Adversarial Attacks and Defense Methods for Power Quality Recognition

Jiwei Tian, Buhong Wang, Member, IEEE, Jing Li, Zhen Wang and Mete Ozay

Abstract

Vulnerability of various machine learning methods to adversarial examples has been recently explored in the literature. Power systems which use these vulnerable methods face a huge threat against adversarial examples. To this end, we first propose a signal-specific method and a universal signal-agnostic method to attack power systems using generated adversarial examples. Black-box attacks based on transferable characteristics and the above two methods are also proposed and evaluated. We then adopt adversarial training to defend systems against adversarial attacks. Experimental analyses demonstrate that our signal-specific attack method provides less perturbation compared to the FGSM (Fast Gradient Sign Method), and our signal-agnostic attack method can generate perturbations fooling most natural signals with high probability. What’s more, the attack method based on the universal signal-agnostic algorithm has a higher transfer rate of black-box attacks than the attack method based on the signal-specific algorithm. In addition, the results show that the proposed adversarial training improves robustness of power systems to adversarial examples.

Index Terms

Adversarial example, adversarial training, universal perturbation.

I. INTRODUCTION

POWER quality refers to a variety of electromagnetic phenomena that characterize voltage and current measured at a given time instance and location in a power system [2]. Disturbance of power quality (PQ) signals can cause severe problems in electrical grids [3]. PQ disturbances occurring in grids may cause financial loss, damage and faulty operation of equipment installed in grids and end-user devices [4]. In end-user systems, PQ disturbances can cause loss of data, memory failures of sensitive loads (such as computers, protection and relay devices), and unstable operation of controls [5]. Therefore, if PQ disturbances are not properly recognized and mitigated by monitoring systems, then power grid assets can be harmed resulting in high costs and safety hazards in extreme but realistic cases. Numerous techniques have been proposed to address the aforementioned problems [4], [6]. Among them, deep neural networks (DNNs) have been employed to improve accuracy for recognition of PQ disturbances replacing hand-crafted features by learned feature representations [7].

Although DNNs show outstanding performance in various tasks, they are vulnerable to a form of attacks called “adversarial example” [8]. Adversarial examples with different attributes have been explored for image, text, and speech processing in [8]–[10]. DNNs have been recently used in cyber physical systems (CPSs) [11], but exploration and assessment of their security is still an open problem. Security risks caused by adversarial machine learning (ML) methods in CPSs have been discussed in [12]. Vulnerabilities of ML algorithms utilized in power systems were first shown in [13]. Additional works have addressed security problems for grid event classification [14], N-1 security [15], power quality signal classification [13] and detection of false data injection attacks [16], [17]. The vulnerabilities of load forecasting algorithms caused by perturbations injected into input temperatures were studied in [18]. A domain-specific framework was proposed to evaluate security and resilience of load forecasting algorithms in [19] and experimental results

An early version of this work was presented in part at the 2019 IEEE SmartGridComm [1]. This work was supported by the National Natural Science Foundation of China under Grant 61902426 and the National University of Defense Technology Research Fund under Grant ZK21-44. Jiwei Tian is with Air Traffic Control and Navigation College, Air Force Engineering University, Xi’an, China (e-mail: tianjiwei2016@163.com). B. Wang and Z. Wang are with Information and Navigation College, Air Force Engineering University, Xi’an, China. J. Li is with Henan University of Technology, Zhengzhou, China.
showed that DNN-based load predictions may suffer from worst-case attacks even with a partial control of network. From a defense perspective, adversarial training [20] and game theory [21] are respectively adopted to design resilient load forecasting systems. Adversarial attacks for DNN-based state estimation were studied in [22].

Characterization of these attacks and defense mechanisms is an open problem that has not been fully explored and understood. An adversarial signal attack is a vital threat, since signals generated by the attack are perceptually similar to the original signals. Therefore, a human operator of the system may not be aware of the presence of the attack. Then, the system may take a wrong action according to the incorrectly predicted signal class affecting the physical system. Vulnerability of DNNs against particular adversarial attacks in power systems was analyzed in [1], [13]. Among them, a variant of FGSM attack [23] is proposed to generate adversarial examples for PQ classification in [13]. A more accurate and computationally efficient attack method and corresponding defense method are proposed in [1]. However, stronger and more practical attack methods (such as universal perturbations and black-box attacks) and corresponding defense strategies need to be explored more comprehensively, which is lacking in the current literature.

In order to defend power systems against these attacks and improve their robustness, learning models deployed in the systems should be carefully analyzed. In this paper, we explore adversarial attacks to CNN (Convolutional Neural Network) algorithms and examine the corresponding defense methods for PQ recognition tasks. The main contributions of this work are summarized as follows:

1. We propose a signal-specific algorithm which outperforms a state-of-the-art method [13] for generating adversarial perturbation of signals in power systems.
2. We analyze existence of universal signal-agnostic perturbations generated by CNNs in power systems, and propose an algorithm to recognize such perturbations. Our results show that generated universal perturbations make models discriminate natural power signals into several statistically dominant signal categories.
3. We explore and analyze black-box attacks: the attacker can train a local substitute model and generate adversarial samples based on the substitute model, and then use its transferable characteristics to attack the target model. The experimental results show that the attack method based on the universal signal-agnostic algorithm has a higher transfer rate of black-box attacks than the attack method based on the signal-specific algorithm, which is a huge and practical threat for CNN based PQ classification.
4. We propose using adversarial training as a defense method to improve robustness of learning models employed in power systems to adversarial attacks.
5. We quantitatively analyze misclassification of different attacks, and summarize the characteristics of different attack methods for PQ signals. We suggest an approach to qualitatively analyze the relationship between the characteristics of adversarial attacks to PQ signals and defense methods in the Smart Grid. For this purpose, we visualize signal measurements and their feature representations learned at different layers of CNNs using the t-SNE algorithm [24].

The paper is organized as follows. In Section II, we provide an overview of adversarial methods and describe our proposed attack algorithms. In Section III, our proposed adversarial training method is introduced. Experimental analyses of the proposed methods are given for various case studies in Section IV. Conclusion and discussion are provided in Section V.

II. ADVERSARIAL ATTACKS FOR POWER QUALITY CLASSIFICATION USING CNNs

A. Attack Models and Objectives

As shown in Fig. [1], a well-trained CNN can be used for power quality assessment in the Smart Grid, which is crucial for its robust operation. However, an attacker can penetrate a communication network of

1The code used for reproducing the experimental analyses and results are provided on https://github.com/JiweiTian/AT_SG.git
2We consider power signals commonly measured by a particular system operator of a power system as natural power signals.
the system, and use adversarial signals instead of original signals to attack the system to fool the CNN for misclassification of the PQ signals (for example, Man-In-The-Middle attack). Note that we consider attacks in the testing phase (adversarial examples and adversarial attacks) rather than in the training phase (poisoning attacks). The objective of the attacker is generating imperceptible adversarial signals similar to the original signals, resulting in their misclassification by algorithms without being noticed by human operators.

In our white-box attack model, we assume that attackers can access parameters of CNNs deployed in power systems and use this knowledge to construct adversarial signals. In practice, these parameters can be obtained by insider threats or using various attack methods, such as eavesdropping on network traffic or database intrusion. Rouhani et al. [25] suggest that white-box attacks represent the most powerful attacker that can appear in real-world settings. Although these assumptions render a strong attacker that may not always represent the practical cases, they enable us to evaluate the robustness and vulnerabilities under the worst-case scenario, providing an upper bound on the impact of proposed attacks on PQ CNN models.

Although many adversarial attack methods have been proposed in image domain, there are significant differences between PQ recognition tasks using power signals and object recognition tasks using two or three dimensional images:

- **Data Characteristics:** A PQ signal is a time-series, whose characteristics are different from that of two or three dimensional images. Therefore, in order to produce imperceptible PQ perturbations, the characteristics of time-series data should be considered to avoid occurrence of abnormal values with great changes. Besides, although image pixels take values from a fixed interval, such as [0,1] or [0,255], PQ signals have no definite range.

- **Levels of Perturbation:** In image processing and computer vision tasks, norms such as \(\ell_0\), \(\ell_2\) and \(\ell_\infty\) are used to model adversarial perturbations. Since PQ signals have no definite range, some adversarial attack methods based on box constraints [26] and “change of variables” such as C&W [27] are not suitable for PQ signals. Although \(\ell_0\) norm-based perturbations such as Sparsefool [28] and “One pixel attack” [29] in image domain will not affect human perception, the outliers generated by the \(\ell_0\) norm-based perturbations will have a greater impact on time-series PQ signals. Besides, for PQ signals, the perturbations should be small enough to ensure that the attacker’s attack is imperceptible to human operators, i.e., the added perturbation should belong to the same type as the original signal from the human operator’s point of view (Class 1). However, the target CNN will regard it as a new type (for example, Class 2).

Fig. 1: Adversarial attacks on power signal recognition algorithms using deep learning methods in the Smart Grid.
signals, it’s hard to determine a maximum budget for $\ell_\infty$ norm-based perturbations (the maximum budget will affect human perception of PQ signals, and each value is allowed to be changed by up to this limit, with no limit on the number of input data that are modified). Therefore, we consider the $\ell_2$ norm case for PQ signals in the paper.

Although FGSM was employed in particular power systems\cite{13}, development of a general and suitable approach for adapting adversarial attacks for PQ recognition in power systems is an open problem. We address this problem by proposing two methods called signal-specific and signal-agnostic attacks. We realize these approaches to generate adversarial signals for PQ recognition by adapting DeepFool \cite{30} and universal attack methods in power systems. Although other adversarial attack methods have been proposed in image domain, many methods and their variants are not suitable for the PQ signal field for various reasons, such as the existence of box constraints (L-BFGS \cite{26}, C&W \cite{27}) and artificially determined attack parameters (FGSM \cite{23}, I-FGSM \cite{31}, JSMA \cite{32}). We consider that our proposed approaches can be employed to adapt a wider class of attack models for various tasks in power systems by researchers, especially attack methods based on geometrical perspective \cite{33} (such as targeted DeepFool \cite{34} \cite{35}, DeepFool-5 \cite{36} and so on).

### B. Signal-Specific Adversarial Signals

In this section, we introduce our signal-specific adversarial attack method which is proposed to generate adversarial perturbations of signals in power systems.

Suppose that $f: \mathbb{R}^n \rightarrow \mathbb{R}^K$ is a classifier computed by a learning model to determine the class label of a signal $x_0 \in \mathbb{R}^n$ among $K$ power quality classes. In this work, deep learning algorithms are used to compute feature representations of PQ signals and classifiers $f$. Therefore, we will identify the classifier by a non-linear function $f(x_0: \mathcal{W})$ which is parameterized by a set of tensors $\mathcal{W}$ denoting weights and bias vectors called model parameters of CNNs. The parameters $\mathcal{W}$ are optimized using a stochastic optimization algorithm such as stochastic gradient descent. Since the target model in this paper is fixed, for the sake of simplicity, we don’t show the dependence of $f$ on $\mathcal{W}$. The class label of the power signal $x_0$ is estimated by

$$\hat{y}(x_0) = \arg \max_k f_k(x_0),$$

where $f_k(x_0)$ is the output of $f(x_0)$ for the $k^{th}$ class. An adversarial perturbation is defined as the minimal (imperceptible) perturbation $\mathbf{r}$ that is sufficient to change the estimated power quality label $\hat{y}(x_0)$ at $x_0$ by

$$\Delta(x_0; \hat{y}(x_0)) \triangleq \min_{\mathbf{r}} \|\mathbf{r}\|_2,$$

subject to $\hat{y}(x_0 + \mathbf{r}) \neq \hat{y}(x_0)$,

where $\| \cdot \|_2$ denotes the $\ell_2$ norm.

In adversarial attacks, attackers can make learned models incorrectly estimate class labels of attacked power signals by adding the minimal imperceptible perturbation computed by (2). For instance, normal power signals can be misclassified as impulse signals, and vice versa. Thereby, adversarial attacks can affect power quality monitoring systems by adversarial data injection, and cause failure of their operation. The minimal perturbation computed using (2) for different classes and samples of PQ signals may be different from each other. To generate minimum perturbation of the signal $x_0$ optimizing the closest decision boundary of $f$ from $x_0$, a signal-specific adversarial attack method (SSA) is proposed in Algorithm 1.

CNNs partition high dimensional class regions with nonlinear decision boundaries by locally piecewise linear functions implemented in network nodes into subregions \cite{37} which are convex polytopes \cite{38}. Wong and Zolter \cite{39} propose that if CNNs are optimized to minimize the worst case loss over convex outer approximation of the set of activations reachable through a norm-bounded perturbation (called

\footnote{Therefore, we consider the FGSM method as the baseline and state-of-the-art for adversarial attacks in PQ recognition in our experimental analyses.}
adversarial polytope), then the trained CNN models is provably robust to any norm-bounded adversarial
attack. We employ this result to generate adversarial signals according to a convex outer approximation
of an adversarial polytope of trained CNN models. For this purpose, we first define the region (polytope)
of the space of adversarial signals by

$$ P = \bigcap_{k=1}^{K} \{ x : f_\hat{y}(x_0)(x) \geq f_k(x) \}.$$  \hspace{1cm}(3) $$

Activations of the model for the signals $x$ belonging to this region for each different $k^{th}$ class are bounded
by that for the signals $x_0$ for the predicted class $\hat{y}(x_0)$. Next, we generate adversarial signals in Algorithm
1 as follows:

- **Step 4 and 5:** The region $P$ is approximated at each $i^{th}$ epoch by
  $$ \hat{P}_i = \bigcap_{k=1}^{K} \{ x : f'_k(x_i) + (w'_k)^T x \leq 0 \},$$  \hspace{1cm}(4) $$
  \text{where } w'_k = \nabla f_k(x_i)^T - \nabla f_{\hat{y}(x_0)}(x_i)^T \text{ (at step 4) and } f'_k = f_k(x_i) - f_{\hat{y}(x_0)}(x_i) \text{ (at step 5).}$$

- **Step 7:** We compute a convex outer approximation (boundary) of the polytope and generate signals
  far from this boundary. Therefore, first the closest hyperplane $\hat{l}$ of the boundary of $P$ is computed
  where $|\cdot|$ denotes the absolute value.

- **Step 8:** Next, the minimum perturbation $r_*(x_0)$ projecting $x_0$ on $\hat{l}$ is approximated at each $i^{th}$ epoch
  by $r_i$.

- **Step 9 and 10:** Then, the adversarial signal $x_i$ and epoch variable $i$ are updated at the step 9 and
  10, respectively.

- **The steps 3-10** are iterated while estimated class labels of the normal $x_0$ and the adversarial signal
  $x_i$ are same.

- **Step 12:** Finally, the algorithm injects the sum of perturbations $r_j, j = 1, 2, \ldots, i$ to the original
  signal to compute the adversarial PQ signal.

C. Signal-Agnostic Adversarial Signals

In our proposed signal-agnostic adversarial attack method (SAA), attackers aim to generate signal-
agnostic and very small perturbation vectors that cause normal PQ signals to be misclassified with high
probability. An attacker can make the PQ signal be misclassified with high probability by adding signal-
agnostic perturbations without spending time calculating the specific perturbation, independent of class of
actual PQ signal (e.g., normal, sag and so on) fed to the system. Presence of such perturbations poses a
great threat to learning models in power systems, as the adversary may simply add the same pre-computed
signal-agnostic perturbation to the new signal and cause misclassification. Therefore, pre-computed signal-
agnostic perturbations are called universal perturbations [40].

The goal of the SSA is to generate a quasi-imperceptible universal perturbation signal vector $v$ such that
most PQ signals sampled from a distribution $\mu$ are misclassified when applying the universal perturbation
signal. We compute the vector $v$ considering the following two constraints:

1) The magnitude of the universal perturbation vector $v$ is controlled by a parameter $\xi > 0$ using
   $$ \|v\|_2 \leq \xi.$$ 

2) The misclassification probability is lower bounded for all signals $x$ sampled from the distribution $\mu$
   by
   $$ P_{x \sim \mu}(\hat{y}(x + v) \neq \hat{y}(x)) \geq 1 - \delta, \text{where } \delta \in (0, 1).$$  \hspace{1cm}(5) $$
Algorithm 1 Generating Signal-Specific Adversarial Signals.

**Input:** \( \mathbf{x} \): An input power quality signal,
\( f \): A learned signal classification model.

**Output:** Adversarial power quality signal \( \hat{\mathbf{x}} \).

1. Initialization: \( i \leftarrow 0 \) and \( \mathbf{x}_0 \leftarrow \mathbf{x} \).
2. while \( \hat{y}(\mathbf{x}_i) = \hat{y}(\mathbf{x}_0) \) do
   3. for \( k \neq \hat{y}(\mathbf{x}_0) \) do
      4. \( \mathbf{w}'_k \leftarrow \nabla f_k(\mathbf{x}_i) - \nabla f_{\hat{y}(\mathbf{x}_0)}(\mathbf{x}_i) \)
      5. \( f'_k \leftarrow f_k(\mathbf{x}_i) - f_{\hat{y}(\mathbf{x}_0)}(\mathbf{x}_i) \)
   6. end for
7. \( \hat{l} \leftarrow \arg \min_{k \neq \hat{y}(\mathbf{x}_0)} |f'_k|/\|\mathbf{w}'_k\|_2 \)
8. \( r_i \leftarrow \|\mathbf{w}'_{\hat{l}}\|_2 \mathbf{w}'_{\hat{l}} \)
9. \( \mathbf{x}_{i+1} \leftarrow \mathbf{x}_i + r_i \)
10. \( i \leftarrow i + 1 \)
11. end while
12. return Adversarial power quality signal \( \mathbf{x} + \sum_{j=1}^{i} r_j \).

Let \( \mathcal{X} = \{\mathbf{x}_i\}_{i=1}^m \) be a set of PQ signals sampled from the distribution \( \mu \). Our method aims at finding a perturbation \( \mathbf{v} \) fooling signals satisfying \( \|\mathbf{v}\|_2 \leq \xi \). Our proposed algorithm (Algorithm 2) gradually crafts adversarial perturbations on signals belonging to \( \mathcal{X} \). At each iteration, the minimal perturbation \( \Delta \mathbf{v}_i \) pushing the sample \( \mathbf{x}_i \) to the decision boundary is calculated and added to the current perturbation. More precisely, if the current perturbation \( \mathbf{v} \) does not fool the sample signal \( \mathbf{x}_i \), then we will compute an additional perturbation \( \Delta \mathbf{v}_i \) with minimal norm to fool the sample signal \( \mathbf{x}_i \) at step 5 of the Algorithm 2 by solving the following optimization problem
\[
\Delta \mathbf{v}_i = \arg \min_{\mathbf{r}} \|\mathbf{r}\|_2
\]
subject to \( \hat{y}(\mathbf{x}_i + \mathbf{v} + \mathbf{r}) \neq \hat{y}(\mathbf{x}_i) \). \hfill (6)

In order to meet the first constraint, the universal perturbation is projected on an \( \ell_2 \) ball of radius \( \xi \) and centered at 0 at step 6 of the Algorithm 2 by
\[
\Pi_{2,\xi}(\mathbf{v} + \Delta \mathbf{v}_i) = \arg \min_{\mathbf{v}'} \|\mathbf{v} + \Delta \mathbf{v}_i - \mathbf{v}'\|_2
\]
subject to \( \|\mathbf{v}'\|_2 \leq \xi \). \hfill (7)

The fooling rate is defined for the perturbed training set \( \mathcal{X} \) by
\[
R(\mathcal{X}, \mathbf{v}) := \frac{1}{\sigma(\mathcal{X})} \sum_{\mathbf{x} \in \mathcal{X}} \mathbb{I}(\hat{y}(\mathbf{x}_i + \mathbf{v}) \neq \hat{y}(\mathbf{x}_i)), \hfill (8)
\]
where \( \mathbb{I}(\cdot) \) is an indicator function and \( \sigma(\mathcal{X}) \) denotes the size of the perturbed training set \( \mathcal{X} \). If the rate \( R(\mathcal{X}, \mathbf{v}) \) exceeds \( 1 - \delta \), then the algorithm terminates. This method does not ensure that we can find universal perturbations with large misclassification rates. However, in our experiments, we can obtain universal perturbations with relatively high misclassification rates using this method with limited signals.

D. Black-box Attacks Against CNN Based Power Quality Recognition

Without knowing the target model \( f \), the attacker can train his own substitute model \( f' \) and craft adversarial examples against \( f' \). Due to the transferability of adversarial examples \cite{41, 42}, this kind of
Algorithm 2 Generating Signal-Agnostic Adversarial Signals.

**Input:** \( \mathcal{X} \): A set of power quality signals.
- \( \xi \): An upper bound of the \( \ell_2 \) norm of the perturbation.
- \( \delta \): A parameter used to control the fooling rate.
- \( f \): A learned signal classification model.

**Output:** A signal-agnostic perturbation vector \( v \).

1: Initialization: \( v \leftarrow 0 \)
2: while \( R(\mathcal{X}, v) \leq 1 - \delta \) do
3: for each signal \( x_i \in \mathcal{X} \) do
4: if \( \hat{y}(x + v) = \hat{y}(x) \) then
5: Compute the minimal perturbation \( \Delta v_i \) that pushes \( x_i + v \) to the decision boundary by solving (6).
6: Update the perturbation by \( v \leftarrow \Pi_{\ell_2,\xi}(v + \Delta v_i) \).
7: end if
8: end for
9: end while
10: return Signal-agnostic perturbation vector \( v \).

| Method | \( \hat{\rho}_{\text{adv}} \) | Misclassification Rate | Average Time (sec.) |
|--------|------------------|----------------------|---------------------|
| FGSM   | \( 770 \times 10^{-3} \) | 0.88                 | \( 1.2 \times 10^{-2} \) |
| SSA    | \( 5.0 \times 10^{-3} \) | 0.98                 | \( 2.2 \times 10^{-1} \) |
| SAA    | \( 26.0 \times 10^{-3} \) | 0.74                 | \( 1.6 \times 10^{-6} \) |

black-box attacks also has a certain success rate. If the adversarial example generated against the model \( f' \) can also mislead the model \( f \), we call this black-box attack successful.

Given the well-trained substitute model \( f' \), the attacker can use various attack methods to generate adversarial examples. In other words, the attacker can carry out white-box attacks based on the substitute model \( f' \) to generate corresponding adversarial examples, then use them to implement black-box attacks against the target model \( f \). In the following experiments, we will explore and analyze black-box attacks based on the substitute model \( f' \) and the above proposed algorithms (Algorithm 1 and Algorithm 2).

### III. Defending Learning Models using Adversarial Training

Adversarial training is used to improve robustness of a learning model to adversarial attacks. In this approach, a model is trained using a large number of adversarial examples generated by different attack methods. The basic requirement of this approach is employment of the strongest possible attack to generate adversarial examples using (1) and (2). Adversarial training improves robustness of DNNs to adversarial attacks [23], [43], provide regularization [23] for DNNs and improve their classification accuracy [44]. In this paper, we adopt adversarial training as our defense method to improve robustness of PQ classification models in power systems. Our proposed adversarial training method is summarized in Algorithm 3.

### IV. Experimental Analyses

#### A. Experimental Setup

We evaluate accuracy of the proposed attack and defense methods for power quality assessment by classifying signals using a convolutional neural network (CNN) [7]. Based on the PQ models in [7], the sampling frequency of signals is set as 3200Hz, the fundamental frequency of signals is set as 50Hz, the
Algorithm 3 Defending Models using Adversarial Training.

**Input:** $D_{tr}$: A training dataset of clean samples.
$N_{iter}$: The number of iterations used for training.

**Output:** A robust power quality classification model $f_\Theta$, where $\Theta$ is a set of learnable parameters of the model $f$.

**Training the model $f_\Theta$ using a training dataset $D_{tr}$:**
1: for $1, 2, \ldots, N_{iter}$ do
2: Optimize $\Theta$ by training $f_\Theta$ using $D_{tr}$.
3: end for
4: Generate a set of adversarial signals $\tilde{X}$ based on normal signals $X$ using Algorithm 1.

**Retraining the model $f_\Theta$ using the set of signals $\tilde{X}$:**
5: for $1, 2, \ldots, N_{iter}$ do
6: Update $\Theta$ by training $f_\Theta$ using $\tilde{X}$ and $X$.
7: end for
8: return A robust power quality classification model $f_\Theta$.

number of total cycles (periods) of the fundamental frequency is set as 10, and the amplitude (per unit) is set as 1. Therefore, the dimension of input signal vectors is 640. The network architecture is given in Fig. 2. In the PQ assessment task, a CNN model was trained to classify signals into 17 categories. Signal types and corresponding representative symbols are shown in Table II. We use the mathematical model of PQ disturbances and parameters proposed in [3]. A labeled dataset of 255000 signals at 30dB of SNR (Signal-Noise-Ratio) was constructed using 15000 signals belonging to each class. The dataset was randomly shuffled, 1/4 of the shuffled data was split as the testing set, and the rest was used for training.

| C-1 | Normal  | C-7 | Harmonics       | C-13 | Sag with Oscillatory transient |
|-----|---------|-----|-----------------|------|--------------------------------|
| C-2 | Sag     | C-8 | Harmonics with Sag | C-14 | Swell with Oscillatory transient |
| C-3 | Swell   | C-9 | Harmonics with Swell | C-15 | Sag with Harmonics               |
| C-4 | Interruption | C-10 | Flicker          | C-16 | Swell with Harmonics             |
| C-5 | Transient/Impulse/Spike | C-11 | Flicker with Sag | C-17 | Notch                           |
| C-6 | Oscillatory transient | C-12 | Flicker with Swell |

In order to analyze accuracy of a learning model using clean signals, we randomly shuffled the dataset, and trained and tested the model 10 times. We trained each model for 15 epochs and Adam [45] was used as the optimizer ($\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e^{-8}$). We obtained 92.05% ± 0.11 and 92.01% ± 0.11 training and test accuracy, respectively. One trained model with 98.46% test accuracy is used as our attack target model. For SSA, we generated corresponding adversarial signals for all clean signals including training data and test data. For SAA, we used a subset of training signals to generate universal perturbations.

B. Analysis of Signal-Specific Adversarial Signals

We compare performance of our proposed SSA and FGSM [23] for generating signal-specific adversarial signals. We evaluate their performance using average robustness [30] by

$$\hat{\rho}_{adv} = \frac{1}{\sigma(D)} \sum_{x \in D} \frac{\| \hat{r}(x) \|_2}{\| x \|_2},$$

where $\hat{r}(x)$ is the perturbation vector generated using an attack method (e.g., FGSM or SSA), and $\sigma(D)$ denotes the size of the test set $D$. As shown in Fig. 3 magnitudes of the adversarial perturbations
The running time. We choose $\epsilon$ higher the corresponding misclassification rate. Besides, the selection of the parameter characteristics of FGSM attack, the larger the parameter $\epsilon$ generated by the SSA are very small and the adversarial signals are imperceptible to human operators. (V/A p.u.).

Fig. 2: Architecture of the CNN used in the experiments, where Conv(number of features, kernel size, stride) denotes a convolution layer, MaxPool(kernel size, stride) denotes a pooling layer and BN denotes batch normalization.

Fig. 3: Experimental analysis of classification of randomly chosen clean signals (original) and the corresponding adversarial signals generated using the FGSM (first row), SSA (second row) and SAA (third row) methods. In titles of subfigures, C- $i$ (C- $j$) denotes a case where a signal belongs to class C- $i$ while its class label is predicted as C- $j$. The x-axis denotes time (s) and the y-axis denotes Voltage/Current (V/A p.u.).

generated by the SSA are very small and the adversarial signals are imperceptible to human operators. The $\epsilon$ parameter of the FGSM [23] is set as 0.5 to achieve the overall 0.88 misclassification rate which is proportion of signals whose labels are changed when perturbations are applied. According to the characteristics of FGSM attack, the larger the parameter $\epsilon$, the larger the adversarial perturbations, and the higher the corresponding misclassification rate. Besides, the selection of the parameter $\epsilon$ has no effect on the running time. We choose $\epsilon = 0.5$ to achieve roughly the same misclassification rate as the other two
methods, so as to analyze the effects of different attack methods. In Fig. 3, we observe that magnitudes of the adversarial perturbations generated by the FGSM \[23\] are larger and more perceptible than those generated by the SSA.

In Table I, we compare misclassification rate, average robustness and average running time of FGSM \[23\] and SSA. Table I shows that the SSA achieves a higher misclassification rate with smaller perturbation compared with the FGSM. Besides, we notice that the average running time spent for adversarial generation using the SSA is larger than using the FGSM. For example, the average running time spent for generating one adversarial signal is 0.22 seconds for the SSA, while that for the FGSM is 0.012 seconds. SSA spent most of the time for optimization of perturbation using Algorithm 1.

| Number of signals in training set (ratio) | 5000 (1.96%) | 10000 (3.92%) | 15000 (5.88%) | 20000 (7.84%) |
|-----------------------------------------|--------------|--------------|--------------|--------------|
| Misclassification rate (%)              | 28           | 35           | 52           | 74           |

C. Analysis of Signal-Agnostic Adversarial Signals

Unlike FGSM and SSA, SAA computes signal-agnostic (universal) perturbations using a training set of signals. We observed that the average time spent for training SAA to achieve misclassification rate 0.74 is 30 minutes. Less than 10\% of the total signals were used to generate the universal perturbation. Table III shows that one can build relatively generalizable universal perturbations with few signals. Besides, additional time is not spent for optimization during testing, and the universal perturbation is applied to each test signal in real-time (1.6 × 10⁻⁶ sec.). Universal perturbations and adversarial samples generated using $\xi = 1$ are shown in Fig. 3 (the average $l_2$ norm of a signal is 18). In the experiments, the misclassification rate can reach to 0.74 and most natural signals are misclassified with high probability.

Fig. 3 shows that voltage values of the universal perturbations are smaller and more imperceptible to human operators compared to the perturbations generated by FGSM. Although SSA and SAA can generate perceptually similar adversarial samples for some classes, the samples generated by different methods can be misclassified to different classes. For instance, class labels of the adversarial samples belonging to $C$-3 which are generated by SSA and SAA, are incorrectly predicted by the learned model as $C$-12 and $C$-17, respectively.

D. Analysis of Performance of Black-Box Attacks

For training substitute models, we suppose that the attacker can access some data in the training set (obtained by invading a database or building an experimental platform for simulation). Two types of black-box setting are considered:

- Type I: The attacker knows the network architecture of the CNN model and does not know the specific model parameters.
- Type II: Both network architecture and specific parameters of the CNN model are not known to attackers (the substitute model architecture has several layers more or less than the corresponding target model in our experiments).

Besides, we also consider datasets of different sizes used for training substitute models, that is, using 1/20, 1/10 and 1/5 of the entire training set. For each black-box type and training data ratio, we trained 10 substitute models independently. The average test accuracy values of well-trained substitute models are shown in Table IV. Based on the well-trained substitute models, we generated adversarial signals based on the proposed attack methods in Section II. In detail, for each well-trained substitute model, we generated corresponding adversarial signals against the entire test data. Then, we use these generated adversarial signals to attack the target model.
Since two attack methods (SSA and SAA) are proposed in Section II, we use these two methods independently to generate adversarial signals and evaluate their effects. The average misclassification rates of adversarial signals against the target model are shown in Table V and VI, respectively. Table V shows that black-box attacks for SSA have a low success rate. Besides, the average misclassification rate of Type I is higher than that of Type II, which implies that the substitute model using the same architecture with the target model has a higher success rate. In contrast, Table VI implies that black box attacks for SAA have a high success rate, and there is no significant difference in the misclassification rate under the two black-box types. This interesting phenomenon reflects that in black-box attacks, signal-agnostic (universal) perturbations have a higher misclassification rate than signal-specific perturbations. The above experimental results show that the attacker can train his substitute models and generate corresponding universal perturbations to carry out black-box attacks, which is a huge and practical threat for CNN based PQ classification.

### TABLE IV: Average test accuracies of well-trained substitute models.

| Training data ratio | 1/20 | 1/10 | 1/5  |
|---------------------|------|------|------|
| Type I              | 94.9%| 96.4%| 99.3%|
| Type II             | 93.9%| 97.3%| 94.4%|

### TABLE V: Average misclassification rate of black-box attacks for SSA.

| Training data ratio | 1/20 | 1/10 | 1/5  |
|---------------------|------|------|------|
| Type I              | 4.7% | 5.1% | 5.3% |
| Type II             | 3.5% | 3.6% | 4.1% |

### TABLE VI: Average misclassification rate of black-box attacks for SAA.

| Training data ratio | 1/20 | 1/10 | 1/5  |
|---------------------|------|------|------|
| Type I              | 73.2%| 74.1%| 75.6%|
| Type II             | 72.6%| 75.5%| 76.1%|

E. Graphical Analysis of Class Predictions under Attacks

We normalized confusion matrices generated according to class predictions obtained using SAA, FGSM and SSA in Fig. 4, Fig. 5 and Fig. 6, respectively. For comparison, the normalized confusion matrix based on the normal test signals is also provided in Fig. 7. These normalized confusion matrices are used to construct class-confusion graphs proposed below.
Fig. 4: A normalized confusion matrix obtained using the SAA.

Fig. 5: A normalized confusion matrix obtained using the FGSM.
We analyze how class predictions of learned models change under attacks exploring class confusion statistics. We construct a directed graph $G = (V, E)$, where each node $v_i \in V$ denotes a class label C-$i$. A weighted directed edge $e_{i,j} \in E$ connecting a node $v_i$ to the node $v_j$ indicates that signals belonging
Fig. 8: Visualization of graphs for each attack method. Directions of edges are depicted by color instead of using directed arrows. For instance, the edge $e_{10,17}$ is directed from the node $C-10$ to the node $C-17$. Therefore, the color of the edge $e_{10,17}$ which is green, is same to that of the node $C-10$ in (a). Thickness of edges is determined using normalized covariance matrices. For instance, in (a), the number of samples belonging to $C-10$ which are misclassified as $C-17$ is larger than that of samples belonging to $C-16$ which are misclassified as $C-17$.

| TABLE VII: Degrees of nodes of the class-confusion graphs of FGSM, SSA and SAA. |
|-----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FGSM            | C-1 | C-2 | C-3 | C-4 | C-5 | C-6 | C-7 | C-8 | C-9 | C-10 | C-11 | C-12 | C-13 | C-14 | C-15 | C-16 | C-17 |
| In-degree ($d_{in}$) | 0   | 0   | 0   | 0   | 0   | 0   | 8   | 0   | 0   | 0    | 0    | 0    | 3    | 11   | 0    | 0    | 16   |
| Out-degree ($d_{out}$) | 2   | 2   | 2   | 2   | 2   | 2   | 3   | 2   | 2   | 2    | 3    | 2    | 2    | 2    | 3    | 3    | 3    |
| SSA             | C-1 | C-2 | C-3 | C-4 | C-5 | C-6 | C-7 | C-8 | C-9 | C-10 | C-11 | C-12 | C-13 | C-14 | C-15 | C-16 | C-17 |
| In-degree ($d_{in}$) | 1   | 2   | 5   | 1   | 2   | 3   | 7   | 5   | 4   | 1    | 3    | 3    | 4    | 4    | 2    | 1    | 5    |
| Out-degree ($d_{out}$) | 4   | 1   | 1   | 3   | 5   | 3   | 1   | 5   | 5   | 2    | 3    | 4    | 3    | 4    | 3    | 3    | 3    |
| SAA             | C-1 | C-2 | C-3 | C-4 | C-5 | C-6 | C-7 | C-8 | C-9 | C-10 | C-11 | C-12 | C-13 | C-14 | C-15 | C-16 | C-17 |
| In-degree ($d_{in}$) | 0   | 0   | 0   | 0   | 0   | 6   | 4   | 1   | 0   | 0    | 0    | 6    | 5    | 0    | 0    | 16   |
| Out-degree ($d_{out}$) | 1   | 2   | 3   | 3   | 1   | 1   | 2   | 3   | 4   | 1    | 3    | 3    | 2    | 2    | 3    | 3    | 3    |

The weight $w_{i,j}$ of the edge $e_{i,j}$ is the ratio of the number of samples of the class $C-i$ which are misclassified as $C-j$, to the total number of samples in the class $C-i$.

Class-confusion graphs computed for each attack method are visualized in Fig. 8. For the sake of clearance of visualizations, we only depict connections denoting classification probability greater than chance. Distributions of node degrees are given in Table VII. Out-degree value of a node $v_i$ associated with a class $C-i$ denotes the number classes to which samples of the $C-i$ are incorrectly assigned by misclassification. For instance, samples belonging to $C-15$ which are attacked by SSA are misclassified to the other three different classes $C-2$, $C-8$ and $C-11$ (see Fig. 8). In-degree value of a node $v_i$ denotes the number classes whose samples are misclassified as $C-i$. For instance, samples belonging to $C-8$ and $C-13$ are misclassified to $C-15$ (see Fig. 8).

Fig. 8(a) and Fig. 8(b) show that perturbations generated by the FGSM and SAA make learned models misclassify signals to a few dominant classes such as $C-7$, $C-13$, $C-14$ and $C-17$. We conjecture that signals belonging to these dominant classes are distributed in larger areas in feature spaces (see Fig. 10).

There are 17 types of PQ signals and the number of signals of each type is the same. Therefore, we assume that the probability of each signal being classified into any kind of signal is $1/17$ (the assumption is only to better show corresponding class-confusion graphs, because displaying large number of edges in figures makes corresponding graphs look messy), which is $1/17$. 

TABLE VIII: Weighted and unweighted graph entropy values.

| Entropy | Out-degree ($d_{out}$) | In-degree ($d_{in}$) |
|---------|------------------------|---------------------|
| $H_w(G)$ | FGSM | SSA | SAA | FGSM | SSA | SAA |
| 13.10   | 21.29 | 13.97 | 10.87 | 20.84 | 11.75 |
| $H(G)$  | FGSM | SSA | SAA | FGSM | SSA | SAA |
| 19.34   | 25.06 | 17.09 | 12.04 | 23.53 | 13.49 |

and thereby, models can be fooled by assigning labels of signals to these classes with higher probability compared to the other classes. Besides, most of the signals belonging to $C-17$, are not misclassified when universal perturbations are applied. The reason may be that decision boundary of $C-17$ may be more complicated compared to that of the other classes, and located far from that of the other classes (see Fig. [10]). Therefore, universal perturbations may not be able to fool models for misclassifying signals of $C-17$.

In order to analyze the statistical relationship between classes whose samples are successfully attacked and failed to be attacked, we compute entropy of each node $v_i$ by

$$H_w(v_i) \triangleq -\sum_{j=1}^{d_i} p_{i,j} \log(p_{i,j}), \quad \text{and} \quad p_{i,j} = \frac{w_{i,j}}{\sum_{j=1}^{d_i} w_{i,j}}, \quad (10)$$

where $d_i \in \{d_{in}, d_{out}\}$ is either in-degree or out-degree of $v_i$. We compute the entropy for all the nodes of the graph $G$ by

$$H_w(G) \triangleq \sum_{v_i \in V} H_w(v_i). \quad (11)$$

We calculated unweighted graph entropy $H(G)$ using $w_{i,j} = 1, \forall e_{i,j} \in E$. Entropy values given in Table VIII show that SSA provides the largest entropy values. This result shows that signals misclassified under SSA are more equally likely distributed to incorrect classes compared to FGSM and SAA. We obtained the lowest values of $H_w(G)$ and $H(G)$ for the in-degree distribution, and those of $H_w(G)$ for the out-degree distribution using FGSM. However, SAA provided the lowest unweighted entropy for the out-degree distribution. This result indicates that SAA makes the learned model misclassify samples to less number of particular signal classes with higher probability compared to FGSM and SAA. This observation is depicted for the graph computed using the SAA with less number of node hubs in Fig. 8(a), compared to the graphs computed using the FGSM (Fig. 8(b)) and the SSA (Fig. 8(c)).

TABLE IX: Comparison of classification accuracy (%) for adversarial training (adv.) using SSA.

| Adv. Training Epochs | Acc. on Training Data | Acc. on Test Data | Acc. on Adv. Training Data | Acc. on Adv. Test Data |
|---------------------|-----------------------|-------------------|---------------------------|------------------------|
| 0                   | 98.54                 | 98.46             | 2.09                      | 2.09                   |
| 1                   | 91.31 ± 10.03         | 91.08 ± 11.33     | 90.14 ± 10.14             | 90.06 ± 10.12          |
| 10                  | 83.26 ± 7.18          | 83.14 ± 8.29      | 93.15 ± 5.72              | 93.26 ± 5.52           |

F. Analysis of Performance of Adversarial Training

We fine-tuned the learned model for 10 additional epochs using generated adversarial signals to improve its robustness against adversarial perturbation. Although we generated adversarial signals for training and test data, only adversarial signals in training data were used for adversarial training. Adversarial signals in test data were used to evaluate performance of adversarial training. For each trial, same adversarial signals were used in all additional epochs. We fine-tuned the model for 10 more epochs using the original data for a fair comparison.
Fig. 9: Analysis of change of average robustness $\hat{\rho}_{adv}$ of the learned models using SSA and FGSM.

Change of $\hat{\rho}_{adv}$ for different adversarial training strategies is depicted in Fig. 9. The robustness $\hat{\rho}_{adv}$ is estimated using SSA, since it is the best performing method, while worst performing SAA is not used in this experiment (see Table I). Fig. 9 shows that adversarial training using adversarial signals generated by SSA significantly increases robustness of the learned models to attacks. For example, robustness of the learned models is improved by 200% after training by three epochs. However, training the learned models with adversarial signals generated by FGSM [23] can decrease the robustness of the learned models (note that we used relatively large perturbations by FGSM $\epsilon = 0.5$ due to its high misclassification rate). The reason of this result may be that the perturbations generated by the FGSM are larger than those generated by the SSA. Adversarial training with extremely perturbed signals decreases robustness of the learned models to adversarial perturbations [30].

G. Accuracy-Robustness Trade-off

Next, learned models are fine-tuned using combined signals (mixing of normal signals and adversarial signals generated by SSA) to analyze change of their accuracy (FGSM and SAA are not used since they need relatively large perturbations to achieve high misclassification rate). Results show that although adversarial training improved the classification accuracy using adversarial PQ signals to 94.09%, the classification accuracy of the model retrained using normal PQ signals is decreased to 81.34% after training with 10 additional epochs (see Table IX). The reason may be that fine-tuning of models with generated adversarial signals can decrease the model’s ability to express normal signals to a certain extent. This phenomenon also exists in other fields such as the image domain [46], [47]. We propose the trade-off between maintaining accuracy of classifiers on normal PQ signals and improving their robustness to adversarial signals as an open challenge, which needs to be better utilized and resolved in future research.

H. Visualization of Learned Feature Representations of PQ Signals under Attacks

Visual inspection of attacks on PQ signals in the Smart Grid is not a well-defined phenomenon, since their interpretation is not clear using measurement devices. To address this problem, we propose an approach by first projecting high dimensional signal and feature vectors of samples to low dimensional spaces. Then, we visualize samples in projected spaces to explore their discrimination among PQ classes.

We used t-SNE [24] for projection of high dimensional vectors. We first projected signal vectors residing in input signal spaces and feature vectors obtained at the last hidden layer of CNN models into a 2D space $\mathcal{X}$ and a 2D feature space $\mathcal{F}$, respectively. Then, we depicted samples on the projected spaces by points which are colored according to their class labels. Projection quality is evaluated by neighborhood
hit (NH), which indicates how well samples belonging to classes are visually discriminated. For a given \( k \) (in our work, \( k = 5 \)), the NH for a point \( x \) is the ratio of its \( k \)-nearest neighbors that belong to the same category as \( x \) \[48\]. The NH for the entire projection is the average NH over its points. 1000 randomly chosen normal PQ signals, adversarial signals generated by SSA and SAA are visualized in Fig. 10. Fig. 10(a), (d) and (g) show poor discrimination of normal samples, and adversarial signals generated by SSA and SAA in the projected spaces \( X_n \), \( X_{ssa} \) and \( X_{saa} \), respectively. This result shows difficulty of discriminating samples using the input signals (normal signals or adversarial signals). This qualitative result is also confirmed by their low NH values.

In Fig. 10(b), samples are depicted in a 2D feature space \( F_r \) obtained before training of CNN models. This result indicates that a CNN model with randomly initialized parameters has a poor visual discrimination. It is reasonable to hypothesize that visual discrimination would be improved after training of models. This is related to the conjecture that CNNs can learn features useful for human perception of classes \[49\]. To analyze this hypothesis, we first trained a CNN model and extracted features of samples at the last hidden layer of the model. Then, we projected features into a 2D feature space denoted by \( F_{tr} \). Samples are depicted in the projected space \( F_{tr} \) in Fig. 10(c). Compared to Fig. 10(a) and (b), visual discrimination is significantly improved in Fig. 10(c) and the corresponding NH is 87.78\% and the test accuracy is 98.9\%.

Fig. 10(e) shows that features of adversarial signals generated by SSA are not visually better discriminative compared with Fig. 10(c). Therefore, the adversarial attacks affect visual discrimination of the features. The test accuracy for the 1000 randomly chosen adversarial signals generated by SSA is
only 2.7%. However, features of the adversarial signals are visually distinguishable to a certain degree, since the SSA can successfully fool top layer softmax classifiers for misclassification of samples. Visual discrimination of samples is improved in a projected space $\mathcal{F}_{\text{adv}}$ of features learned after adversarial training of the model (see Fig. 10(f)). This result visually verifies the efficiency of adversarial training.

In Fig. 10(i), we visualize samples generated by SAA in a projected space of features $\mathcal{F}_{\text{saa}}$ learned after adversarial training of the model. Although the NH is only 57.78%, visual discrimination of adversarial PQ signals is improved by adversarial training. This may suggest that adversarial training of models using adversarial PQ signals generated by an attack method can improve their robustness against adversarial PQ signals generated by other attack methods. Results obtained using our proposed approach are complementary to those obtained by visualization of raw signals in Fig. 3. We believe that researchers can use the proposed approach for qualitative analysis of class discrimination of normal and adversarial PQ signals under attacks in the Smart Grid.

V. CONCLUSION AND DISCUSSION

We proposed signal-specific adversarial attack (SSA) and signal-agnostic adversarial attack (SAA) methods to attack learning models for classification of power quality signals by generating adversarial signals in the Smart Grid. We proposed and evaluated black-box attacks based on transferable characteristics and the above two methods. We adopted adversarial training to defend learned models against adversarial signals. Experimental results show that our proposed SSA provides less perturbation compared to the state-of-the-art FGSM which was adapted for power signal analysis, while the SAA can generate the universal perturbation that can fool learned models. The attack method based on the universal signal-agnostic algorithm has a higher transfer rate of black-box attacks than the attack method based on the signal-specific algorithm, which is a huge and practical threat for CNN based PQ classification. In the analyses, the proposed adversarial training method improved robustness of the learning models. In addition, we proposed an approach for qualitative analysis of normal power quality signals and visual inspection of adversarial attacks. We elucidated the mechanisms behind robustness of models to adversarial attacks corresponding to the quantitative results using the proposed visualization approach. We believe that the proposed approaches will be useful for researchers and practitioners to analyze power systems and lead new research directions. In the future work, we plan to employ our approaches for a wider class of attack models (such as gray-box and black-box attacks with higher misclassification rates) and defense methods for data, network and device security in the Smart Grid.

REFERENCES

[1] J. Tian, T. Li, F. Shang, K. Cao, J. Li, and M. Ozay, “Adaptive normalized attacks for learning adversarial attacks and defenses in power systems,” in 2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm). IEEE, 2019, pp. 1–6.
[2] “IEEE recommended practice for monitoring electric power quality,” IEEE Std 1159-2009 (Revision of IEEE Std 1159-1995), pp. 1–94, June 2009.
[3] R. Igual, C. Medrano, F. J. Arcega, and G. Mantescu, “Integral mathematical model of power quality disturbances,” in 2018 18th International Conference on Harmonics and Quality of Power (ICHQP). IEEE, 2018, pp. 1–6.
[4] O. P. Mahela, A. G. Shaik, and N. Gupta, “A critical review of detection and classification of power quality events,” Renewable and Sustainable Energy Reviews, vol. 41, pp. 495–505, 2015.
[5] S. Deokara and L. Waghmare, “Integrated DWT–FFT approach for detection and classification of power quality disturbances,” International Journal of Electrical Power & Energy Systems, vol. 61, pp. 594–605, 2014.
[6] M. Mishra, “Power quality disturbance detection and classification using signal processing and soft computing techniques: A comprehensive review,” International Transactions on Electrical Energy Systems, p. e12008, 2018.
[7] S. Wang and H. Chen, “A novel deep learning method for the classification of power quality disturbances using deep convolutional neural network,” Applied Energy, vol. 235, pp. 1126–1140, 2019.
[8] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” ArXiv:1412.6572, 2014.
[9] H. Xu, Y. Ma, H. Liu, D. Deb, H. Liu, J. Tang, and A. Jain, “Adversarial attacks and defenses in images, graphs and text: A review,” arXiv preprint arXiv:1909.08072, 2019.
[10] E. R. Balda, A. Behboodi, and R. Mathar, “Perturbation analysis of learning algorithms: Generation of adversarial examples from classification to regression,” IEEE Transactions on Signal Processing, vol. 67, no. 23, pp. 6078–6091, 2019.
Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 1778–1783.

Y. Chen, Y. Tan, and B. Zhang, “Exploiting vulnerabilities of load forecasting through adversarial attacks,” in *Proceedings of the Tenth ACM International Conference on Future Energy Systems*. ACM, 2019, pp. 1–11.

X. Zhou, Y. Li, C. A. Barreto, J. Li, P. Volgyesi, H. Neema, and X. Koutsoukos, “Evaluating resilience of grid load predictions under stealthy adversarial attacks,” in *2019 Resilience Week (RWS)*, vol. 1. IEEE, 2019, pp. 206–212.

Z. Tang, J. Jiao, P. Zhang, M. Yue, C. Chen, and J. Yan, “Enabling cyberattack-resilient load forecasting through adversarial machine learning,” in *2019 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 2019, pp. 1–5.

C. Barreto and X. Koutsoukos, “Design of load forecast systems resilient against cyber-attacks,” in *International Conference on Decision and Game Theory for Security*. Springer, 2019, pp. 1–20.

T. Liu and T. Shu, “Adversarial false data injection attack against nonlinear AC state estimation with ANN in smart grid,” in *International Conference on Security and Privacy in Communication Systems*. Springer, 2019, pp. 365–379.

I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” *arXiv preprint arXiv:1412.6572*, 2014.

L. van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008.

B. D. Rouani, M. Samragh, T. Javidi, and F. Koushanfar, “Safe machine learning and defeating adversarial attacks,” *IEEE Security & Privacy*, vol. 17, no. 2, pp. 31–38, 2019.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, “Intriguing properties of neural networks,” *arXiv preprint arXiv:1312.6199*, 2013.

N. Carlini and D. Wagner, “Towards evaluating the robustness of neural networks,” in *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2017, pp. 39–57.

A. Modas, S.-M. Moosavi-Dezfooli, and P. Frossard, “Sparsefool: a few pixels make a big difference,” in *Proceedings of 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 9087–9096.

J. Su, D. V. Vargas, and K. Sakurai, “One pixel attack for fooling deep neural networks,” *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 5, pp. 828–841, 2019.

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

A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” *arXiv preprint arXiv:1607.02533*, 2016.

N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, “The limitations of deep learning in adversarial settings,” in *2016 IEEE European symposium on security and privacy (EuroS&P)*. IEEE, 2016, pp. 372–387.

C. Fawzi, S.-M. Moosavi-Dezfooli, and P. Frossard, “The robustness of deep networks: A geometrical perspective,” *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 50–62, 2017.

C. Liao, H. Zhong, A. Squicciarini, S. Zhu, and D. Miller, “Backdoor embedding in convolutional neural network models via invisible perturbation,” *arXiv preprint arXiv:1808.10307*, 2018.

M. Esmaeilpour, P. Cardinal, and A. L. Koerich, “From sound representation to model robustness,” *arXiv preprint arXiv:2007.13703*, 2020.

J. Lu, T. Issaranon, and D. Forsyth, “Safetynet: Detecting and rejecting adversarial examples robustly,” in *Proceedings of 2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 446–454.

G. F. Montufar, R. Pascanu, K. Cho, and Y. Bengio, “On the number of linear regions of deep neural networks,” in *Advances in neural information processing systems*, 2014, pp. 2924–2932.

M. Raghu, B. Poole, J. Kleinberg, S. Ganguli, and J. Sohl-Dickstein, “On the expressive power of deep neural networks,” in *International conference on machine learning*. PMLR, 2017, pp. 2847–2854.

E. Wong and Z. Kolter, “Provable defenses against adversarial examples via the convex outer adversarial polytope,” in *International Conference on Machine Learning (ICML)*. PMLR, 2018, pp. 5286–5295.

S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, “Universal adversarial perturbations,” in *Proceedings of 2017 IEEE conference on computer vision and pattern recognition (CVPR)*, 2017, pp. 1765–1773.

N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, “Practical black-box attacks against machine learning,” in *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, 2017, pp. 506–519.

N. Papernot, P. McDaniel, and I. Goodfellow, “Transferability in machine learning: from phenomena to black-box attacks using adversarial samples,” *arXiv preprint arXiv:1605.07277*, 2016.
[45] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980, 2014.
[46] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan, “Theoretically principled trade-off between robustness and accuracy,” arXiv preprint arXiv:1901.08573, 2019.
[47] E. Arani, F. Sarfraz, and B. Zonooz, “Adversarial concurrent training: Optimizing robustness and accuracy trade-off of deep neural networks,” arXiv preprint arXiv:2008.07015, 2020.
[48] P. E. Rauber, S. G. Fadel, A. X. Falcao, and A. C. Telea, “Visualizing the hidden activity of artificial neural networks,” IEEE transactions on visualization and computer graphics, vol. 23, no. 1, pp. 101–110, 2016.
[49] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in Proceedings of 2018 IEEE conference on computer vision and pattern recognition (CVPR), 2018, pp. 586–595.