Application of Adversarial Machine learning in Protocol and Modulation Misclassification

Marwan Krunz, Wenhan Zhang, and Gregory Ditzler

University of Arizona, Tucson, AZ 85721, USA

ABSTRACT

This paper explores the application of adversarial machine learning (AML) in RF communications, and more specifically the impact of intelligently crafted AML perturbations on the accuracy of deep neural network (DNN) based technology (protocol) and modulation-scheme classifiers. For protocol classification, we consider multiple heterogeneous wireless technologies that operate over shared spectrum, exemplified by the coexistence of Wi-Fi, LTE LAA (Licensed Assisted Access), and 5G NR-Unlicensed (5G NR-U) devices in the unlicensed 5 GHz bands. Time-interleaving-based spectrum sharing is assumed. Given a window of received I/Q samples, a legitimate DNN-based classifier (called the *defender's classifier*) is often used to identify the underlying protocol/technology. Similarly, DNN classifiers are often used to discern the underlying modulation scheme. For both types of classifiers, we study an attack model in which an adversarial device eavesdrops on ongoing transmissions and uses its own *attacker's classifier* to generate low-power AML perturbations that significantly degrade the accuracy of the defender's classifier. We consider several DNN architectures for protocol and modulation classification (based on recurrent and convolutional neural networks) that normally exhibit high classification accuracy under random noise (i.e., AWGN). By applying AML-generated perturbations, we show how the accuracy of these classifiers degrades significantly, even when the signal-to-perturbation ratio (SPR) is high. Several attack vectors are formulated, depending on how much knowledge the attacker has of the defender's classifier. On the one extreme, we study a "white-box" attack, whereby the attacker has complete knowledge of the defender's classifier and its training dataset. We gradually relax this assuming, ultimately considering an almost "black-box" attack. Mitigation techniques based on AML training are presented and are shown to help in countering AML attacks.

Keywords: Shared spectrum, signal classification, deep learning, adversarial machine learning, wireless security.

1. INTRODUCTION

The demand for wireless capacity continues to outgrow spectrum availability, especially at low and mid bands (e.g., sub-6 GHz). Various spectrum sharing architectures have been proposed to utilize the spectrum more efficiently. For example, a three-tiered spectrum authorization access system was proposed for the Citizens Broadband Radio Service (CBRS) band, which enables commercial users to coexist with incumbent federal and non-federal users.[1,2] A listen-before-talk (LBT) approach was adopted for the some of the Unlicensed National Information Infrastructure (UNII) bands (5.15 – 5.85 GHz) to allow LTE license assisted access (LAA) and 5G New Radio unlicensed (NR-U) cellular technologies to share the spectrum with Wi-Fi devices.[3–5] The coexistence of various technologies inevitably introduces cross-technology interference, which may be inadvertent (e.g., due to hardware malfunction or protocol incompatibility) or adversarial (e.g., rogue devices that wish to disrupt legitimate communications). A protocol (or technology) classifier can be used to rapidly identify an ongoing transmission without decoding its signal and accordingly determine the legitimacy or otherwise of such a transmission. Use cases for protocol classifiers include traffic estimation, fair allocation of the channel airtime, identification of fake signals, and more.

Further author information: (Send correspondence to Marwan Krunz)
Marwan Krunz: E-mail: krunz@arizona.edu
Wenhan Zhang: E-mail: wenhanzhang@email.arizona.edu
Gregory Ditzler: E-mail: ditzler@arizona.edu

Conventional techniques for technology classification rely on unique features of the underlying protocols, such as the periodicity of the cyclic prefix (CP).[6] Many of these techniques are derivatives of spectrum sensing for opportunistic (primary/secondary) access, where the main goal is to identify whether or not a particular primary signal is present. These techniques are ineffective when the underlying technologies exhibit similarities in their protocol semantics, e.g., the CP duration in LTE is one of the possible CP durations in 5G NR. Further, extracting the features of interest often involves processing the signal over a long period to compute, for example, its autocorrelation structure or to cross-correlate its samples against a priori known and unique part of it (e.g., the preamble). Such delay can be problematic for fast spectrum adaption.

Alternatively, technology identification can be performed using deep neural network (DNN) classifiers, which can be trained using the In-phase (I) and Quadrature (Q) components of the received samples. Several such classifiers have been proposed in the literature.[7–12] The authors successfully applied a convolutional neural network (CNN) based classifier to fingerprint Wi-Fi devices using I/Q data with deliberately added imperfections.[8] Other authors applied CNN-based classifiers for dynamic spectrum sharing.[9] Their models can detect the user category (i.e., in/out network) and packet delivery rate. DNN classifiers were also applied to detect a priori unknown spoofed signals.[10]

Despite their high classification performance, DNN-based classifiers can be vulnerable to AML attacks.[13–16] Such attacks have been studied in object classification/recognition problems,[17] and more recently in RF signal classification.[18] The general idea is to train a surrogate DNN classifier, hereafter called the *attacker's classifier*, to produce intelligently crafted perturbations. These perturbations, when combined with the original signal, can mislead the *defender's classifier*, causing it to incorrectly classify the received samples (see Figure 1). Note that in contrast with image detection settings, where the same training dataset is used at both the attacker's and defender's classifiers, different datasets may be used for RF signal classification to account for differences in the channel conditions between the attacker and defender. Recent research efforts investigated AML perturbations for baseband signal classification.[18–21] However, an idealized "oracle" attack model was assumed, where the attacker has complete knowledge of the defender's classifier. In practical RF environments, the attacker has limited information of the defender's classifier (e.g., it only knows the type of layers in the DNN). As shown in this paper, the amount of information that the adversary has of the defender's classifier can have profound impact on the attack's strength. In addition, the received waveforms for the attacker can also be different from the defender due to the various channel conditions. We explore such aspects and propose possible defense mechanisms against AML attacks. Specifically, we study the impact of augmenting the defender's training dataset with properly tuned AML perturbations. Our results show that the proposed defense scheme can improve the robustness of the DNN model under perturbations of different power levels.

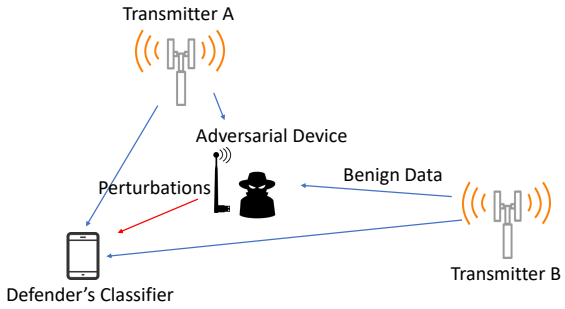

Figure 1. AML-based attack on the defender's signal classifier.

2. MACHINE LEARNING-BASED SIGNAL CLASSIFICATION

This section proposes three DNN structures that can accurately classify signal protocol or modulation schemes. Other works considered machine learning for RF signal classification;[8–10] however, these approaches only used the moving filter to capture the pattern behind a given signal but neglected the dependency naturally existing in the waveform. The general idea behind using a DNN is that the samples associated with I/Q components can be

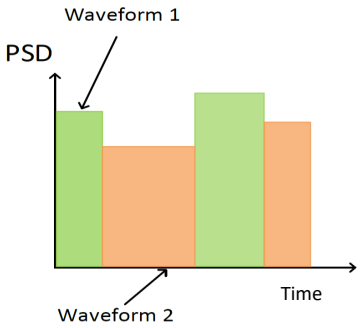

Figure 2. The power spectral density (PSD) vs. time for spectrum sharing in an interleaving model.

fed into a DNN trained to recognize a particular type of signal. In contrast to traditional feature-based spectrum sensing, DNN-based classification is data-driven, and does not require explicit specification of any technology-dependent features. We build an *interleaving spectrum sharing* model (as shown in Figure 2), where only one type of technology can be active at a given time. We then consider the modulation and protocol classification problems under such model.

2.1 Modulation Classification

Previous DNN research focused on the CNN structure where the filter (or kernel) is used to capture features hidden in a segmented sequence of I/Q samples. Further, real-time orthogonal frequency division multiplexing (OFDM) signal modulation classification has recently been investigated using DNNs and software-defined radio.[22] For example, Zhang et al. generated a modulation classification dataset under a dynamic fading channel, including six types of OFDM modulation signals. Triple-skip residual stack (a kind of CNN) was trained and showed high accuracy even with low SNR signals. Other works extended DNNs to signals with different modulations, channel conditions, and other available resources. Garrett et al.[23] applied branch CNNs to achieve approximately 80% classification accuracy averaged on 29 types of signals, including multi-carrier signals and higher-order modulations. O'Shea et al. [24] investigated the DNN-based modulation classification tasks for digital and analog signals. The authors[24] also shared a dataset that consists of eleven modulation schemes. In addition, a CNN-based classifier is designed and trained to classify the modulation of a given signal. The dataset and the classifier are published as RML.2016a and VT-CNN2, respectively. In this work, we verify the effectiveness of the VT-CNN2 structure (as shown in Figure 3 (c)) for modulation scheme classification then we evaluate VT-CNN2's performance under the proposed AML attacks.

2.2 Protocol Classification

Most existing DNN-based RF classification efforts focus on the modulation scheme identification;[22–24] however, there are many other supervised classification use cases for DNNs in wireless communication. We also seek to explore how effective these approaches can classify wireless network protocols. Therefore, in addition to modulation classification, we also seek to classify wireless protocols over the shared bands. While there are many possible protocols to classify, we have selected a limited number of protocols. The three protocols examined in this work have several modulation schemes in common operated over the same bandwidth. Hence, modulation classification techniques cannot differentiate between the waveforms of such protocols, which calls for new machine learning-based classifiers that capture other protocol-related embedded features in the observed transmissions.

A traditional protocol detection approach relies on a time-correlated view of the received sequences, which is similar to how a recurrent neural network (RNN) captures temporal dynamic behavior over the input sequence. Therefore, we investigate the application of RNNs for heterogeneous protocol classification over a shared spectrum, focusing on Wi-Fi, LTE-LAA, and 5G NR-U in the unlicensed 5 GHz bands. Next, we consider more advanced RNN models by incorporating the bidirectional and multi-layer (hierarchical) gated structures. These gates can balance the influence of the most recent input and the trained state during the recurrence. In addition,

the bidirectional layer(s) allow RNNs to simultaneously use past (backward) and future (forward) states during the training. Hence, the bidirectional design can help the RNN detect the backward dependency that is difficult to capture by the forward structure. Finally, the proposed DNNs shown in Figure 3 are trained to classify the wireless protocols and evaluate the adversarial perturbations.

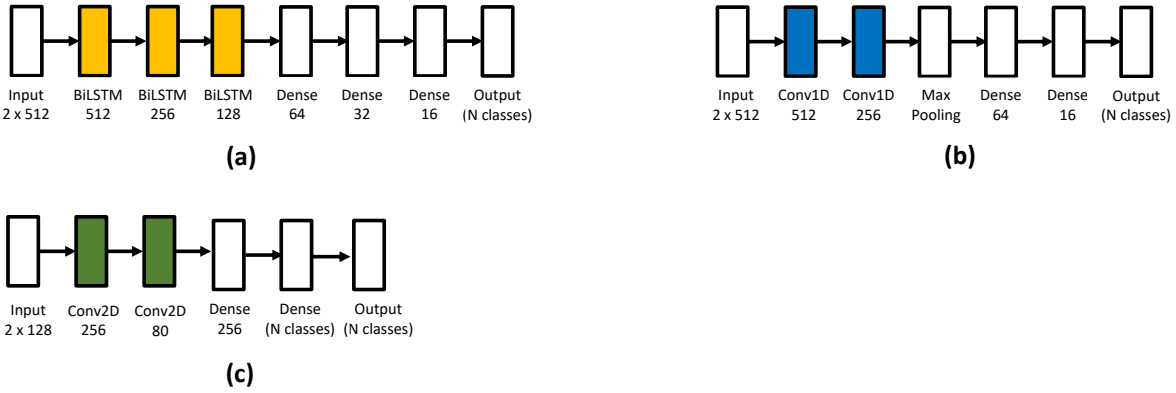

Figure 3. DNN structures used for RF signal classification. Structures (a) is an RNN-based classifiers, structures (b) is a traditional CNN classifier, and structure (c) is the VT-CNN2 classifier.

3. ADVERSARIAL MACHINE LEARNING PERTURBATIONS

This work considers the wireless communication system shown in Figure 1. In this setting, the legitimate user – or *defender* – has a DNN trained to classify the received signals, whereas the transmitter sends one of several possible protocols or modulation schemes in an interleaved manner. The attacker, shown as the adversary device in Figure 1, overhears the transmitter's signals (called *benign data*) and uses the signals to train its own DNN. The purpose of the adversary's DNN is to generate adversarial signals that a virtually indistinguishable from the benign signal; however, the perturbation leads to deleterious predictions. In our model, the defender will receive the benign transmission along with the perturbation generated by the adversary, resulting in the misclassification of the received samples.

The communication scenario in Figure 1 can be formulated as follows: we define $\mathbf{H}_{td}$, $\mathbf{H}_{ta}$, and $\mathbf{H}_{ad}$ as the channel matrices between the transmitter and defender, the transmitter and attacker, and the attacker and defender, respectively. We assume AWGN (n) at any receiving device and denote x_o as the transmitted waveform. Therefore, the defender receives $x_d^t = \mathbf{H}_{td}x_o + n_d$ and the attacker receives $x_a^t = \mathbf{H}_{ta}x_o + n_a$. To launch an AML attack, the adversary uses its received signal x_a^t to generate and transmit the perturbations η. Under this attack, the defender receives $x_d^{t,a} = \mathbf{H}_{td}x_o + \mathbf{H}_{ad}\eta + n_d$. We assume the attacker is close enough to the defender, so the interference received by the defender is approximated equals to the η generated by the attacker.

Once the adversary has access to a signal x_d^t, they can generate a perturbation to produce a new signal that is crafted to fool the defender's DNN. We use the Fast Gradient Sign Method (FGSM) in this work to generate the perturbation because FGSM is widely accepted as a benchmark in the AML community.[13,25] This technique uses the gradients of a neural network to generate a perturbation η and, subsequently, the adversarial data x_{adv}. Typically, a DNN is expected to predict the same class for x and its neighboured input $x_{adv} = x + \eta$ if every element of η is less than the threshold. This logic is because a classifier is trained to find the boundary of different classes, the input with a similar feature can be assigned to the same class (e.g., x and x_{adv}) if $||\eta||_\infty < \epsilon$. Nevertheless, the adversary's goal is to amplify the effect of this small η so that the targeted classifier cannot accurately classify x_{adv}.

We simplify the DNN as the mapping function $f(x;\theta) : \mathbb{R}^{n\times 2} \mapsto \mathcal{Y}$ where n are the number of samples and $\mathcal{Y}$ is the set of outcomes (e.g., modulation format, etc.). The adversary crafts the sample x_{adv} in a way such that $\|x - x_{adv}\| \neq \eta$. Even though the difference between the x_{adv} and x is small, the difference between the

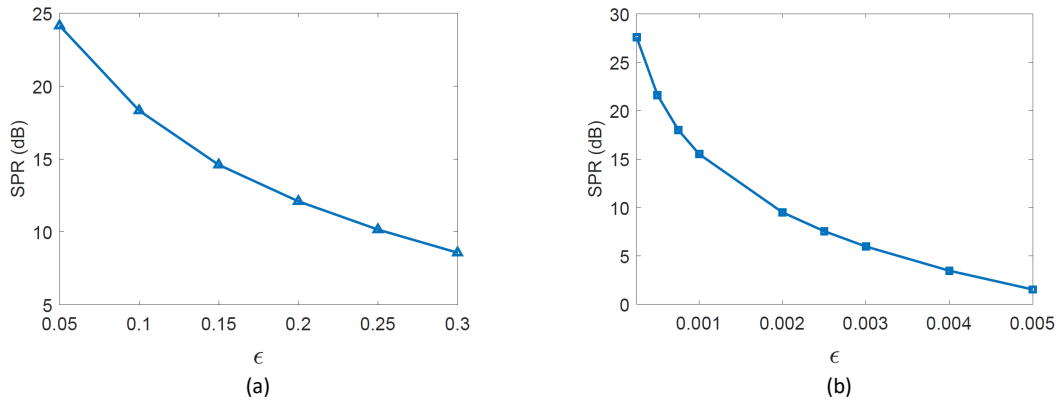

Figure 4. Relationship between SPR and ϵ: (a) protocol classification dataset, and (b) modulation classification dataset.

outputs $f(x+\eta;\theta)$ and $f(x;\theta)$ is not minor. In fact, these small perturbations that are generated based on the gradient of the loss correspond to large changes in the output space, which can make them difficult to detect. As a result, we can expect that small perturbations at the input can result in a change in the network's label from the benign sample. The adversarial perturbation is formally given by $\eta = \epsilon \text{sign}(\nabla_x L(x,y;\theta))$, where $L(x,y;\theta)$ is the cross entropy loss function with parameters θ.[13] The adversarial data are generated by maximizing the loss function with respect to the classifier's input x based on the gradients $\nabla_x L(x,y;\theta)$. The final adversarial perturbation is given by:

$$x_{adv} = x + \epsilon \,\text{sign}(\nabla_x L(x,y;\theta)). \tag{1}$$

To visualize the energy of the generated perturbations, we define the Signal to Perturbation Ratio (SPR) between the received signal and the perturbation as $E(x)/E(\eta)$ in dB. Note that the received signal already includes the interference due to channel effects between the transmitter and defender, and the defender's classifier is trained to classify these interfered signals into different categories. Such interference is raised by the wireless channel but has nothing to do with the adversarial perturbation. In other words, SPR is the energy ratio between the benign signal received by the defender and the perturbation added by the adversary. Next, we calculate the relationship between SPR and ϵ for the protocol and modulation classification datasets. As shown in Figure 4, the ratio drops faster with smaller ϵ and slows down when ϵ becomes larger. Note that the scale of ϵ differs from the two datasets corresponding to the overall magnitudes signals x in each dataset. For example, the amplitudes of I/Q samples for some signal types in the modulation classification dataset are much smaller than the magnitudes in the protocol classification dataset.

4. ADVERSARIAL VECTORS WITH VARYING ATTACK'S KNOWLEDGE

The strength of the adversarial attack (i.e., how much the defender's performance is reduced) depends on the adversary's knowledge about the defender and the information that the adversary has access to.[26] For example, existing AML attacks[13,14] require knowing either the model internals or its training data. These work craft adversarial examples using detailed DNN architecture and parameters. We release such conditions by learning a substitute model, giving different access for the adversary to the actual model. There is a wide range of knowledge that could be available. Therefore, we categorize the adversary's knowledge and availability to five levels described in this section.

4.1 White-box Attack

A white-box attack scenario refers to the situation where the adversary has access to the defender's classifier and datasets. Therefore, the loss function used in AML attacks can be formulated as: $L(x_d^{t,a}, y_d; \theta)$. A white-box attack scenario is the worst-case situation for the defender because the adversary has all the same information as the defender. However, the defender can protect some of their information, so the attacker must generate

perturbations based on imperfect knowledge. Therefore, we consider the different levels of knowledge for the attacker in both classifier and data and evaluate the classification accuracy accordingly.

4.2 Limited Knowledge of Classifier

In an imperfect knowledge situation, the attacker must learn a classifier $f_a(x;\theta_a)$ based on different knowledge levels of the defender's classifier $f_d(x;\theta_d)$. The loss can be represented by $L(x_d^{t,a}, y_a; \theta_a)$ because the label for the corresponding input has to be estimated by attacker's classifier. Therefore, the difference between $f_a(x;\theta_a)$ and $f_d(x;\theta_d)$ has a direct impact on the loss function. In this paper, we study incompleteness of the knowledge for the attacker and test their impacts on the generated perturbations as follows:

Attack (a): In this setting, the attacker has access to all the hyperparameters of the defender but not the seed used to train the neural network (e.g., the attacker knows the network structure, number of nodes at each layer, training routine, etc.). Therefore, although the hyperparameters are the same for the defender and adversary, the mapping function of the final trained classifiers will be different due to DNN weight initialization. Hence, the same network configuration with different random seeds will lead to different networks because of the non-covexity of the optimization task.

Attack (b): The attacker knows the overall DNN architecture but does not know the other hyperparameters. For example, an adversary knows the defender is using a CNN but not the hyperparameters (e.g., the filter number, kernel size, strides, etc.). However, such an adversarial knowledge setting can significantly impact the final weights of the DNN after the training. In this case, we consider all the defender's architecture and hyperparameters, except the filter number used in each layer, are known by the attacker.

Attack (c): In this setting, the attacker only knows the classifier type (e.g., CNN or RNN). Therefore, the adversary uses the classifier with different layers on the attacker side to generate the perturbations. We apply the same type of DNN but with varying layers (e.g., we use structure (a) for the defender but a two-layer RNN for the adversary).

Attack (d): In this scenario, the attacker does not know the classifier type. In our case, we use the CNN and RNN for signal classification. However, the mapping function f differs significantly with DNN types, especially since a CNN could represent features spatially, different from an RNN's sequential representation of features. Under this knowledge scenario, the CNN is used by the adversary to generate perturbations, but an RNN is used by the defender.

4.3 Limited Knowledge of Training Data

In the real environment, benign waveforms received by the attacker are $x_a^t = \mathbf{H}_{ta}x_o + n$, and signals received by the defender are $x_d^t = \mathbf{H}_{td}x_o + n$. The waveforms received by the attacker and the receiver are different considering the channel impact. As a result, the attacker needs to train its classifier f_a using x_a^t instead of x_d^t. Due to the training data being different from the defender's, the trained parameters for the attacker θ_a can vary a lot from the defender's θ_d even with the same hyperparameters. As a result, the loss function $L(x_d^{t,a}, y_d; \theta_d)$ is approximated by $L(x_a^t, y_a; \theta_a) + \eta^T \nabla_{x_a^t} L(x_a^t, y_a; \theta_a)$. We denote this type of adversarial signal as *Attack (e)*: The attacker has access to a different dataset than the defender to train its classifier. The transmitter broadcasts the signal, so the attacker and the defender receive the waveforms that contain the same bit-level information. Due to the channel impact, datasets are different in baseband format. Various random seeds are used to generate the interference caused by such an effect. We consider the AWGN channel between all the communication nodes and the same levels of SNR for the signals received by the defender and attacker.

5. ADVERSARIAL TRAINING TO IMPROVE ROBUSTNESS

In order to mitigate the accuracy drop caused by AML attacks, we propose to defend the classifier by adversarial training.[13] The primary objective is to increase model robustness by injecting adversarial examples into the training set. Adversarial training does not change the structure of the given classifier, because it only augments these perturbed data while training the targeted model. We suppose the attacker generates the adversarial signal

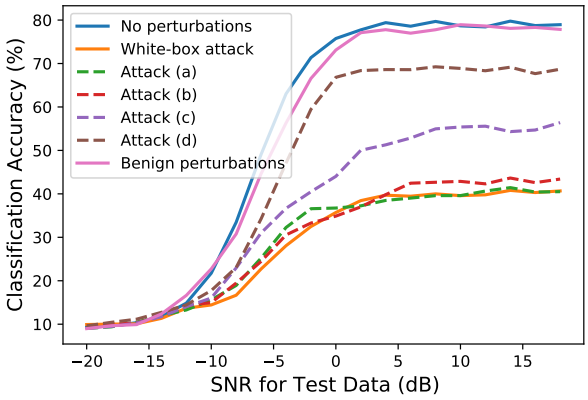

Figure 5. Classification accuracy vs. SNR of the test dataset. The VT-CNN2 classifier is used at the defender. The attacker's classifier is modified from VT-CNN2 to account for its limited knowledge ($\epsilon = 0.001$).

using FGSM. The augmentation can be done by feeding the model with both the original and crafted data, or learning with a modified objective function:

$$\tilde{L}(x, y; , \theta) = \alpha L(x, y; \theta) + (1 - \alpha) L(x + \eta, y; \theta). \tag{2}$$

where L is the cross entropy loss and $\alpha \in [0, 1]$. This form of data augmentation instead uses inputs that are unlikely to occur naturally but that expose flaws in how the model conceptualizes its decision function. The central idea behind this strategy is to increase the model's robustness by ensuring that it can predict the same class for legitimate and perturbed examples. We assume the attacker knows the neural network used by the defender with all parameters before the adversarial training (similar to white-box assumption, which is a strong attack). However, the defender retrains the same neural network architecture with benign and self-perturbed data (adversarial data) to improve the robustness. The attacker does not have access to the retrained neural network. The self-perturbed examples are generated from the original data set, so they have the same amount as the original ones. As a result, only half of the data are benign in the new training dataset. The attacker generates FGSM perturbations based on the neural network trained with only benign data using different ϵ. We then test and compare the classification accuracy of the original model and the adversarially trained model under these attacks.

6. SIMULATION RESULTS

For the modulation classification problem, we used RML.2016a dataset that has eleven modulation schemes, including 8PSK, BPSK, QPSK, QAM16, QAM64, CPFSK, GFSK, PAM4, WBFM, AM-DSB and AM-SSB, with SNR ranges from -18 dB to 20 dB with a step size of 2dB. For the wireless protocol classification problem, we use *Matlab Communication* and *5G Toolboxes* to generate waveforms of LTE, Wi-Fi, and 5G NR protocols with channel bandwidth of 20 MHz, and consider the baseband I/Q samples at the receiver (with added noise) as input to the classifier. By applying a sliding window, these samples are divided into multiple sequences, each consisting of 512 I/Q pairs. These sequences are used as datasets to train and test various classifiers. Approximately 15,000 of such segments were obtained, split into 70% for training and 30% for testing. In addition, we assume an AWGN channel for all transmission. The Wi-Fi waveform is transmitted by generating baseband samples of 802.11ac (VHT) with BPSK modulation and 1/2 code rate. LTE waveforms are generated assuming downlink reference measurement channel with R.9. This waveform uses 64 QAM modulation. We also generate 5G waveforms using 5G downlink fixed reference channel under QPSK modulation and a code rate of 1/3, with a subcarrier spacing of 15 kHz.

We studied FGSM perturbations under limited-knowledge attacks for the modulation classification dataset. To keep the energy of the perturbation low, we explore the FGSM attacks with $\epsilon = 0.001$ as an example. As

Table 1. Protocol Classification Accuracy for Different Attacks and ϵ

ϵ	W-B attack	Attack (a)	Attack (b)	Attack (c)	Attack (d)	Attack (e)	Benign Perturbs
0.05	62.03%	63.47%	84.24%	82.72%	83.83%	85.98%	91.73%
0.15	28.74%	31.85%	49.83%	48.49%	58.66%	46.97%	91.38%
0.25	10.79%	16.02%	21.36%	19.79%	42.31%	33.22%	90.51%

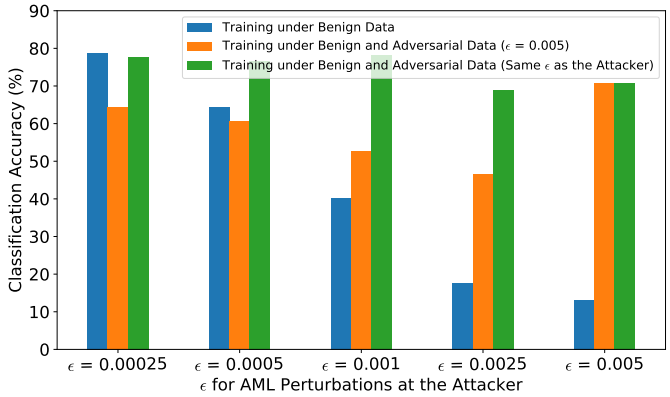

Figure 6. Classification accuracy for different perturbed data under adversarial training for RML 2016.a dataset with SNR = 16 dB (VT-CNN2 classifier is used at both the attacker and defender).

shown in Figure 5, attack (a) and (b) shows very close accuracy with the white-box attack. This result suggests that a small structure change may not heavily impact the FGSM adversarial signals for VT-CNN2 on the RML 2016.10a dataset. However, when the knowledge of the attacker is further reduced, the impact of FGSM becomes weaker (shown as the attack (c) and (d)).

We also investigate the protocol classification problem. The accuracy for structure (a) under different attacks is summarized in Table 1. The impact of attack (a) is the closest to the white-box attack, and it is because the attacker has the same hyperparameters as the defender. One can still inherit most of the properties from the other, even with different training seeds. The DNN used by the attacker exchanges the filter number of the first two layers for attack (b) and has fewer layers (i.e., we remove the third layer of structure (a)) for attack (c). They have similar performance on the defender, which indicates that these hyperparameters have comparable influences. Attack (d) has the worst attack impact among the imperfect knowledge attacks. The attacker in this mode applies the CNN structure (b) to generate the adversarial signals for the RNN structure (a). Even though both types of the classifier can achieve high classification accuracy on the received waveforms, the actual mapping function can differ significantly from each other. As a result, a well-crafted perturbation by the CNN may not achieve the expected effect on the RNN. Attack (e) uses the different training datasets to generate the AML perturbations. Thus, it shows more variance than other attacks.

For the defense mechanisms, we apply adversarial training technique to the modulation classification dataset. We first generate the self-perturbed data with $\epsilon = 0.005$ because we care about the small perturbations, and the largest ϵ considered in this dataset is 0.005. Figure 6 shows the classification accuracy for training under different datasets. The orange bars indicate that the classifier has a higher accuracy under attacks when trained with self-perturbed data. Even though we only consider the adversarial examples with $\epsilon = 0.005$, the accuracy for $\epsilon = 0.0025$ and $\epsilon = 0.001$ is also improved. Such improvement is because the ϵ in the FGSM attack only impacts the energy of the perturbation. In other words, the perturbation vectors generated under $\epsilon = 0.0025$ and $\epsilon = 0.005$ point to the same direction but with different scales. The adversarial examples with high ϵ helps the neural network know the direction of the perturbations. It can help improve the accuracy of lower ϵ examples with the same perturbation direction. However, adversarial training is not always helpful. The classification accuracy for small ϵ (e.g., less than 0.0005) is reduced when including the adversarial examples into the training. This may be because the self-perturbed data poison the original dataset, which makes the neural network unable to learn the precise mapping relationship between the input and the label during the training. The green bars stand for the result for adversarial training when the defender knows the ϵ that the attacker uses. In this case,

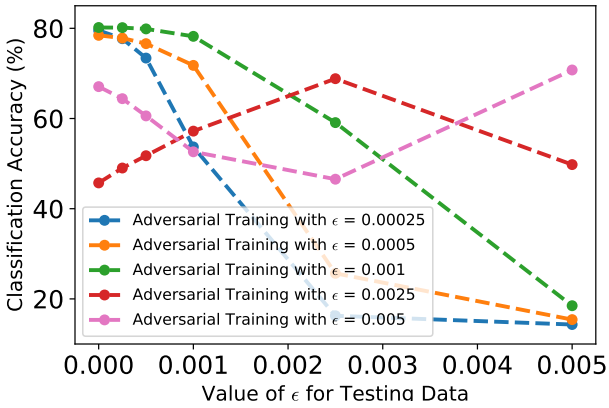

Figure 7. Classification accuracy for different perturbed data under adversarial training using the RML 2016.a dataset (SNR = 16 dB, VT-CNN2 classifier is used for both the attacker and defender).

the defender can generate the self-perturbed data with the same ϵ used by the attacker, so they have better performance when tested under the attack. It indicates that the choice of the ϵ to generate the adversarial data used by the defender could impact the robustness of the classifier. In Figure 7, we further corroborate the above observations by studying the classification accuracy versus ϵ of the test data, considering various (fixed) values of ϵ of the defender's training set. This figure is essentially a generalization of the results presented by the orange bar of Figure 6. The defender's classifier performs the best when it is trained with the same ϵ used by the attacker.

7. CONCLUSIONS

In this paper, we studied various CNN- and RNN-based classifiers for protocol and modulation identification based on received I/Q samples (without signal decoding). These classifiers typically exhibit high classification accuracy under random noise. However, under FGSM perturbations (a type of AML-based noise), we observed that such classifiers become highly inaccurate even when the attacker has limited knowledge of the defender's classification engine. Compared with traditional jamming, where the attacker transmits only AWGN noise, the proposed AML-based attacks require significantly less transmission power to mislead the defender's classifier. To improve the robustness of the studied DNN-based classifiers, we proposed the use of adversarial training for the defender's classifier. However, data injection increases the training cost, for it requires more samples for the training process. In addition, it may not be as effective under other AML attacks (beyond FGSM). Our future work includes adversarial defenses for baseband signals that do not involve adversarial training since the generation of these samples causes a significant overhead during training.

ACKNOWLEDGMENTS

This research was supported by the U.S. Army Small Business Innovation Research Program Office and the Army Research Office under Contract No. W911NF-21-C-0016, by NSF (grants CNS-1563655, CNS-1731164, IIP-1822071, and CAREER Award #1943552), and by the Broadband Wireless Access & Applications Center (BWAC). Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of NSF or ARO.

REFERENCES

[1] Xin, C. and Song, M., "Analysis of the on-demand spectrum access architecture for CBRS cognitive radio networks," *IEEE Transactions on Wireless Communications* **19**(2), 970–978 (2020).

[2] Lees, W. M., Wunderlich, A., Jeavons, P. J., Hale, P. D., and Souryal, M. R., "Deep learning classification of 3.5-GHz band spectrograms with applications to spectrum sensing," *IEEE Transactions on Cognitive Communications and Networking* **5**(2), 224–236 (2019).

[3] Hirzallah, M., Krunz, M., Kecicioglu, B., and Hamzeh, B., "5G New Radio Unlicensed: Challenges and evaluation," *IEEE Transactions on Cognitive Communications and Networking* **7**, 689–701 (Sept. 2021).

[4] Hirzallah, M., Krunz, M., and Xiao, Y., "Harmonious cross-technology coexistence with heterogeneous traffic in unlicensed bands: Analysis and approximations," *IEEE Transactions on Cognitive Communications and Networking* **5**(3), 690–701 (2019).

[5] Naik, G., Liu, J., and Park, J., "Coexistence of wireless technologies in the 5 GHz bands: A survey of existing solutions and a roadmap for future research," *IEEE Communications Surveys & Tutorials* **20**(3), 1777–1798 (2018).

[6] Hirzallah, M., Afifi, W., and Krunz, M., "Full-duplex-based rate/mode adaptation strategies for Wi-Fi/LTE-U coexistence: A POMDP approach," *IEEE Journal on Selected Areas in Communications* **35**(1), 20–29 (2017).

[7] Qin, Z., Ye, H., Li, G. Y., and Juang, B.-H. F., "Deep learning in physical layer communications," *IEEE Wireless Communications* **26**(2), 93–99 (2019).

[8] Sankhe, K., Belgiovine, M., Zhou, F., Riyaz, S., Ioannidis, S., and Chowdhury, K., "ORACLE: Optimized radio classification through convolutional neural networks," in [*Proc. of the IEEE Conference on Computer Communications (INFOCOM)*], 370–378 (2019).

[9] Shi, Y., Davaslioglu, K., Sagduyu, Y., Headley, W., Fowler, M., and Green, G., "Deep learning for RF signal classification in unknown and dynamic spectrum environments," in [*Proc. of the IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*], (2019).

[10] Shang, F., Wang, B., Li, T., Tian, J., Cao, K., and Guo, R., "Adversarial examples on deep-learning-based ADS-B spoofing detection," *IEEE Wireless Communications Letters* **9**(10), 1734–1737 (2020).

[11] Zhang, W., Feng, M., Krunz, M., and Abyaneh, A. H. Y., "Signal detection and classification in shared spectrum: A deep learning approach," in [*Proc. of the IEEE Conference on Computer Communications (INFOCOM)*], 1–10 (2021).

[12] Zhang, W. and Krunz, M., "Machine learning based protocol classification in unlicensed 5 GHz bands," in [*Prof. of IEEE International Conference on Communications Workshops (ICC Workshops)*], (2022).

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

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

[15] Moosavi-Dezfooli, S.-M., Fawzi, A., and Frossard, P., "Deepfool: a simple and accurate method to fool deep neural networks," in [*Proceedings of the IEEE conference on computer vision and pattern recognition*], 2574–2582 (2016).

[16] Liu, H. and Ditzler, G., "Adversarial audio attacks that evade temporal dependency," in [*2020 IEEE Symposium Series on Computational Intelligence (SSCI)*], 639–646 (2020).

[17] Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., and Song, D., "Robust physical-world attacks on deep learning visual classification," in [*Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], 1625–1634 (2018).

[18] Sadeghi, M. and Larsson, E., "Physical adversarial attacks against end-to-end autoencoder communication systems," *Communications Letters IEEE* **23**(5), 847–850 (2019).

[19] Kim, B., Sagduyu, Y., Davaslioglu, K., Erpek, T., and Ulukus, S., "Over-the-air adversarial attacks on deep learning based modulation classifier over wireless channels," in [*Proc. of the Annual Conference Information Sciences and Systems*], (2020).

[20] Lin, Y., Zhao, H., Tu, Y., Mao, S., and Dou, Z., "Threats of adversarial attacks in DNN-based modulation recognition," in [*Proc. of the IEEE Conference on Computer Communications (INFOCOM)*], 2469–2478 (2020).

[21] Zhang, W., Krunz, M., and Ditzler, G., "Intelligent jamming of deep neural network based signal classification for shared spectrum," in [*Prof. of the IEEE Military Communications Conference (MILCOM)*], 987–992 (2021).

[22] Zhang, L., Lin, C., Yan, W., Ling, Q., and Wang, Y., "Real-time ofdm signal modulation classification based on deep learning and software-defined radio," *IEEE Communications Letters* **25**(9), 2988–2992 (2021).

[23] Vanhoy, G., Thurston, N., Burger, A., Breckenridge, J., and Bose, T., "Hierarchical modulation classification using deep learning," in [*Proc. of the IEEE Military Communications Conference (MILCOM)*], 20–25 (2018).

[24] O'Shea, T. J., Roy, T., and Clancy, T. C., "Over-the-air deep learning based radio signal classification," *IEEE Journal of Selected Topics in Signal Processing* **12**(1), 168–179 (2018).

[25] Schwartz, D. and Ditzler, G., "An algorithm for adversarial training to improve neural network robustness," in [*IEEE/INNS International Joint Conference on Neural Networks*], (2021).

[26] Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A., "Practical black-box attacks against machine learning," in [*Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*], 506–519 (2017).