right\}$ we learn $f(\cdot;\theta_s)$ through iterative minimization of some loss function $\mathcal{L}(f(X^s;\theta_s), Y^s)$ (e.g., cross-entropy loss), where $X^s := \{x_i^s\}_{i=1}^{N_s}$ and $Y^s := \{y_i^s\}_{i=1}^{N_s}$. The trained DNN is then deployed to perform inference on test dataset $\mathcal{D}^t := \left\{(x_i^t, y_i^t)_{i=1}^{N_t}\right\}$ where the label set $Y^t := \{y_i^s\}_{i=1}^{N_t}$ is unknown. In the static learning scenario, it is assumed that $\mathcal{D}^s$ and $\mathcal{D}^t$ are sampled from the same distribution. However, in real world deployment context, training and test data distribution may shift which results in poor performance [11] of $f(\cdot;\theta_s)$ on $X^t := \{x_i^s\}_{i=1}^{N_t}$. To address this issue, TTA, a paradigm of transductive learning has been proposed [12]. During inference, TTA initially acquires a function $f(\cdot;\theta_s)$ learned from the source training set or obtained from some online source. Subsequently, TTA adapts to the test data X^t without supervision and obtains the adapted DNN $f(\cdot;\theta_t)$. Some existing works [3, 13] on TTA have provided empirical evidence of performance improvement by only re-estimating the normalization statistics of Batch Normalization (BN) layers from test data. TTA is capable of characterizing the distribution of X^t to a certain degree, thus enhancing performance. The absence of supervision is typically covered by two unsupervised forms of losses.

Entropy Minimization. This line of TTA algorithms [2, 4, 14] minimizes the entropy of predictions over a batch of data $\{x_i^s\}_{i=1}^{B}$ to prevent collapsing to a trivial solution. Minimizing entropy improves the confidence of DNN, thus leading to improved performance on unseen test data.

Invariance Regularization. Invariance regularization based TTA algorithms perform some data-augmentation (e.g. rotation [15], adversarial perturbation [16]) on test data during inference. The inconsistency of the DNN's prediction on different augmented test data is leveraged as an unsupervised loss function to update the learnable parameters during inference. Maintaining consistency through this mechanism aids the DNN in attaining improved generalization across new data distributions, consequently enhancing test-time performance in the process.

B. Conventional Adversarial Machine Learning

To explain the distinctive property of the proposed attack strategy compared to traditional AML, two types of vulnerabilities of traditional AML are described.

Imperceptible Adversarial Examples. Most research focuses primarily on identifying imperceptible adversarial examples, which have been shown to be effective in deceiving DNNs during inference [17–19]. These adversarial examples are generated in an iterative manner through adding some bounded perturbation δ^* to the original sample $x^t_{adv} = x + \delta^*$ as follows:

$$\delta = \underset{\delta : \|\delta\|_p \leq \epsilon}{arg\,max}\, \mathcal{L}(f(x^t + \delta^{\cdot}\theta_s), y^t) \tag{1}$$

where $\mathcal{L}(.)$ is the loss function and $\|\delta\|_p \leq \epsilon$ is the l_p norm bound of the perturbation. Such perturbations must be directly injected into the test data.

Data Poisoning Attack. In such attacks, adversaries insert adversarially perturbed samples into a training set with the intention of causing the trained model to misclassify benign samples. In poisoning attacks one key assumption is that the adversary has access to the training data and training routine. However, in our ADA attack settings, an adversary doesn't need such kind of access to fullfill the attack objective.

C. AML for RFMLS

Numerous previous works [8, 9] have demonstrated the efficacy of different attacks on RFMLS . For example, studies like [20, 21] have studied exploratory attacks, in which adversaries construct a DNN to understand the transmission patterns in a certain channel condition to disrupt transmissions that would have been successful otherwise. Evasion type attacks [9, 22, 23] in wireless assume access to a DNN's gradient information to calculate bounded perturbation, which added to the input forces incorrect predictions out of the DNN. In [24, 25] over-the-air spectrum poisoning attacks have been investigated in which adversaries manipulate a transmitter's spectrum sensing data by transmitting signals during the victim transmitter's spectrum sensing period. Works such as [26] have investigated Trojan attacks against modulation classifiers by subtly altering training data, which manifest as phase shifts and are activated during test time. Our research introduces a paradigm shift in attack scenarios, presenting a fundamentally novel approach in which adversaries can execute attacks without directly modifying samples or gaining access to the victim's training routine.

III. Problem Statement

Reusing notation in section II-A, we assume that a user (victim) has obtained a DNN $f(\cdot;\theta_s)$, trained on source data $\mathcal{D}^s := \left\{(x_i^s, y_i^s)_{i=1}^{N_s}\right\}$ to solve a multi-class wireless classification problem. However, due to changing channel condition the distribution of the test data is continuously changing which manifests as covariate shift, $(p(X^s) \neq p(X^t) \, and \, p(X^s|Y^s) \neq p(X^t|Y^t)$. To maintain optimum performance while the data distribution changes, the victim deploys a TTA algorithm with only the test data available in batch mode $X_B^t := \{x_i^t\}_{i=1}^{N_B}$. The objective of the victim can be described as follows:

$$\theta_t^*(X_B^t) = \underset{\theta_t^A \subset \theta_s}{\arg\min} \, \mathcal{L}_{TTA}(X_B^t; \theta_t^A) \tag{2}$$

Here, θ_t^A indicates all adaptable parameters in the source model including the normalization statistics and $\mathcal{L}_{TTA}$ indicates the unsupervised loss being used by the TTA algorithm. We assume a scenario in which a portion of the samples in a test batch $X_{com}^t \in X_B^t$ has been exposed to and compromised by an adversary whereas adversary cannot modify the other portion of the data $X_{B\setminus com}^t$. Throughout the rest of the paper, we will refer to these samples which the adversary can modify as exposed samples for brevity. However, we assume that the adversary operate in a white-box setting in which the adversary possesses knowledge of the pre-adapted model parameters and has some form of read-only access to benign samples $X_{B\setminus com}^t$ in the test batch (for instance, involving a malicious insider). The goal of the adversary is to craft perturbation δ_{com}^t such that the overall error rate of prediction in the benign data within the same batch increases after adaptation.

IV. Proposed ADA Approach

In this section, we explain the ADA approach. We start by discussing the compatibility of our proposed loss in practical deployment scenarios and appropriate selection of the perturbation bound. Then we provide details on crafting adversarial samples that can effect unperturbed samples.

A. Loss Function

To construct adversarial samples, the adversary needs to calculate the perturbation gradients by iterative minimization of some loss function $\mathcal{L}_{adv}$. If we assume that a malicious insider is providing the adversary with accurate labels $Y_{B\setminus com}^t$ of the read-only samples as in [10], the choice of $\mathcal{L}_{adv}$ for the adversary could be the loss of cross-entropy $\sum_{x_i \in X_{B\setminus com}} \sum_{c \in \mathcal{C}} y_i^c \log(f(x_i;\theta_t))$, where $\mathcal{C}$ is the set of classes. However, labeling wireless data through manual inspection is not straightforward like computer vision tasks, thus it is not realistic to use cross-entropy loss that needs access to labels. Assuming white box access, we design a loss function $\mathcal{L}_{feat}$ (Equation 3) that pulls the feature representations across classes towards the center by iterative minimization of the distance of each benign samples feature and the centriod in the penultimate layer of the DNN as:

$$\begin{aligned}\mathcal{L}_{feat} = \sum_{x_i \in X_{B\setminus com}^t} & d(\;\; f_{k-1}(x_i;\theta_{pre}) \\ & - \frac{1}{\left|X_{B\setminus com}^t\right|} \sum_{x_i \in X_{B\setminus com}^t} f_{k-1}(x_i;\theta_{pre} \;\;)\end{aligned} \tag{3}$$

$f_{k-1}(.)$ denotes the penultimate layer of the DNN assuming it has k layers, d denotes some distance (e.g. L1, Mean Square Error (MSE) metrics) and θ_{pre} is the model parameters before adaptation. Equation 5 measures the distance metric between the feature centroid of the batch and feature vector of each sample. As filtering out less-confident (high-entropy) samples is commonly used to improve robustness in TTA [14, 28], to enforce generation of low-entropy adversarial samples, we add the following additional loss term:

$$\mathcal{L}_{entropy} = \sum_{x_i \in X_{com}} f(x_i;\theta_{pre}) \, \log(f(x_i;\theta_{pre})) \tag{4}$$

The final loss for adversarial action thus becomes :

$$\mathcal{L}_{adv} = \mathcal{L}_{feat} + \lambda \cdot \mathcal{L}_{entropy} \tag{5}$$

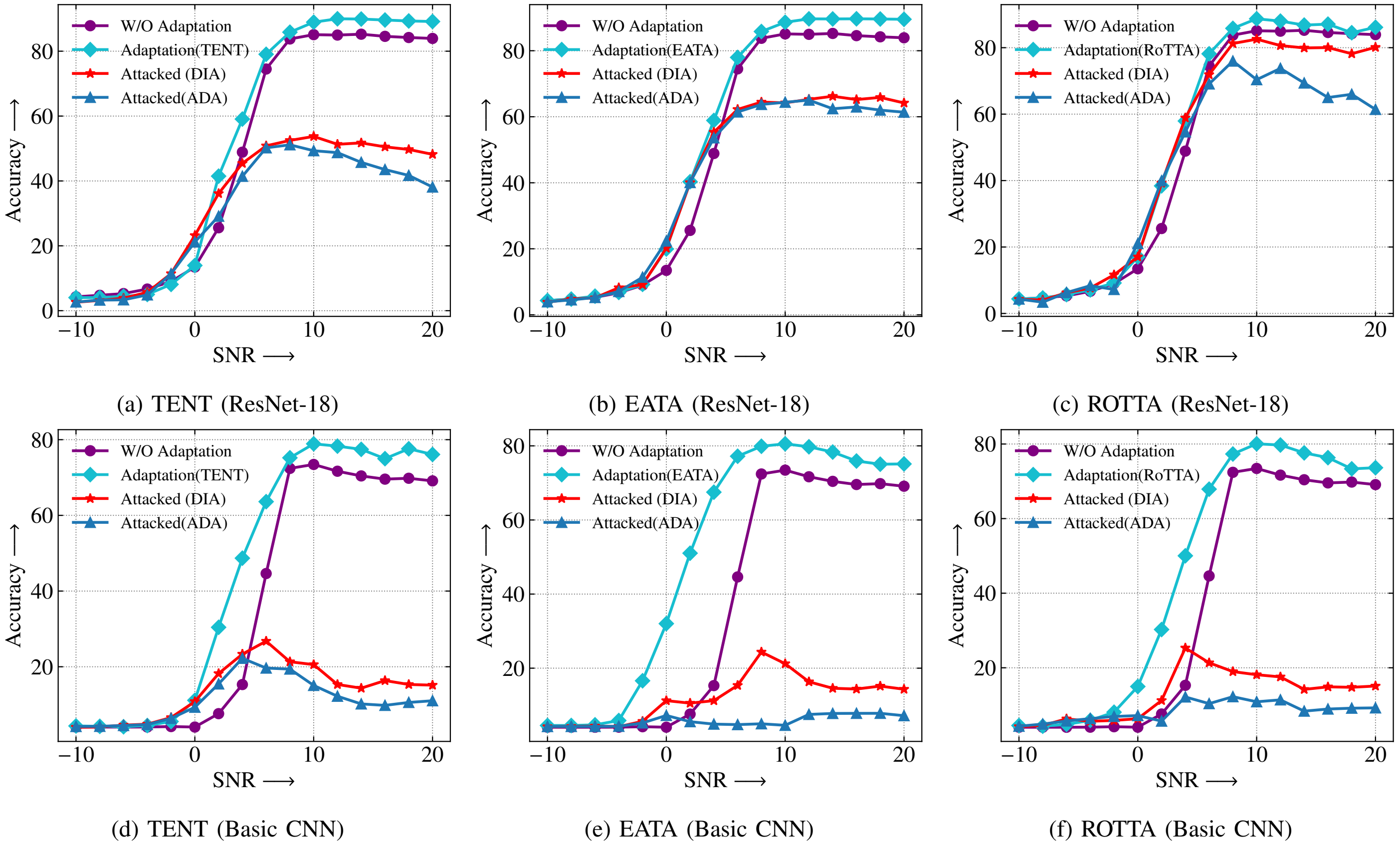
Here, λ is the regularization constraint to control the effect of the loss term of the entropy. Note that our proposed loss does not assume access to labeled samples as in [10], therefore it is generalizable across different application scenarios.

$$\max_{\delta_{com}^t : \, \|\delta_{com}^t\|_p \leq \epsilon} \mathbb{L}(\cdot; \theta_t^*((X_{com}^t + \delta_{com}^t) \cup X_{B\setminus com}^t) \tag{6}$$

Here, $\|.\|_p$ is the l_p norm bound used in adversarial attacks for stealthiness. For our case we choose l_2 norm $\|.\|_2$ as it would bound the perturbation power thus camouflaging the adversarial action.

B. Generation of Adversarial Perturbation

The process of crafting the adversarial perturbation in ADA is an iterative process described in detail in Algorithm 1. From the perspective of an attacker, ADA can be formulated as a bilevel optimization problem where a victim is trying to improve the predictive performance of DNN using some TTA algorithm (2), and in outer maximization the attacker is trying to increase the overall classification error in benign samples $X_{B\setminus com}^t$. However, since we assume that the attacker does not have information about the TTA algorithm that is being used, it simply tries to add perturbations to the compromised samples (line 7) with the objective of increasing the error rate of the pre-adapted model $f(;\theta_{pre})$ in $X_{B\setminus com}^t$. The ADA adversary calculates the gradients of the perturbation δ_{com} with respect to the proposed adversarial loss $\mathcal{L}_{adv}(\cdot)$ (Line 8). Finally, the optimal perturbation δ^* for the attack is calculated utilizing the Projected Gradient Descent (PGD) approach used in [29] (line 9-10). It should be noted that in the ADA settings, the adversary changes the perturbation budget ϵ according



(a) TENT (ResNet-18)

(b) EATA (ResNet-18)

(c) ROTTA (ResNet-18)

(d) TENT (Basic CNN)

(e) EATA (Basic CNN)

(f) ROTTA (Basic CNN)

Fig. 2: Attack performance benchmarking for ADA across different TTA algorithms. As source DNN 1D-ResNet used in [7] (top row) and basic Convolutional Neural Network (CNN) used in [27] (bottom row) have been used.

to the channel condition to achieve stealthiness and highly efficient attacks. Specifically, in ADA settings, the adversary sets the perturbation constraint ϵ to $0.063\times$ the peak value of the corresponding I and Q channels of the current batch of data [1].

V. Performance Evaluation

A. Description of Dataset

We used the RadioML 2018.01A dataset [27] to investigate the vulnerabilities of various TTA algorithms within our ADA attack framework. This dataset encompasses 24 commonly employed modulation classes across a range of signal-to-noise ratios (SNR). Initially, we partition the dataset for each SNR into training and testing subsets, with 80% of the data allocated for training and 20% for testing. To ensure robustness of our evaluations, we employ three distinct seeds to create random splits, and all reported performance metrics represent the average results obtained from these three runs. Additionally, we exclude data with extremely low SNR $(< -10dB)$ as it may not be indicative of our task, and data with exceedingly high SNR $(> 20dB)$ as performance tends to saturate in this range.

B. Experimental Setup

For the source model from which a victim begins adaptation, we employ two 1D CNN-based architectures, as this is a widely adopted approach in various AML studies [23, 30]. Nonetheless, our ADA framework can be applied to various architecture choices for the source model, as long as they are compatible with the respective TTA algorithm. As the deeper DNN, we use the Residual Network (ResNet) structure used with 18 convolution layers (ResNet-18) as the source model for adaptation. We input the I/Q samples to the DNN as a two-channel sequence. We keep the kernel size and number of channels the same as the original ResNet model architecture [31] except that all the operations proposed for images were replaced by corresponding 1D alternative (e.g., Conv1D, Maxpool1D). As a shallower architecture, we select the DNN proposed in [27] without residual connections. To obtain the source model, we train DNN for 600 epochs with Adam Optimizer with learning rate =0.001 on training data from the 10dB SNR level. Test data from other SNR levels are used for inference and adaptation of DNN. Unless otherwise specified, for all reported results, we randomly sampled

[1] It has been found that across SNR range -10 dB to 20 dB, setting the perturbation budget in this way keeps the noise floor shift within 1% across all batches.

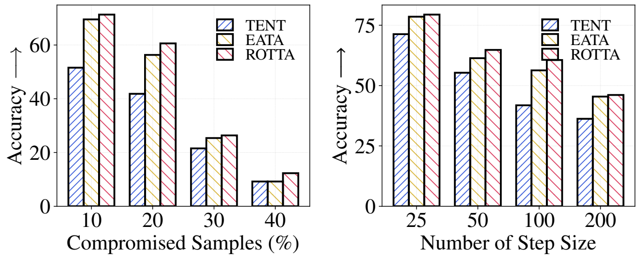
Algorithm 1: ADA Algorithm

```
Input: A test data batch $X_B^t$ ; Adversarial learning rate $\alpha$; Pre adapted model parameters $\theta_{pre}$; Number of adversarial iteration steps $n$; Number of compromised samples $n_{com}$.
Define: Adversarial perturbation $\delta_{com} = \mathbf{0}^{n_{com} \times 2 \times 1024}$; perturbation constraint $\epsilon$.
Output: Adversarial perturbation vector $\delta_{com}^*$
Randomly select data samples $X_{com}^t \subset X_B^t$, $|X_{com}^t| = n_{com}$ as compromised samples to be perturbed
for step = 1, 2, ....., n do
    Calculate perturbation constraint $\epsilon$ for current batch of data
    Add adversarial perturbation to the compromised samples $X_B^t = \{X_{com}^t + \delta_{com}\} \cup X_{B \setminus com}$
    Calculate the adversarial loss $\mathcal{L}_{adv}(X_{B \setminus com};\ \theta_{pre})$ using Equation 5
    Calculate the perturbation gradient $grad = \bigtriangledown_{\delta_{com}} \mathcal{L}_{adv}(X_{B \setminus com};\ \theta_{pre})$
    Update $\delta_{com}^*$ through gradient projection $\delta_{com} \leftarrow -\frac{\delta_{com} + \alpha \cdot sign(grad)}{\|\delta_{com} + \alpha \cdot sign(grad)\|_2} * \epsilon$
end for
Return: $X_{com}^t + \delta_{com}^*$
TEST:
Update the parameters with any TTA algorithm $\mathbf{TTA}(\cdot)$, : $\theta_A^* \leftarrow \mathbf{TTA}(\theta_A)$; $\theta_t^* = \theta_A^* \cup \{\theta_{pre} \setminus \theta_A\}$
Check the performance of $X_{B \setminus com}^t$ with updated DNN $f(X_{B \setminus com}^t; \theta_t^*)$
```

20% data from each test batch consisting of 64 samples as compromised samples. We set the step size α for adversarial perturbation creation to 0.1 and the number of PGD steps to 100. We evaluated the performance of three state-of-the-art TTA methods, namely TENT [2], EATA [14] and ROTTA [28], against attacks using our ADA framework. We follow the original implementation of TENT and EATA. As the algorithm ROTTA involves some augmentations specific to images (e.g. color jitters), we replace such augmentations with Gaussian noise ($\mu = 0; \sigma = [0.01, 0.02, 0.03]$). The reported accuracy values excludes the adversarial samples and are calculated only based on the unperturbed normal samples. The value of λ in Equation 5 is chosen to be 1. However, empirically we have found that any value within range of 1 to 3 provides nearly identical attack performance.

C. ADA *Performance Across Different Baseline TTA methods*

To assess the effectiveness of our proposed approach, we conduct adversarial actions using Distribution Invading Attack (DIA) [10] under identical conditions as ADA. As depicted in Figure 2, our ADA attack demonstrates efficacy across various TTA algorithms and source DNN architectures. In particular, our framework outperforms the closest comparable baseline DIA by up to 20.3%. Despite TENT's effectiveness in enhancing performance, it exhibits the highest vulnerability to adversarial actions during adaptation. The ADA framework marginally outperforms DIA by only 3.12% during adaptation using EATA, as this process involves explicitly filtering out high-entropy samples. The attack exhibits increased potency in the high SNR regime, where the model performs well even in the absence of adaptation. This suggests that merely improving the generalization performance of the TTA algorithms does not inherently ensure enhanced robustness.



(a) Varying % of Compromised Samples

(b) Varying iteration steps of finding perturbation

Fig. 3: Performance of ADA for across different hyper parameters. The reported values are the average of the SNR range (0-20) dB

D. *Effect of DNN Architecture Choice on Attack Performance*

As depicted in Figure 2, the choice of source model architecture has significant influence on the performance of TTA under attack. We observe that irrespective of the architecture complexity and generalization performance our proposed ADA framework is effective in uncovering the susceptibility of TTA algorithms. Moreover, it's evident that the basic CNN is more susceptible to attacks compared to the ResNet-18 model. This observation aligns with previous findings [32] indicating that architectures with greater capability and improved generalization performance tend to exhibit better overall robustness.

E. *Effect of the Portion of Compromised Samples and Iteration Steps*

In Figure 3, we analyze the performance of the ADA attack on varying proportions of compromised samples and the number of iteration steps involved to craft adversarial perturbations. In particular, with only 10% of the samples compromised, TENT shows the worst performance degradation (9.23%) compared to static inference with the source model. However, when 40% of samples are compromised, all three TTA algorithms degrade to performing random predictions. Regarding the effect of step size, our proposed ADA framework requires a reasonable number of iterations to craft the perturbation vector for an effective attack. In scenarios where an ADA adversary operates under a constrained budget (e.g., 25 iteration steps), the utilization of TTA algorithms can offer improved performance compared to employing inference solely based on the source DNN.

VI. Conclusion

This paper presents a novel attack strategy designed to exploit vulnerabilities inherent in TTA within RFMLS tasks. Unlike previous approaches, ADA does not rely on access to labeled samples, which makes it particularly suitable for RFMLS scenarios where obtaining labeled data can be challenging for adversaries. The effectiveness of ADA is demonstrated in various TTA algorithms and source model architectures. The proposed framework can function as a security test benchmark for the TTA before deployment in real-world mission-critical tasks. By comprehending these vulnerabilities, practitioners can advance the development of more secure and resilient RFMLS performing TTA, thereby safeguarding them against potential threats in dynamic channel conditions.

Acknowledgment of Support and Disclaimer

This work is funded in part by the Air Force Office of Scientific Research under contract number FA9550-23-1-0261, by the Office of Naval Research under contract number N00014-23-1-2221, as well as by National Science Foundation grants CNS-2134973, CNS-2120447, ECCS-2229472, CNS-2312875, and ECCS-2329013. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the US Government.

References

[1] 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," *Proc. of IEEE Conference on Computer Communications (INFOCOM)*, 2020.

[2] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *International Conference on Learning Representations*, 2021.

[3] S. Schneider, E. Rusak, L. Eck, O. Bringmann, W. Brendel, and M. Bethge, "Improving robustness against common corruptions by covariate shift adaptation," *Advances in neural information processing systems*, vol. 33, pp. 11539–11551, 2020.

[4] S. Goyal, M. Sun, A. Raghunathan, and J. Z. Kolter, "Test time adaptation via conjugate pseudo-labels," *Advances in Neural Information Processing Systems*, vol. 35, pp. 6204–6218, 2022.

[5] S. Sinha, P. Gehler, F. Locatello, and B. Schiele, "Test: Test-time self-training under distribution shift," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2759–2769, 2023.

[6] S. Ebrahimi, S. O. Arik, and T. Pfister, "Test-time adaptation for visual document understanding," *arXiv preprint arXiv:2206.07240*, 2022.

[7] S. Rifat, J. Ashdown, K. Turck, and F. Restuccia, "Zero-shot dynamic neural network adaptation in tactical wireless systems," in *MILCOM 2023-2023 IEEE Military Communications Conference (MILCOM)*, pp. 418–423, IEEE, 2023.

[8] W. Zhang, M. Krunz, and G. Ditzler, "Stealthy adversarial attacks on machine learning-based classifiers of wireless signals," *IEEE Transactions on Machine Learning in Communications and Networking*, 2024.

[9] F. Restuccia, S. D'Oro, A. Al-Shawabka, B. C. Rendon, K. Chowdhury, S. Ioannidis, and T. Melodia, "Generalized wireless adversarial deep learning," in *Proceedings of the 2nd ACM Workshop on Wireless Security and Machine Learning*, pp. 49–54, 2020.

[10] T. Wu, F. Jia, X. Qi, J. T. Wang, V. Sehwag, S. Mahloujifar, and P. Mittal, "Uncovering adversarial risks of test-time adaptation," *arXiv preprint arXiv:2301.12576*, 2023.

[11] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," *arXiv preprint arXiv:1903.12261*, 2019.

[12] V. N. Vapnik, V. Vapnik, *et al.*, "Statistical learning theory," 1998.

[13] T. Gong, J. Jeong, T. Kim, Y. Kim, J. Shin, and S.-J. Lee, "Note: Robust continual test-time adaptation against temporal correlation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 27253–27266, 2022.

[14] S. Niu, J. Wu, Y. Zhang, Y. Chen, S. Zheng, P. Zhao, and M. Tan, "Efficient test-time model adaptation without forgetting," in *International conference on machine learning*, pp. 16888–16905, PMLR, 2022.

[15] Q. Wang, O. Fink, L. Van Gool, and D. Dai, "Continual test-time domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7201–7211, 2022.

[16] A. T. Nguyen, T. Nguyen-Tang, S.-N. Lim, and P. H. Torr, "Tipi: Test time adaptation with transformation invariance," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24162–24171, 2023.

[17] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[18] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*, pp. 39–57, Ieee, 2017.

[19] Y. Vorobeychik, M. Kantarcioglu, R. Brachman, P. Stone, and F. Rossi, "Adversarial machine learning," 2018.

[20] Z. Luo, S. Zhao, Z. Lu, J. Xu, and Y. E. Sagduyu, "When attackers meet ai: Learning-empowered attacks in cooperative spectrum sensing," *IEEE Transactions on Mobile Computing*, vol. 21, no. 5, pp. 1892–1908, 2020.

[21] T. Erpek, Y. E. Sagduyu, and Y. Shi, "Deep learning for launching and mitigating wireless jamming attacks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 1, pp. 2–14, 2018.

[22] B. Flowers, R. M. Buehrer, and W. C. Headley, "Evaluating adversarial evasion attacks in the context of wireless communications," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1102–1113, 2019.

[23] M. Z. Hameed, A. György, and D. Gündüz, "The best defense is a good offense: Adversarial attacks to avoid modulation detection," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1074–1087, 2020.

[24] Y. Shi, T. Erpek, Y. E. Sagduyu, and J. H. Li, "Spectrum data poisoning with adversarial deep learning," in *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*, pp. 407–412, IEEE, 2018.

[25] Y. E. Sagduyu, Y. Shi, and T. Erpek, "Adversarial deep learning for over-the-air spectrum poisoning attacks," *IEEE Transactions on Mobile Computing*, vol. 20, no. 2, pp. 306–319, 2019.

[26] K. Davaslioglu and Y. E. Sagduyu, "Trojan attacks on wireless signal classification with adversarial machine learning," in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pp. 1–6, IEEE, 2019.

[27] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air Deep Learning Based Radio Signal Classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, 2018.

[28] L. Yuan, B. Xie, and S. Li, "Robust test-time adaptation in dynamic scenarios," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15922–15932, 2023.

[29] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[30] S. Zhang, Y. Lin, J. Yu, J. Zhang, Q. Xuan, D. Xu, J. Wang, and M. Wang, "Hfad: Homomorphic filtering adversarial defense against adversarial attacks in automatic modulation classification," *IEEE Transactions on Cognitive Communications and Networking*, 2024.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[32] H. Huang, Y. Wang, S. Erfani, Q. Gu, J. Bailey, and X. Ma, "Exploring architectural ingredients of adversarially robust deep neural networks," *Advances in Neural Information Processing Systems*, vol. 34, pp. 5545–5559, 2021.