IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 14 August 2024, accepted 26 August 2024, date of publication 30 August 2024, date of current version 10 September 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3452476

RESEARCH ARTICLE

Adversarially Robust Fault Zone Prediction in Smart Grids With Bayesian Neural Networks

EMAD EFATINASAB[1], **ALBERTO SINIGAGLIA**[2], **NAHAL AZADI**[1], **GIAN ANTONIO SUSTO**[1,2], **(Senior Member, IEEE), AND MIRCO RAMPAZZO**[1], **(Member, IEEE)**

[1]Department of Information Engineering, University of Padua, 35131 Padua, Italy
[2]Human Inspired Technology Research Center, University of Padua, 35121 Padua, Italy

Corresponding author: Emad Efatinasab (emad.efatinasab@phd.unipd.it)

ABSTRACT The rapid growth of the global population, economy, and urbanization is significantly increasing energy consumption, necessitating the integration of renewable energy sources. This integration presents challenges that demand innovative solutions to maintain grid stability and efficiency. Smart grids offer enhanced reliability, efficiency, sustainability, and bi-directional communication. However, the reliance on advanced technologies in smart grids introduces vulnerabilities, particularly concerning adversarial attacks. This paper addresses two critical issues in smart grid fault prediction: the vulnerability of machine learning models to adversarial attacks and the operational challenges posed by false alarms. We propose a Bayesian Neural Network (BNN) framework for fault zone prediction that quantifies uncertainty in predictions, enhancing robustness and reducing false alarms. Our BNN model achieves up to 0.958 accuracy and 0.960 precision in fault zone prediction. To counter adversarial attacks, we developed an uncertainty-based detection scheme that leverages prediction uncertainty. This framework distinguishes between normal and adversarial data using predictive entropy and mutual information as metrics. It detects complex white-box adversarial attacks, which are challenging due to attackers' detailed knowledge of the model, with a mean accuracy of 0.891 using predictive entropy and 0.981 using mutual information. The model's performance, combined with minimal computational overhead, underscores its practicality and robustness for enhancing smart grid security.

INDEX TERMS Adversarial attacks, Bayesian neural networks, fault prediction, smart grids, uncertainty quantification.

I. INTRODUCTION

The swift growth of the global population and economy, coupled with increasing urbanization, is anticipated to elevate energy consumption substantially. This escalating demand coincides with the integration of renewable energy sources, introducing distinctive challenges. The balancing act between managing the heightened energy needs and ensuring the stable incorporation of renewable energy systems necessitates innovative solutions to maintain grid stability, enhance efficiency, and promote sustainability in energy distribution. Smart grids represent a groundbreaking shift in the realm of energy distribution [1], [2]. This technology exemplifies a modern electric power grid marked by improved reliability, efficiency, sustainability, and bi-directional communication capabilities [2]. In contrast to conventional power distribution systems, smart grids utilize real-time data, communication networks, and intelligent control mechanisms to enhance the efficiency of electricity generation, distribution, and consumption. This enables a two-way exchange of information between utilities and customers, fostering a dynamic and responsive energy ecosystem [2]. Given their pivotal role as the cornerstone of a nation's energy infrastructure, smart grids are classified among critical infrastructures necessitating protection [3]. With smart grids relying more on data-driven technologies,

The associate editor coordinating the review of this manuscript and approving it for publication was Ali Raza.

ensuring robust security measures becomes paramount to safeguard the confidentiality, integrity, and availability within the energy infrastructure [4]. The broad interconnection among devices and remote access points amplifies the attack surface, thereby creating potential vulnerabilities for malicious actors to exploit [5], [6]. While the effective integration of Artificial Intelligence (AI) technologies highlights the transformative impact of smart grids on modernizing the electricity sector, they also pose significant vulnerabilities, making them a critical aspect to address within these systems [7]. Machine Learning (ML) and AI have demonstrated remarkable efficiency in various tasks within smart grids. Despite numerous papers in the literature presenting new models and methodologies for different aspects of smart grids [8], [9], [10], [11], [12], [13], [14], [15], [16], a significant gap persists in addressing the inherent vulnerabilities associated with these methodologies, particularly concerning adversarial attacks. In the context of machine learning, ‘‘adversarial’’ refers to instances in which harmful inputs or data are purposefully designed to fool or influence a model's predictions or behavior. Adversarial attacks exploit vulnerabilities in models by introducing subtle perturbations to input data.

One critical application of AI in smart grids is fault zone prediction, which involves anticipating areas where faults have accrued or are likely to occur. This capability is essential for maintaining grid stability and preventing outages. While AI and ML techniques have been widely employed to provide advanced predictive capabilities and enhance the efficiency of grid management, it is imperative to address the inherent vulnerabilities of these methodologies. One significant yet often overlooked issue in the literature is their susceptibility to adversarial attacks. They also introduce potential challenges beyond the susceptibility of these models to adversarial attacks. Misclassifications are inevitable, and issuing notifications to grid operators for every detected fault could result in numerous false alarms, creating operational difficulties. High rates of false alarms can undermine the credibility of fault prediction systems, leading to unnecessary maintenance actions, increased operational costs, and potential disruptions in grid operations. The issue of false alarms and the need for robust fault detection methods have not been adequately addressed in the literature on the practical implementation of these systems. Therefore, enhancing the security and robustness of AI-based fault prediction methods is crucial for the reliable operation of smart grids. By addressing these goals, researchers can significantly enhance the practical applicability of smart grid fault prediction models, ensuring that they are not only theoretically sound but also robust, reliable, and secure in real-world environments.

In this paper we introduce a framework based on Bayesian neural network (BNN) for the task of fault zone prediction in smart grids. As fault prediction is an exceedingly sensitive scenario, possible false alarms or misclassifications can lead to significant problems. In such a critical context, the ability to quantify uncertainty becomes paramount. By understanding the uncertainty associated with each prediction, operators can make more informed decisions, thereby reducing the likelihood of false alarms and improving the overall robustness of the fault prediction system. Leveraging BNN we can tackle this issue by offering a quantifiable measure of uncertainty in the model's predictions while also enhancing robustness through Bayesian regularization. This approach not only enhances the reliability of the predictions but also helps in identifying potential misclassifications. We illustrate that our model achieves an accuracy of up to 0.958 in the fault zone prediction task. To address the vulnerabilities posed by adversarial attacks, we will develop an adversarial detection scheme that leverages the uncertainty of the predictions to detect potent white-box adversarial attacks. We show that using this uncertainty-based detection mechanism, we are able to reach up to 0.981 accuracy in detecting potent white-box adversarial attacks.
Our contributions can be summarized as follows.

- We introduce a novel realistic system and threat model that mirrors real-world scenarios encountered during the training of a fault zone prediction system and potential scenarios of attacks against these systems.
- We propose an LSTM-based BNN, a resilient framework for predicting fault zones in smart grids, which helps quantify the uncertainty of predictions. We assess the performance of our fault zone prediction model against existing models in the literature. Our evaluation provides a reference for future studies on fault prediction models and their security.
- We propose a lightweight uncertainty-based adversarial attack detection system capable of detecting complex adversarial attacks generated with different amounts of adversarial noise.
- We evaluate our models, attacks, and defenses on a publicly available dataset, showing the efficacy of the attacks and the capabilities of our adversarial attack detection scheme. We show an accuracy of up to 0.958 for our fault zone prediction model and an accuracy of up to 0.981 for our uncertainty-based adversarial attack detection system.

The rest of the paper is organized as follows: We mentioned the challenges and limitations of related works in Section II. We propose the system and threat models in Section III. In Section IV, we discuss the attacks employed against fault zone prediction systems. We describe our methodology of Bayesian fault prediction system and Uncertainty-based adversarial attack detection scheme in Section V. Then, we evaluate our attacks, Bayesian fault prediction system, and Uncertainty-based adversarial attack detection scheme in Section VI. Finally, we conclude our work in Section VII.

II. RELATED WORKS

A. BAYESIAN NEURAL NETWORKS

A BNN is typically understood as a stochastic artificial neural network trained using Bayesian reasoning, although its definition can vary slightly in different sources [17]. The fundamental principle of the Bayesian approach is to build posterior probability distributions for all unknown variables in a model based on the given data samples [18]. In the Bayesian approach, the parameter space ω is represented by a distribution $p(\omega)$, treating each parameter of the model as a random variable. The likelihood function $p(Y \mid X, \omega)$ describes the likelihood of observing the data under the model. The final aim of the Bayesian approach is to estimate the posterior distribution $p(\omega|X, Y)$. However, such a task can be extremely computationally expensive [19], and thus, it requires to rely on approximations. One such approximation widely used tries to find a good posterior approximation via optimization, matching $q(\omega) \approx p(\omega|X, Y)$, $q(\omega)$ being the prior, by minimizing $KL[p(\omega|X, Y)||q(\omega)]$. To do so, it uses a mean-field assumption on the parameters, thus assuming them independent from each other and minimizing such distance using ELBO [20]. By doing so, we get a feasible optimization objective shown in equation 1.

$$\text{argmax}_{\omega}\mathcal{L}(\omega) = \mathbb{E}_{p(\omega)}[p(y|x, \omega)] - KL[p(\omega|X, Y)||q(\omega)] \tag{1}$$

where the Kullback–Leibler divergence is defined as $KL[p(x)||q(x)] = \int_{x\in\mathcal{X}} p(x) \log \frac{p(x)}{q(x)}$, and $p(\omega)$ usually is chosen from a family of parametrized distributions. In such cases, the task is to find $\omega = \{\mu, \sigma\}$ such that the posterior $\mathbb{E}_{q(\omega)}[p(y|x, \omega)]$ is maximized, thus obtaining good performances in the task we are interested while minimizing $KL[q(\omega|X, Y)||p(\omega)]$, thus not moving too far from the prior $p(\omega)$. Initially, such methods used the REINFORCE [21] estimator to estimate the gradient [22]:

$$\begin{aligned}\frac{d\mathcal{L}(\omega)}{d\omega} &\approx \mathbb{E}_{\omega\sim q(\omega)}\left[\nabla_{\omega} \log q(\omega) \log \frac{p(X, Y, \omega)}{q(\omega)}\right]\\ &= \mathbb{E}_{\omega\sim q(\omega)}[\nabla_{\omega} \log q(\omega)(\log p(Y, |X, \omega)\\ &\quad - KL[q(\omega|X, Y)||p(\omega)]).\end{aligned} \tag{2}$$

This method allows us to have a black box estimator for the parameters. However, such an estimator suffers from high variance. Instead, thanks to the properties of Gaussian distributions, we can efficiently find such parameters via Bayes By Backprop (BBB) [23] using the reparametrization trick [20]. This technique allows us to sample elements from any Gaussian distribution while still being differentiable with its parameters, thanks to the equation shown in equation 3.

$$\theta = \mu + \sigma \odot \epsilon, \ \epsilon \sim N(0, 1) \tag{3}$$

Once such posterior has been successfully approximated, it can be used to calculate the posterior predictive distribution, shown in equation 4:

$$\begin{aligned}p(Y^*|X^*, X, Y) &= \mathbb{E}_{p(\omega|X,Y)}[p(Y^* \mid X^*, \omega)]\\ &= \int p(Y^*|X^*, \omega)p(\omega|X, Y)d\omega,\end{aligned} \tag{4}$$

where X^* is the test input and Y^* is the prediction [24]. Such integral can be approximated via Monte Carlo Sampling. Stochastically sampling network parameters during training is known as weight perturbation. However, in [23], they debate how such approximation may lead to an underestimation of the uncertainty.

For this reason, in the last years, new approaches have been proposed to tackle both the uncertainty underestimation and the variance of the gradient, one of which is called Flipout [25], decorrelates gradients within each data batch, enhancing the inference process of BNNs. It exploits the fact that, usually, a mean-field assumption is made for the posterior approximation, thus considering the parameters being independent, and that the distribution for the weight perturbation is symmetric around zero. Thanks to this assumption, Flipout uses a shared weight perturbation matrix N for all the samples in the minibatch sampled from a distribution symmetric around zero $q(\theta)$, and a different rank-1 matrix $s_n {s'_n}^T$ for each sample, where s_n, s'_n are vectors whose entries are sampled uniformly from $\{+1, -1\}$. Concretely, the final perturbation is computed as follows:

$$\hat{P} = N + s_n {s'_n}^T, \ N \sim q(\theta), \ s_n \text{ and } s'_n \sim U(-1, +1). \tag{5}$$

By employing this approach, Flipout decorrelates the gradients between minimatch samples, thus efficiently reducing the variance of the produced gradient and allowing the optimization to be faster and more effective. In the original paper [25], it has been shown to be particularly effective in training LSTM-based models, which is why it will be employed for this work.

B. FAULT PREDICTION

Numerous studies have explored fault detection and classification techniques in smart grids [8], [9], [10], [11], [12], [13], [14], [15], [16]. As delineated by Saha et al. [26], the classification of fault location methodologies in power systems encompasses traditional, observant, and intelligent approaches. One example of a traditional approach could be a customer informing the operator about issues like downed wires or a burning smell from a cable. In contrast, the observant approach employs intelligent meters or local detectors that automatically alert the system operator via communication feedback. Finally, the intelligent approach utilizes smart sensors or expert systems like AI to detect faults autonomously [10]. This study concentrates explicitly on intelligent methodologies for fault detection, including expert systems, Machine learning, and Deep learning, all directed toward detecting faults within the system. Artificial Neural Networks (ANNs) have been widely investigated in the literature for fault identification and prediction [27], [28], [29],

[30], [31], [32], [33], [34], [35]. For instance, Thukaram et al. [35] suggested a hybrid method that merges Support Vector Machine (SVM) with ANN architectures. Also, the use of a Recurrent neural network has been reported in the literature for the fault classification task [36], [37], [38]. For instance, Zhang et al. [38] presented a methodology utilizing the attention mechanism, Bidirectional Gated Recurrent Unit (GRU), and a dual-structured network to analyze data from multiple viewpoints.

Many studies have employed classical ML algorithms, such as Random Forest, for the task of fault prediction in the literature [39], [40], [41], [42]. Ghaemi et al. [42] presented an ensemble technique to improve the accuracy of fault node localization. Their approach combines the strengths of SVMs, K-Nearest Neighbors (KNN), and Random Forests. Furthermore, some works have utilized Bayesian networks for fault diagnosis. For instance, Yongli et al. [43] proposed three element-oriented models utilizing simplified Bayesian networks with Noisy-Or and Noisy-And nodes to estimate the faulty section of a transmission power system. Majidi et al. [44] proposed a fuzzy-c clustering method to identify potential fault points. Wilches-Bernal et al. [41] introduced a novel fault location and classification algorithm that combines mathematical morphology with random forests. Sapountzoglou et al. [45] proposed using a gradient-boosting tree model to detect, identify, and localize faults in low-voltage smart distribution grids.

Although fault prediction models for smart grids have been widely discussed and implemented in the literature, their security and robustness have not been thoroughly examined. These models are, in fact, susceptible to adversarial attacks [4], [46]. For instance, Ardito et al. [46] explored the robustness of fault type and zone classification systems in the face of such attacks. Their study involved evaluating the resilience of smart grid failure prediction systems by releasing datasets, conducting benchmarks, and assessing performance under adversarial conditions. The literature reveals a gap between the theoretical models and their industrial applications. While many studies have proposed innovative algorithms for fault detection and prediction, few have thoroughly examined issues such as adversarial attacks, false alarms, and the reliability of AI models under varying conditions. This disconnect highlights the need for a more integrated approach that combines theoretical rigor with practical considerations guiding future research efforts toward developing more secure and reliable fault prediction systems.

III. SYSTEM AND THREAT MODEL

We will now explore the system and threat model pertinent to our study. First, we will outline the standard functionality of the system when it operates in environments free from adversarial interference. Following this, we will examine the potential capabilities of attackers and the assumptions regarding their knowledge of the system.

A. SYSTEM MODEL

In a secure and unthreatened environment, without malicious attempts to disrupt the system, the model processes data from the smart grid infrastructure to perform fault zone predictions. The fault zone prediction model aims to identify the geographical location of faults within the smart grid. This involves analyzing data from various sources such as sensors, meters, and other monitoring devices distributed across the grid. The system gathers real-time data on electrical parameters like voltage, current, and frequency. This further involves analyzing the data to determine the specific area where a fault has occurred, which is essential for prompt and effective maintenance and repair operations. We assume that our model has been trained on clean, uncorrupted data, allowing them to accurately learn patterns and correlations from historical data. Once trained, these models are deployed within the smart grid system, where they continuously monitor and analyze incoming data to predict faults with high accuracy.

B. THREAT MODEL

As we strive to implement efficient defenses against adversaries targeting ML models in the smart grid, we define our threat model to encompass the most advantageous scenarios for the attacker. We assume the adversary can infiltrate the system and inject malicious data into the grid. This can be achieved through various methods, as exploiting both known and novel vulnerabilities has proven effective for gaining remote access [47], [48]. Such vulnerabilities may exist in various components of the smart grid, including sensors, communication channels, and control systems, making the grid susceptible to sophisticated cyberattacks. Once inside the infrastructure, the adversary aims to compromise the fault prediction models using adversarial examples. In the case of fault zone prediction, the adversary manipulates the models to misidentify the geographical location of faults within the smart grid. By altering the input data, the attacker can cause the model to predict that a fault has occurred in a different area than where it actually is. This would lead recovery teams to be dispatched to incorrect locations, causing delays in addressing the real issue and exacerbating the impact on operational efficiency. Such disruptions can lead to increased downtime, higher operational costs, and a loss of trust in the grid's reliability. In our threat model, we consider a white-box scenario where the attacker has access to both the data used for testing the model and the model's architecture and parameters. This scenario is highly advantageous for the attacker, enabling them to leverage this information to craft highly effective adversarial samples. Additionally, with access to the model weights, the adversary can fine-tune attack parameters offline, further enhancing the precision and impact of their attacks [4]. It is worth noting that while the white-box scenario may not be a common occurrence in the real world, it represents the most favorable conditions from the attacker's perspective. By considering a situation where the attacker has the upper hand, we can thoroughly evaluate the robustness of our defenses and ensure they are resilient even against highly knowledgeable and resourceful adversaries.

IV. ATTACKS

We now discuss the attacks that we employ against fault zone prediction systems in smart grids. In our white-box threat model, the adversary possesses full knowledge of the data and the trained model. For this reason, a potential attacker can exploit first-order information in order to carry out the attack:

$$\max_{\epsilon} \; L(f(x+\epsilon), y)$$
$$\text{s.t. } \|\epsilon\|_p \leq \gamma. \tag{6}$$

We analyze prominent adversarial attacks to reveal vulnerabilities in neural networks. We focus on specific attacks highlighted in the literature for their significance and capacity to uncover weaknesses.

- *Fast Gradient Sign Method (FGSM):* This attack quickly creates adversarial examples by utilizing the sign of the gradient of the loss function. Known for its computational efficiency, FGSM serves as a foundational benchmark for evaluating model robustness [49].
- *Carlini & Wagner (CW):* This sophisticated attack treats the creation of adversarial examples as an optimization problem, aiming to find minimal perturbations that lead to misclassification with minimal perceptibility. The CW attack challenges models with nearly imperceptible adversarial examples, testing their resistance to subtle perturbations [50].
- *Projected Gradient Descent (PGD):* PGD employs an iterative optimization approach but includes a projection step to keep perturbations within a specified constraint set. Known for creating potent adversarial examples, PGD allows for a rigorous examination of model robustness under stringent conditions [51].
- *Expectation Over Transformation Projected Gradient Descent (EOTPGD):* EOTPGD enhances PGD by incorporating randomness in the transformations applied to the input before calculating the gradient. This method aims to create adversarial examples that are robust to a range of transformations, further challenging the model's robustness under diverse conditions. This attack is specifically designed to target BNNs by leveraging their probabilistic nature [52].

V. METHODOLOGY

In this section, we introduce our proposed fault zone prediction system framework, illustrated graphically in Figure 1. We utilize two primary data sources for prediction: the legitimate sensor data collected from the smart grid and the potentially malicious data injected by adversaries. Initially, we employ the BNN model to make predictions. Subsequently, we assess the predictive uncertainty using measures such as predictive entropy and mutual information. Following prediction, we implement an uncertainty-based adversarial attack detection mechanism to identify adversarial samples. If a prediction is flagged as an adversarial attack, it is discarded, and an alarm is raised for grid operators. Conversely, if the prediction is deemed legitimate, we transmit it along with its associated uncertainty to the grid operators for further decision-making. This approach enables us to provide grid operators with predictive insights while accounting for the uncertainty inherent in the model's predictions. By incorporating uncertainty-aware detection mechanisms, we enhance the system's resilience to adversarial attacks, thereby fostering more robust and reliable decision-making in smart grid operations. While we list the components of our pipeline in order or appearance, it is worth noting that, in real-world scenarios, the first step would be training the fault prediction system. Indeed, our uncertainty-based adversarial attack detection mechanism uses the trained prediction model in its implementation. After training both the model and the uncertainty-based adversarial attack detection, components can be organized as detailed in Figure 1.

We present our fault zone prediction system in Section V-A and our uncertainty-based adversarial attack detection in Section V-B.

A. BAYESIAN FAULT PREDICTION SYSTEM

In our Fault zone prediction system implementation, a Bayesian Long Short-Term Memory network (LSTM) is trained using the Bayesian Torch library [53], which exploits Variational layers with Flipout Monte Carlo estimators [25] offering a probabilistic framework to handle uncertainty in model parameters. The model architecture consists of two LSTM layers followed by a fully connected layer, each incorporating the Flipout estimator used to learn the posterior. During training, the model is optimized using the Adam optimizer, and the loss function comprises two components: the standard cross-entropy loss and the Kullback-Leibler (KL) divergence. The addition of the KL divergence term comes from the definition of the Evidence Lower BOund (ELBO), and requires the model to not only minimize classification errors but also to minimize the discrepancy between the learned posterior distribution and a predefined prior distribution over the model parameters. We apply a scaling factor of 0.1 to the KL divergence term to control its regularizing effect, which prevents excessive deviation from the prior. For the posterior approximation, we choose to use the family of Gaussian distributions. The training process spans 120 epochs. At each epoch, the model's parameters are updated using backpropagation, with the optimizer adjusting the weights based on the gradients computed from the augmented loss function. The Bayesian LSTM is trained using Variational layers with Flipout Monte Carlo estimators, allowing it to capture uncertainty in its predictions and provide probabilistic outputs. This approach is particularly advantageous in scenarios where uncertainty estimation is

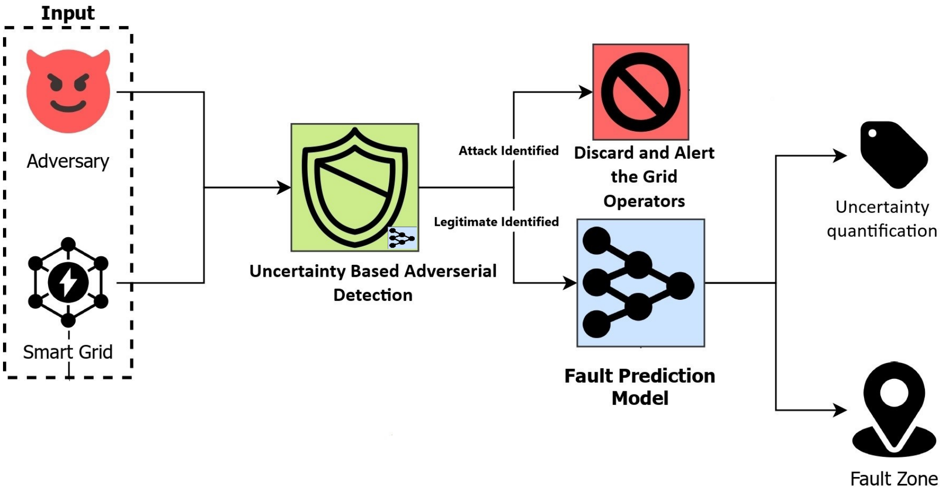

Input
Adversary
Smart Grid
Uncertainty Based Adverserial Detection
Attack Identified
Legitimate Identified
Discard and Alert the Grid Operators
Fault Prediction Model
Uncertainty quantification
Fault Zone

FIGURE 1. The proposed fault zone prediction system framework.

crucial, such as fault prediction in smart grids. To enhance the efficiency of Monte Carlo sampling and ensure stable training, we employ the Flipout technique in each layer. The specific details of our Bayesian LSTM architecture can be seen in Table 1. We employed the Golden Search Optimization algorithm (GSO) [54] to optimize our model's hyperparameters.

TABLE 1. Architecture of the Bayesian LSTM fault zone prediction system.

Layer	Type	Details
LSTM Layer 1	LSTMFlipout	(51,220)
LSTM Layer 2	LSTMFlipout	(220,440)
Fully Connected Layer (FC)	LinearFlipout	(440,4)
FC Output	Softmax Activation	Applied to Output

B. UNCERTAINTY-BASED ADVERSARIAL ATTACK DETECTION

In this section, we propose an uncertainty-based adversarial attack detection mechanism for adversarial attacks in smart grid systems. The approach leverages the predictive uncertainties computed by our Bayesian LSTM model to distinguish between legitimate and adversarial data. This section describes the various components of the detection mechanism and the experimental setup.

The Bayesian LSTM model is used to predict the fault zones in the smart grid system. Alongside the predictions, the model computes predictive uncertainties, which are crucial for detecting adversarial attacks.

In order to carry out the detection, we are interested in metrics that correlate with the severity of the adversarial attack. To do so, we will exploit the findings of [55] regarding properties of the Gaussian posterior approximation.

It is known that given the exact posterior distribution $p(Y^*|X^*, X, Y)$, we can get an estimate of the total uncertainty (TU) summing the aleatoric uncertainty (AU) and the epistemic uncertainty (EU), as shown in equation 7.

$$\underbrace{H[Y^*|X^*, X, Y]}_{\text{TU}} = \underbrace{I[Y^*; \omega|X^*, X, Y]}_{\text{EU}} + \underbrace{\mathbb{E}_{p(\omega|X,Y)}[H[Y^*|X^*, \omega]]}_{\text{AU}} \tag{7}$$

The H stands for entropy, defined as $H(X) = -\int_{x\in\mathcal{X}} p(x)\log p(x)$ and I stands for mutual information, defined as $I(X, Y) = KL[p(x, y)||p_X(x) \otimes p_Y(y)]$. Specifically, the left-hand side term in 7 is the total uncertainty, also known as predictive entropy, and the right-hand side terms are, respectively, the epistemic uncertainty, which we will call mutual information for the subsequent sections, and the aleatoric uncertainty.

Predictive entropy is indeed a good metric, as it incorporates information about the uncertainty of the sample for the model and the problem itself. However, due to approximations done to make the problem tractable, it fluctuates a lot, and thus, its measure might not be accurate. For this reason, we will also consider epistemic uncertainty, the term for mutual information, as it only considers uncertainty for the trained model. Such a definition of total uncertainty, however, is true for the exact posterior. Indeed, assuming a perfect model, $p(\omega|X, Y)$ would converge to a lambda distribution with all probability mass in the optimal ω, thus minimizing the epistemic uncertainty, leaving the aleatoric uncertainty as the only candidate for the total uncertainty.

Yet, in order to make the posterior estimation tractable, we aim to find a distribution $q(\omega)$ that approximates the true posterior $p(\omega|X, Y)$ from a specific class of family of distribution, which usually is the Gaussian family, as in our case. Due to such a choice, the posterior is forced to be unimodal. This unimodality is not that restrictive for simpler models, to the extreme for linear models, whose exact posterior is a Gaussian distribution, but does not hold at all for Neural Networks. In [19], the authors try to understand the topology of such a posterior, showing how complex it might look. For this reason, it's reasonable to assume that a Gaussian approximation is highly restrictive.

Furthermore, an adversarial attack tries to maximize the Negative Log Likelihood (NLL), which for this work corresponds to the Categorical Crossentropy:

$$L(\omega) = -\sum_{i}^{n}\sum_{j}^{c} y_{i,j}\log(p_\omega(y|x_i)_j), \tag{8}$$

where n is the number of samples in the minibatch, c the number of available classes, and y_i is a one-hot encoded vector.

It can be seen how an attacker, to maximize it, only needs to minimize the output of the model for the correct class. Indeed, since y_i is a one-hot encoded vector, the only term contributing to the loss is the one for the correct answer. Suppose such correct class is at index t and that a minibatch is composed by only 1 sample, then we aim at maximizing the following loss:

$$\max L(\omega) = -y_t \log p(y|x)_t. \tag{9}$$

It can be seen in equation 10 that, by taking the limit, we only need to minimize that single term of the model's output, shown in equation:

$$\lim_{p(y|x)_t \to 0} -y_t \log p(y|x)_t \to +\infty. \tag{10}$$

Furthermore, it can be seen that once fixed $p(y|x)_t$, $\sum_{1\neq t} p(y|x)_t = 1 - p(y|x)_t$, and that any allocation of that $1 - p(y|x)_t$ probability mass will lead to the same loss. Therefore, an attacker not only needs to minimize the probability of the correct answer but also is free to allocate the remaining probability mass in whichever criterion it prefers.

On top of this, [55] showed how Bayesian models, due to the Gaussian assumption of the posterior probability, are heavily outperformed by ensembles due to the lack of diversity in the sampled parameters. This is due to the fact that ensembles' models are free to learn different modes of the posterior, where a single BNN with Gaussian posterior approximation is limited to a single mode.

Finally, since (1) the posterior is badly approximated, (2) an attacker is free to allocate probability mass across the wrong labels with any criterion it wants, (3) parameters sampled by the learned posterior lack diversity, it's hard to assume that the current setting for BNN training is close enough to the initial assumptions for the total uncertainty to make it hold.

Indeed, already other works [56] have shown how the behavior of BNN's properties, especially for the predictive entropy and the epistemic uncertainty, is almost opposite to what the theory suggests, and they impute such mismatch to the coarse approximations done along the way to make the problem tractable too.

However, if such opposite behavior is detrimental to the promises that BNNs make about uncertainty estimation, we can still use such peculiarity for the current task. Indeed, we are not interested in the uncertainty of the model but in metrics that correlate with the probability that a sample is adversarial.

Indeed, also in [56] they observe that the predictive entropy increases at some point when out of distribution, which is enough in order to make an adversarial attack detection, as an adversarial sample can be seen as a cherrypicked out of distribution sample.

We indeed have observed this trend in our experiments, as shown in Figure 2 and Figure 3. Indeed, the posterior $p(Y^*|X^*, X, Y)$ in our formulation is a categorical distribution. The entropy H for a discrete distribution is defined as follows:

$$H[p(x)] = -\sum_{x\in\mathcal{X}} p(x)\log p(x) \tag{11}$$

where $\mathcal{X}$ is the support of $p(x)$. Thanks to the fact that $p(Y)$ is discrete with finite support, we can calculate the maximum entropy in closed form of the unconditional distribution $p(Y)$, which in our case is 2. The mean predictive entropy for various attacks, with increasing epsilon values and for the original data, is depicted in Figure 2. Similarly, the mean mutual information for different attacks, with increasing epsilon values and for the original data, is illustrated in Figure 3. It is evident that the mean predictive entropy and mean mutual information decreases with increasing epsilon values for all attack methods. The predictive entropy and mutual information for the original data remain significantly higher compared to the adversarial examples, indicating that the model is more uncertain when predicting original samples compared to adversarial ones. This gap is more evident in the mutual information case, where we have a sharp decrease in mutual information after the attack with the lowest epsilon. This result indicates that the model demonstrates greater certainty when making predictions on adversarial data compared to original data.

We will generate adversarial attacks, as described in Section IV, against our pre-trained fault zone prediction model. The perturbation magnitudes (ϵ) for these attacks will range from 0.05 to 0.5, incrementing by 0.05. For each input in the test set, the Bayesian LSTM model will be used to obtain predictions and compute the mean predictive entropy and mean mutual information for both the original and adversarial inputs. Our detection mechanism classifies inputs based on their uncertainties to identify adversarial attacks. We will explore various threshold values for classifying these uncertainties to determine if an input is adversarial. The goal is to find the optimal threshold that maximizes overall detection accuracy while maintaining a high accuracy for legitimate data. Specifically, we will classify inputs based on predictive entropy and mutual information. Inputs with predictive entropy or mean mutual information above a certain threshold will be classified as legitimate, while those with predictive entropy and mutual information below the threshold will be classified as adversarial. In our process of identifying the optimal threshold for adversarial

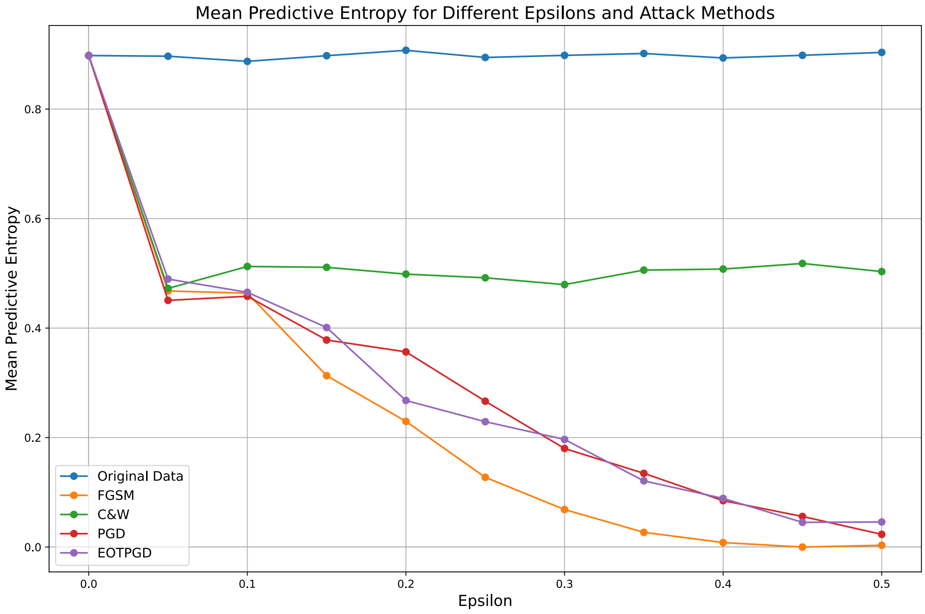


FIGURE 2. Mean predictive entropy for attacks and original data with different epsilons.

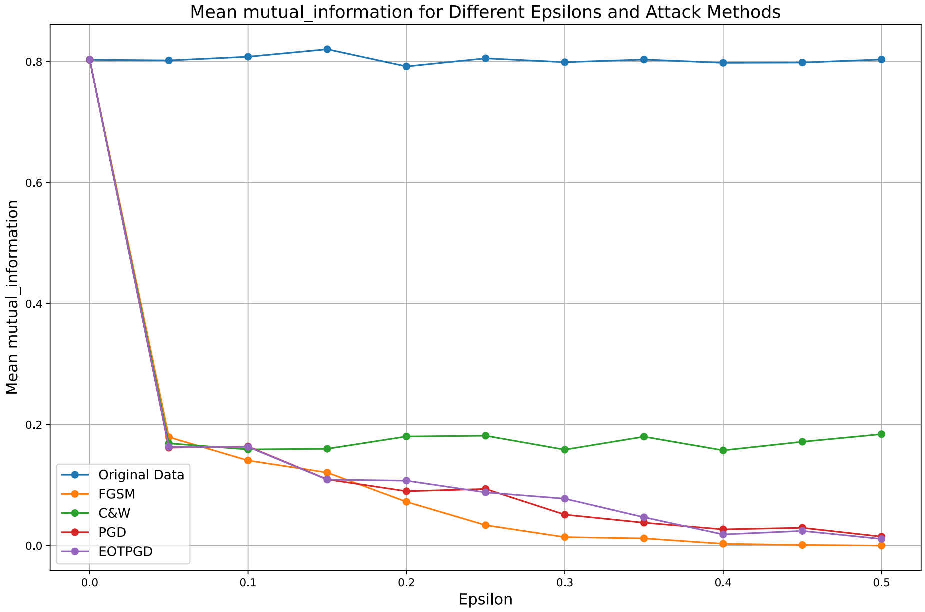


FIGURE 3. Mean mutual information for attacks and original data with different epsilons.

attack detection, we will leverage the predictive entropy and mutual information of the main model on legitimate data. We will employ a grid search method to explore various threshold values, aiming to maximize the overall detection accuracy of adversarial attacks. We will do the same for mutual information. Importantly, while optimizing for adversarial detection, we will ensure that the detection accuracy for legitimate data consistently exceeds a predefined threshold, in this case the threshold is 0.85. This constraint is crucial as we aim to avoid misclassifying legitimate data as adversarial attacks. In summary, we will evaluate the detection mechanism by calculating the mean predictive entropy and mean mutual information for both original and adversarial data. The classification results will be analyzed for each type of attack and for the original data. The optimal threshold will be determined by balancing the trade-off between detecting adversarial samples and maintaining high accuracy for legitimate data.

VI. EVALUATION

We now explore the evaluation of our fault zone prediction system, including various attacks and the uncertainty-based adversarial attack detection mechanism. Our evaluation covers all scenarios detailed in the previous sections. As metrics, we use accuracy, F1 score, precision and recall to evaluate our model, defined as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \tag{12}$$

$$Precision = \frac{TP}{TP + FP}, \tag{13}$$

$$Recall = \frac{TP}{TP + FN}, \tag{14}$$

$$F1 = \frac{2TP}{2TP + FP + FN}. \tag{15}$$

First, we provide details about the dataset used for our evaluation in Section VI-A. To establish a baseline for evaluating the success of our attacks and defenses, we assess our model's performance on the fault zone prediction task in Section VI-B. Next, we evaluate the effectiveness of our attacks in Section VI-C. Finally, in Section VI-D, we examine the capabilities of the uncertainty-based adversarial attack detection mechanism.

A. DATASET

Despite the widespread use of simulation tools such as PSCAD [57], [58], MATLAB Simulink [59], RSCAD [60], and MATPOWER [61] in smart grid failure prediction systems, there is a notable absence of publicly accessible datasets generated by these tools. Therefore, we utilize the dataset introduced by Ardito et al. [46], which is the only publicly available dataset that includes extensive simulated fault data based on the IEEE-13 test node feeder. The IEEE-13 node test feeder features a 4.16 kV voltage generator, 13 buses designed for fault simulation, and facilities for three-phase signal measurement. The distribution system is segmented into four zones to pinpoint fault locations. The dataset was completed by injecting 11 distinct fault types with 22 different resistances for each fault type into four critical zones near the load flow buses 671, 633, 675, and 680. The total duration for fault simulation was $t = [0.0 - 0.02]$ seconds, with each fault and resistance combination applied at $t = 0.01$ seconds and cleared at $t = 0.02$ seconds, resulting in a fault duration of $t_f = [0.01 - 0.02]$ seconds, and a healthy (non-faulty) period of $t_h = [0 - 0.01]$ seconds [46]. This dataset encompasses 51 features, integrating data from both conventional and renewable energy sources. It has been meticulously compiled to provide a benchmark for evaluating the robustness of fault prediction systems in smart electrical grids against adversarial attacks. We split our dataset into three parts: 85% for training, 5% for validation, and 10% for testing. Normalization will be applied as a preprocessing step to scale our data to a consistent range. After normalization, since we are using an LSTM model, we will partition the dataset into windows of a predefined size. These windows are created by sliding through the data iteratively, with a step size equal to half of the window size. For our dataset, we have chosen a window size of 16 seconds.

B. BASELINE EVALUATION

In this section, our focus shifts to evaluating the performance of our fault zone prediction system. Initially, we assess the baseline performance of our system in its pristine state, devoid of any exposure to adversarial attacks. Our training procedure entails harnessing the training data to optimize our Bayesian LSTM-based fault prediction system. Subsequently, we gauge the efficacy of our model on the test set, scrutinizing its ability to predict fault zones accurately. In this evaluation process, we assess the performance of our model using multiple forward passes to estimate the model's uncertainty and robustness. In order to estimate the predictive entropy for the test set samples, for each of them, ten different sets of parameters will be sampled from the learned posterior and used as a Monte Carlo approximation of $p(Y^*|X^*, X, Y)$. This method leverages the Bayesian approach to capture the uncertainty inherent in the model's predictions. We then calculate the test accuracy by comparing these averaged predictions against the true labels. This approach ensures that the model's uncertainty and variability in predictions are adequately captured. Our evaluation results are noteworthy, we observe that our model can reach the mean accuracy of 0.958 in fault zone prediction task. Our approach demonstrates a substantial improvement, with a mean 24.80% increase in performance compared to the main paper introducing the dataset [46]. We have also achieved the same accuracy as the state-of-the-art model [4]. However, the BNN approach provides significant advantages that justify their use. These networks offer robust uncertainty quantification, which is crucial for making more informed decisions and enhancing the detection of adversarial attacks with our uncertainty-based detection. The benefits of our BNN, including reliable uncertainty estimates and improved adversarial robustness through our uncertainty-based detection mechanism, demonstrate their superiority over the state-of-the-art model. The detailed results are presented comprehensively in Table 2.

TABLE 2. Comparison of the model's accuracy, F1 score, precision, and recall.

Model	Accuracy	F1 Score	Precision	Recall
MLP [46]	0.710	NA	NA	NA
Decision Tree [4]	0.818	NA	NA	NA
Random Forest [4]	0.831	NA	NA	NA
XGBoost [4]	0.841	NA	NA	NA
GRU [4]	0.958	NA	NA	NA
Bayesian LSTM	**0.958**	**0.958**	**0.960**	**0.958**

C. ATTACK EVALUATION

In this evaluation, we comprehensively assess the efficacy of white-box attacks on our model. Using the TorchAttacks library [62], we implement attacks mentioned in section IV

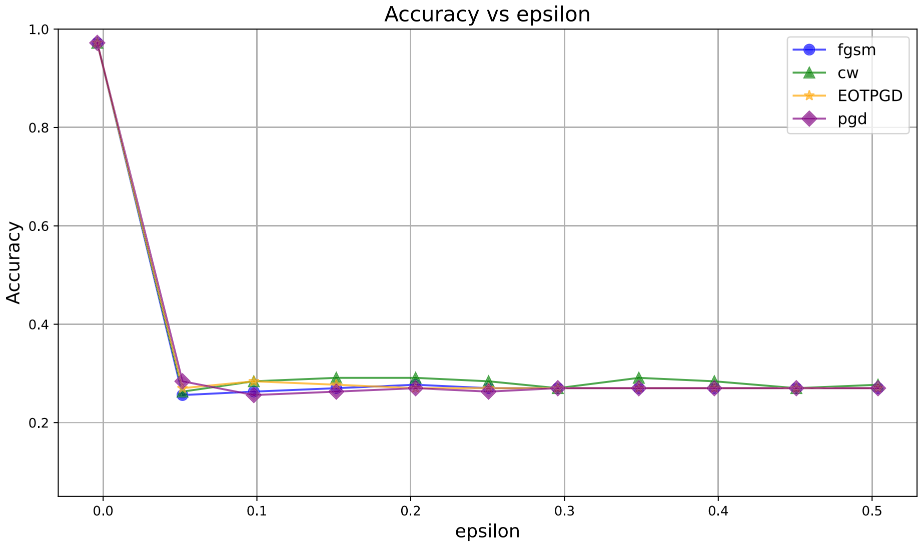


FIGURE 4. Model's accuracy at varying epsilon values on the white-box attacks.

to evaluate the model's susceptibility without incorporating any countermeasures or defenses. However, since the TorchAttacks library is not inherently suited for BNNs, we make necessary modifications. These changes allow the attacks to accept the dual outputs of BNNs, where the second output is the KL divergence, thus enabling effective adversarial testing against our Bayesian model. The attacks are executed with varying epsilon values, representing the strength and degree of perturbation for each attack. Specifically, we explore epsilon values ranging from 0.05 to 0.50. This range helps us understand the model's robustness under different levels of adversarial perturbations. The results of these attacks across various tasks are visually presented in Figure 4, illustrating the impact of each epsilon value on the model's performance. It is evident from the results that even with a low epsilon value of 0.05, the attacks are successful in decreasing the accuracy of the model.

D. UNCERTAINTY-BASED ADVERSARIAL ATTACK DETECTION EVALUATION

In this section, we evaluate the performance of our uncertainty-based adversarial attack detection system by calculating the mean predictive entropy and mean mutual information for both the main model and the adversarial attacks using the test set. To achieve this, we define a range of epsilon values representing different strengths of the adversarial attacks. For each epsilon value, we generate adversarial examples using various attack methods mentioned in section IV. For each set of inputs (original and adversarial), we perform multiple forward passes with different sets of parameters sampled from the learned posterior through the Bayesian LSTM model to obtain samples of the model's outputs. By averaging these outputs, we estimate the predictive distribution. We then compute the predictive entropy and mutual information for each set of outputs to quantify the model's uncertainty.

Finally, we aggregate these uncertainty measures to calculate the mean predictive entropy and mutual information for both the original and adversarial data at each epsilon level. The results can be seen in the figure 2 and 3 already discussed in Section VI-D.

We then implement a classification function f that labels uncertainties below the threshold λ as normal and those above as adversarial.

$$f(X^*) = \begin{cases} \text{adversarial} & \text{if } g(X^*) < \lambda \\ \text{original} & \text{otherwise,} \end{cases}$$

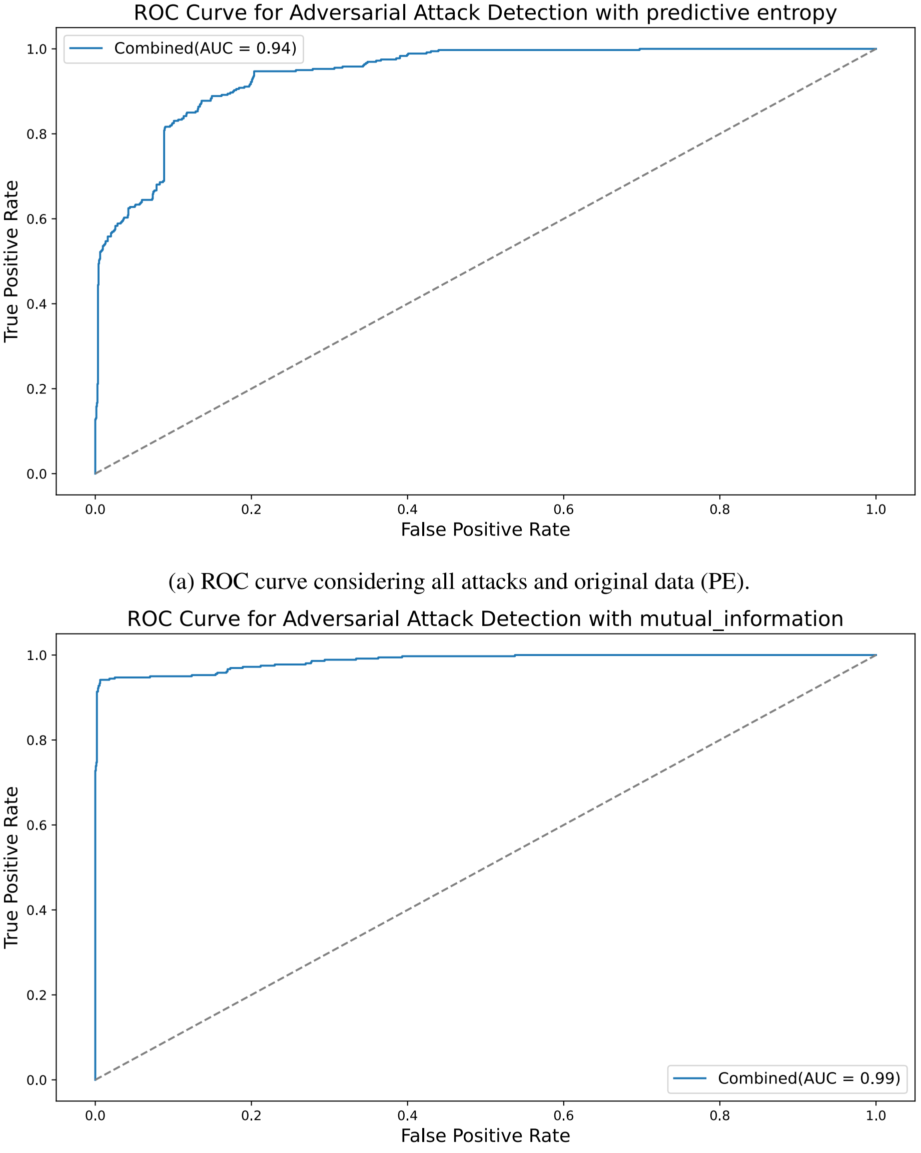
where $g(X^*)$ can be either the predictive entropy, shown in equation 16.

$$H\left[\mathbb{E}_{\omega\sim p(\omega|X,Y)}[p(Y^*|X^*,\omega)]\right] \tag{16}$$

or the epistemic uncertainty, shown in equation 17.

$$I[Y^*;\omega|X^*,X,Y] \tag{17}$$

We explored a range of threshold values to identify the optimal threshold for classifying uncertainties in original and adversarial data across different epsilon values. For each epsilon, we collected uncertainties from the original data and adversarial data generated using various attack methods. By iterating through the defined threshold values, we classified the uncertainties and combined these classifications with the true labels to calculate overall accuracy. We ensured that the accuracy of the original data met a constraint of at least 0.85 before considering the overall accuracy. Finally, using the optimal threshold, which in our case is 0.647 in the case of predictive entropy and 0.499 in the case of mutual information, we perform the final classifications for both original and adversarial data and calculate the accuracy

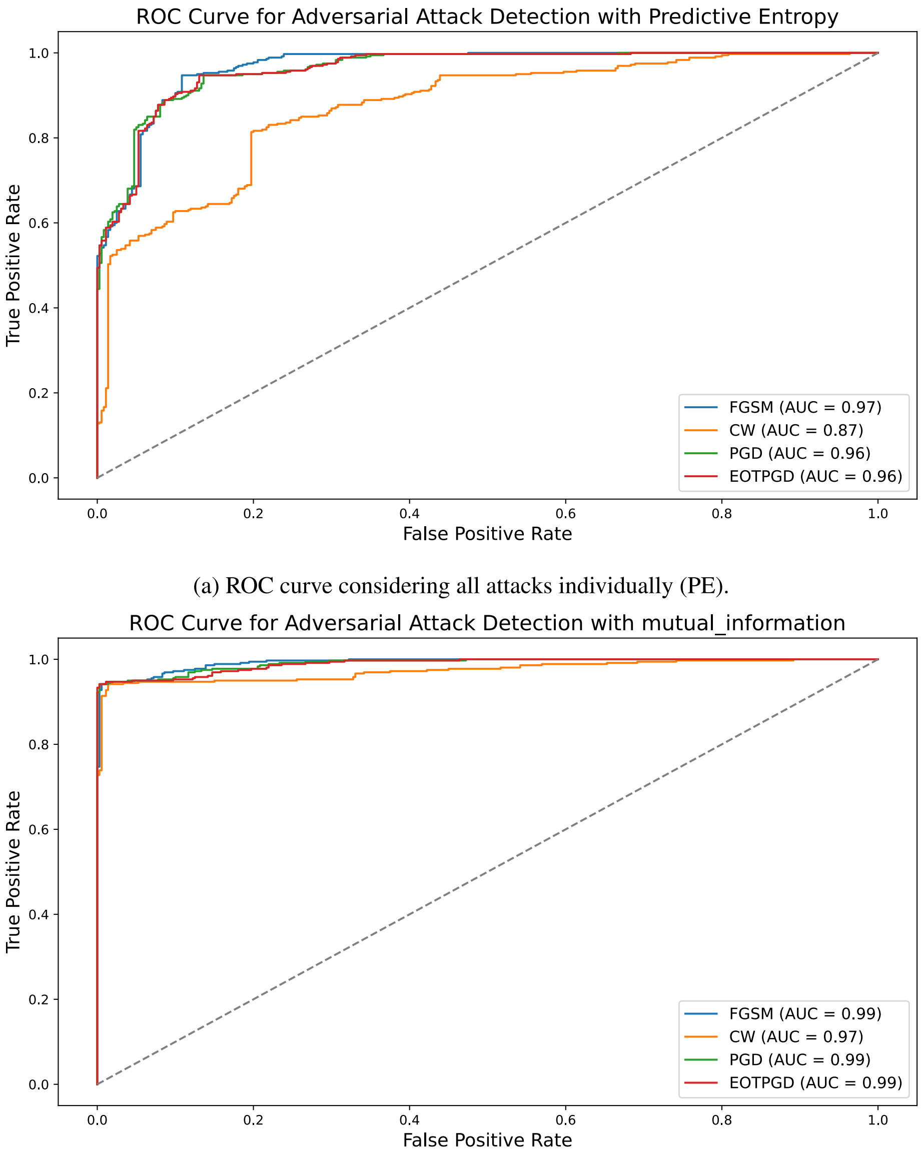


(a) ROC curve considering all attacks and original data (PE).

(b) ROC curve considering all attacks and original data (MI).

FIGURE 5. **ROC curves for different uncertainty metrics.**

for each type of attack, as well as for the original data. The results are noteworthy: our detection system achieves a mean accuracy of 0.891 when using predictive entropy as the uncertainty metric and 0.981 when using mutual information for detecting both adversarial and original data. When focusing solely on adversarial attacks while keeping the constraint of 0.85 accuracy on the original data, the detection accuracy improves to 0.897 with predictive entropy and 0.998 with mutual information. For detecting original data, the accuracy is 0.869 with predictive entropy and 0.913 with mutual information. The results highlight that using mutual information as an uncertainty metric provides superior accuracy compared to predictive entropy. Specifically, results for each attack can be seen in Table 3. To provide a more comprehensive evaluation of our detection scheme, we will utilize ROC (Receiver Operating Characteristic) curve plots. These plots, displayed in Figures 5b, 5a, 6a and 6b, will offer a detailed visualization of the scheme's performance. Considering the low complexity of our detection schemes, the accuracy of the adversarial detection scheme is acceptable across all scenarios. The ability to maintain high accuracy without adding significant computational overhead makes our approach highly practical for real-world applications. This balance of efficiency and effectiveness is crucial for deploying robust security measures in resource-constrained environments like smart grids.



(a) ROC curve considering all attacks individually (PE).

(b) ROC curve considering all attacks individually (MI).

FIGURE 6. **ROC curves considering all attacks individually using different uncertainty metrics.**

TABLE 3. **Mean accuracy of uncertainty-based adversarial detection scheme under different considerations.**

Source	Accuracy (using PE)	Accuracy (using MI)
Attacks + Original data	0.891	0.981
All Attacks	0.897	0.998
Original Data	0.869	0.913
FGSM Attack	0.938	1.0
C&W Attack	0.752	0.994
PGD Attack	0.95	1.0
EOTPGD Attack	0.947	1.0

VII. CONCLUSION

Smart grids emerge as pivotal in modernizing energy infrastructure, promising improved reliability, efficiency, and sustainability. Within smart grids, fault prediction systems play a crucial role in ensuring uninterrupted energy delivery and system reliability. Despite the growing attention to fault prediction systems in the literature, their security aspects are frequently overlooked. This oversight can potentially lead to safety concerns and operational delays, undermining the effectiveness of these systems. Also, neglecting to address misclassification events in these systems could diminish their reliability in real-world scenarios. These misclassifications have the potential to undermine the trustworthiness and effectiveness of these systems, thereby limiting their practical applications in critical tasks like fault zone prediction.

To address these challenges, we propose a BNN framework for fault zone prediction in smart grids, offering quantifiable uncertainty measures to improve decision-making and system robustness. Leveraging Bayesian regularization, our model enhances reliability and quantifies the uncertainty of the predictions, which can help the grid operators in the identification of potential misclassifications, achieving an accuracy of up to 0.958 in fault zone prediction. Furthermore, we introduce an uncertainty-based adversarial attack detection system capable of detecting complex attacks with up to 0.981 accuracy when classifying malicious and original data. Through the evaluation of a publicly available dataset, our study demonstrates the efficacy of our proposed models, attacks, and defenses. By providing a comprehensive framework for fault prediction and adversarial attack detection in smart grids, our work contributes to enhancing the security and reliability of modern energy distribution systems.

A. LIMITATION

Despite the promising results demonstrated in our study, there are several limitations that need to be addressed in future work. One primary limitation is the dataset used for evaluation. Our study relies on a publicly available dataset, which may not fully capture the complexities and variances of real-world smart grid environments. The scarcity of comprehensive, real-world data from actual grid operations limits the ability to generalize our findings across diverse and dynamic grid conditions. The proposed model's reliance on Bayesian regularization to enhance reliability and quantify uncertainty is another aspect that requires further exploration. While Bayesian methods offer theoretical advantages, their computational complexity and scalability to large-scale smart grid systems remain a challenge. Finally, while we achieved high accuracy in both fault zone prediction and adversarial attack detection, the integration of our framework into existing smart grid infrastructure poses practical challenges. Ensuring seamless integration, compatibility with current systems, and minimal disruption during deployment are critical factors that need to be addressed to transition from research to real-world application effectively.

B. FUTURE WORK

Utilizing richer posteriors and ensembles, we aim to enhance the predictive capabilities of our BNN. This involves exploring more complex posterior distributions and ensemble techniques to improve accuracy and reliability in fault zone prediction. Additionally, investigating the relationship between BNNs and adversarial attacks is crucial. Understanding how adversarial perturbations affect uncertainty estimates provided by BNNs can reveal vulnerabilities and guide the development of robustness strategies. Exploring alternatives for zero-order attacks, which rely solely on model output queries without gradient information, presents another different set of challenges and opportunities. Furthermore, leveraging the success of BNNs in fault zone prediction, we propose extending this framework to enhance stability prediction in smart grids. By modeling uncertainty in grid dynamics, BNNs can provide reliable predictions of system stability under varying conditions and disturbances.

APPENDIX

NOMENCLATURE

$+\infty$	Positive infinity.
$-y_t \log p(y \mid x)_t$	Negative log likelihood term for the target variable y_t.
arg max	Argument of the maximum.
ϵ	Perturbation or noise vector.
γ	Constraint threshold for the p-norm of ϵ.
$\int$	Integral sign.
λ	Threshold value used for classifying X^* as adversarial or original.
$\lim_{p(y \mid x)_t \to 0}$	Limit as the probability $p(y \mid x)_t$ approaches 0.
$\log \frac{p(X,Y,\omega)}{q(\omega)}$	Logarithm of the ratio of joint probability $p(X, Y, \omega)$ and distribution $q(\omega)$.
$\log q(\omega)$	Logarithm of the probability distribution $q(\omega)$.
log	Natural logarithm.
$\mathbb{E}_{\omega \sim q(\omega)}$	Expectation with respect to the distribution $q(\omega)$.
$\mathbb{E}_{p(\omega)}$	Expectation with respect to the distribution $p(\omega)$.
$\max_\omega$	Maximization over the parameter ω.
$\max_\epsilon$	Optimization operation to maximize the objective with respect to ϵ.
∇_ω	Gradient with respect to model parameters ω.
ω	Model parameters.
$\sum_t$	Summation index over t.
$\sum_{i=1}^{n}$	Summation over i from 1 to n, where n is the number of samples.
$\sum_{j=1}^{c}$	Summation over j from 1 to c, where c is the number of classes.
$\sum_{x \in X}$	Summation over all possible values of x in the set X.
Accuracy	The ratio of correctly predicted instances to the total instances.
F1	The harmonic mean of precision and recall, providing a single score to evaluate the performance.
FN	False Negatives: The number of incorrectly predicted negative instances.
FP	False Positives: The number of incorrectly predicted positive instances.
KL	Kullback-Leibler divergence.
Precision	The ratio of correctly predicted positive instances to the total predicted positives.

Recall	The ratio of correctly predicted positive instances to the total actual positives.
TN	True Negatives: The number of correctly predicted negative instances.
TP	True Positives: The number of correctly predicted positive instances.
$d\omega$	Differential of ω in the integral.
$f(x+\epsilon)$	Function evaluated at the perturbed input $x+\epsilon$.
$f(X^*)$	Function that classifies X^* as either adversarial or original based on a threshold λ.
$g(X^*)$	Function used to determine the classification of X^*; it can be either predictive entropy or epistemic uncertainty.
$H[E_{\omega\sim p(\omega\mid X,Y)}\cdot$ $[p(Y^*\mid X^*,\omega)]]$	Predictive entropy of Y^* given X^* and ω, with expectation over the distribution $p(\omega\mid X,Y)$.
$H[p(x)]$	Entropy of the probability distribution $p(x)$.
$I[Y^*;\omega\mid X^*,X,Y]$	Epistemic uncertainty or mutual information between Y^* and ω given X^*, X, and Y.
$L(\omega)$	Likelihood function.
$L(\omega)$	Loss function with parameter ω.
$L(\omega)$	Loss function with respect to model parameters ω.
$L(f(x+\epsilon),y)$	Loss function evaluated with perturbed input $x+\epsilon$ and target y.
N	Random variable sampled from distribution $q(\theta)$.
P	Value of the function or quantity of interest.
$p(x)$	Probability of the event x.
$p(y\mid x)_t$	Probability of y given x for index t.
$p_\omega(y_i\mid x_i)_j$	Probability of y_i being in the j-th class given input x_i and parameter ω.
$q(\omega)$	Approximate distribution of ω.
$q(\theta)$	Probability distribution for N.
s'_n	Random variable sampled from uniform distribution $U(-1,+1)$.
s_n	Random variable sampled from uniform distribution $U(-1,+1)$.
T	Another term or variable in the equation.
y_t	Target variable for index t.
$y_{i,j}$	Indicator variable for the j-th class of the i-th sample.
Adversarial Attacks	Attempts to fool a model by providing deceptive input.
Adversarial Data	Data that has been manipulated to deceive a model.
AI	Artificial Intelligence.

ANN	Artificial Neural Network.
AU	Aleatoric Uncertainty.
AUC	Area Under the Curve: The area under the ROC curve, representing the model's ability to distinguish between classes.
BNN	Bayesian Neural Network.
CNN	Convolutional Neural Network.
CW	Carlini & Wagner.
DL	Deep Learning.
ELBO	Evidence Lower Bound.
EOTPGD	Expectation Over Transformation Projected Gradient Descent.
EU	Epistemic Uncertainty.
FGSM	Fast Gradient Sign Method.
GB	Gradient Boosting.
GRU	Bidirectional Gated Recurrent Unit.
KNN	K-nearest Neighbor.
LSTM	Long Short-Term Memory.
ML	Machine Learning.
MLP	Multilayer Perceptron.
PGD	Projected Gradient Descent.
RF	Random Forest.
ROC	Receiver Operating Characteristic
SVM	Support Vector Machine
TU	Total Uncertainty
XGBoost	Extreme Gradient Boost

REFERENCES

[1] R. Bayindir, I. Colak, G. Fulli, and K. Demirtas, "Smart grid technologies and applications," *Renew. Sustain. Energy Rev.*, vol. 66, pp. 499–516, Dec. 2016.

[2] H. A. Muqeet, R. Liaqat, M. Jamil, and A. A. Khan, "A state-of-the-art review of smart energy systems and their management in a smart grid environment," *Energies*, vol. 16, no. 1, p. 472, Jan. 2023.

[3] Forbes. *3 Alarming Threats to the U.S. Energy Grid—Cyber, Physical, and Existential Events*. Accessed: Apr. 20, 2024. [Online]. Available: https://www.forbes.com/sites/chuckbrooks/2023/02/15/3-alarming-threats-to-the-us-energy-grid–cyber-physical-and-existential-events/

[4] E. Efatinasab, F. Marchiori, A. Brighente, M. Rampazzo, and M. Conti, "FaultGuard: A generative approach to resilient fault prediction in smart electrical grids," in *Detection of Intrusions and Malware, and Vulnerability Assessment*, F. Maggi, M. Egele, M. Payer, and M. Carminati, Eds., Cham, Switzerland: Springer, 1007, pp. 503–524.

[5] M. N. Nafees, N. Saxena, A. Cardenas, S. Grijalva, and P. Burnap, "Smart grid cyber-physical situational awareness of complex operational technology attacks: A review," *ACM Comput. Surv.*, vol. 55, no. 10, pp. 1–36, Oct. 2023.

[6] E. Efatinasab, A. Brighente, M. Rampazzo, N. Azadi, and M. Conti, "GAN-GRID: A novel generative attack on smart grid stability prediction," 2024, *arXiv:2405.12076*.

[7] Y. He, G. J. Mendis, and J. Wei, "Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2505–2516, Sep. 2017.

[8] C. A. Andresen, B. N. Torsæter, H. Haugdal, and K. Uhlen, "Fault detection and prediction in smart grids," in *Proc. IEEE 9th Int. Workshop Appl. Meas. for Power Syst. (AMPS)*, Sep. 2018, pp. 1–6.

[9] K. Chen, C. Huang, and J. He, "Fault detection, classification and location for transmission lines and distribution systems: A review on the methods," *High Voltage*, vol. 1, no. 1, pp. 25–33, Apr. 2016.

[10] J. De La Cruz, E. Gómez-Luna, M. Ali, J. C. Vasquez, and J. M. Guerrero, "Fault location for distribution smart grids: Literature overview, challenges, solutions, and future trends," *Energies*, vol. 16, no. 5, p. 2280, Feb. 2023.

[11] N. Hussain, M. Nasir, J. C. Vasquez, and J. M. Guerrero, "Recent developments and challenges on AC microgrids fault detection and protection systems—A review," *Energies*, vol. 13, no. 9, p. 2149, May 2020.

[12] M. M. A. Mahfouz and M. A. H. El-Sayed, "Smart grid fault detection and classification with multi-distributed generation based on current signals approach," *IET Gener., Transmiss. Distrib.*, vol. 10, no. 16, pp. 4040–4047, Dec. 2016.

[13] M. Mousa, S. Abdelwahed, and J. Klüss, "Review of diverse types of fault, their impacts, and their solutions in smart grid," in *Proc. SoutheastCon*, Apr. 2019, pp. 1–7.

[14] S. Rahman Fahim, S. K. Sarker, S. M. Muyeen, M. R. I. Sheikh, and S. K. Das, "Microgrid fault detection and classification: Machine learning based approach, comparison, and reviews," *Energies*, vol. 13, no. 13, p. 3460, Jul. 2020.

[15] P. Stefanidou-Voziki, N. Sapountzoglou, B. Raison, and J. L. Dominguez-Garcia, "A review of fault location and classification methods in distribution grids," *Electr. Power Syst. Res.*, vol. 209, Aug. 2022, Art. no. 108031.

[16] A. Zidan, M. Khairalla, A. M. Abdrabou, T. Khalifa, K. Shaban, A. Abdrabou, R. El Shatshat, and A. M. Gaouda, "Fault detection, isolation, and service restoration in distribution systems: State-of-the-art and future trends," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2170–2185, Sep. 2017.

[17] L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, "Hands-on Bayesian neural networks—A tutorial for deep learning users," *IEEE Comput. Intell. Mag.*, vol. 17, no. 2, pp. 29–48, May 2022.

[18] J. Lampinen and A. Vehtari, "Bayesian approach for neural networks—Review and case studies," *Neural Netw.*, vol. 14, no. 3, pp. 257–274, Apr. 2001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608000000988

[19] P. Izmailov, S. Vikram, M. D. Hoffman, and A. G. G. Wilson, "What are Bayesian neural network posteriors really like?" in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 4629–4640.

[20] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[21] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 229–256, May 1992.

[22] C. Zhang, J. Bütepage, H. Kjellström, and S. Mandt, "Advances in variational inference," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 2008–2026, Aug. 2019.

[23] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1613–1622.

[24] Y. Pang, S. Cheng, J. Hu, and Y. Liu, "Evaluating the robustness of Bayesian neural networks against different types of attacks," 2021, *arXiv:2106.09223*.

[25] Y. Wen, P. Vicol, J. Ba, D. Tran, and R. Grosse, "Flipout: Efficient pseudo-independent weight perturbations on mini-batches," 2018, *arXiv:1803.04386*.

[26] M. M. Saha, J. J. Izykowski, and E. Rosolowski, *Fault Location on Power Networks*. London, U.K.: Springer-Verlag, 2009.

[27] M. A. Al-shaher, M. M. Sabry, and A. S. Saleh, "Fault location in multi-ring distribution network using artificial neural network," *Electr. Power Syst. Res.*, vol. 64, no. 2, pp. 87–92, Feb. 2003.

[28] Y. Aslan, "An alternative approach to fault location on power distribution feeders with embedded remote-end power generation using artificial neural networks," *Electr. Eng.*, vol. 94, no. 3, pp. 125–134, Sep. 2012.

[29] J. Coser, D. T. do Vale, and J. G. Rolim, "Design and training of artificial neural networks for locating low current faults in distribution systems," in *Proc. Int. Conf. Intell. Syst. Appl. Power Syst.*, Nov. 2007, pp. 1–6.

[30] F. Dehghani, F. Khodnia, and E. Dehghan, "Fault location of unbalanced power distribution feeder with distributed generation using neural networks," *CIRED, Open Access Proc. J.*, vol. 2017, no. 1, pp. 1134–1137, Oct. 2017.

[31] P. E. Farias, A. P. de Morais, J. P. Rossini, and G. Cardoso, "Non-linear high impedance fault distance estimation in power distribution systems: A continually online-trained neural network approach," *Electr. Power Syst. Res.*, vol. 157, pp. 20–28, Apr. 2018.

[32] S. Javadian, A. Nasrabadi, M.-R. Haghifam, and J. Rezvantalab, "Determining fault's type and accurate location in distribution systems with DG using MLP neural networks," in *Proc. Int. Conf. Clean Electr. Power*, 2009, pp. 284–289.

[33] J. C. S. Souza, M. A. P. Rodrigues, M. T. Schilling, and M. B. D. C. Filho, "Fault location in electrical power systems using intelligent systems techniques," *IEEE Trans. Power Del.*, vol. 16, no. 1, pp. 59–67, Jan. 2001.

[34] M. U. Usman, J. Ospina, and Md. O. Faruque, "Fault classification and location identification in a smart distribution network using ANN," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, Aug. 2018, pp. 1–6.

[35] D. Thukaram, H. P. Khincha, and H. P. Vijaynarasimha, "Artificial neural network and support vector machine approach for locating faults in radial distribution systems," *IEEE Trans. Power Del.*, vol. 20, no. 2, pp. 710–721, Apr. 2005.

[36] M. R. Shadi, M.-T. Ameli, and S. Azad, "A real-time hierarchical framework for fault detection, classification, and location in power systems using PMUs data and deep learning," *Int. J. Electr. Power Energy Syst.*, vol. 134, Jan. 2022, Art. no. 107399.

[37] B. Bhattacharya and A. Sinha, "Intelligent fault analysis in electrical power grids," in *Proc. IEEE 29th Int. Conf. Tools Artif. Intell. (ICTAI)*, Nov. 2017, pp. 985–990.

[38] F. Zhang, Q. Liu, Y. Liu, N. Tong, S. Chen, and C. Zhang, "Novel fault location method for power systems based on attention mechanism and double structure GRU neural network," *IEEE Access*, vol. 8, pp. 75237–75248, 2020.

[39] H. Okumus and F. M. Nuroglu, "A random forest-based approach for fault location detection in distribution systems," *Electr. Eng.*, vol. 103, no. 1, pp. 257–264, Feb. 2021.

[40] Z. El Mrabet, N. Sugunaraj, P. Ranganathan, and S. Abhyankar, "Random forest regressor-based approach for detecting fault location and duration in power systems," *Sensors*, vol. 22, no. 2, p. 458, Jan. 2022.

[41] F. Wilches-Bernal, M. Jiménez-Aparicio, and M. J. Reno, "An algorithm for fast fault location and classification based on mathematical morphology and machine learning," in *Proc. IEEE Power Energy Soc. Innov. Smart Grid Technol. Conf. (ISGT)*, Apr. 2022, pp. 1–5.

[42] A. Ghaemi, A. Safari, H. Afsharirad, and H. Shayeghi, "Accuracy enhance of fault classification and location in a smart distribution network based on stacked ensemble learning," *Electric Power Syst. Res.*, vol. 205, Apr. 2022, Art. no. 107766.

[43] Z. Yongli, H. Limin, and L. Jinling, "Bayesian networks-based approach for power systems fault diagnosis," *IEEE Trans. Power Del.*, vol. 21, no. 2, pp. 634–639, Apr. 2006.

[44] M. Majidi, A. Arabali, and M. Etezadi-Amoli, "Fault location in distribution networks by compressive sensing," *IEEE Trans. Power Del.*, vol. 30, no. 4, pp. 1761–1769, Aug. 2015.

[45] N. Sapountzoglou, J. Lago, and B. Raison, "Fault diagnosis in low voltage smart distribution grids using gradient boosting trees," *Electr. Power Syst. Res.*, vol. 182, May 2020, Art. no. 106254.

[46] C. Ardito, Y. Deldjoo, T. Di Noia, E. Di Sciascio, and F. Nazary, "IEEE13-AdvAttack a novel dataset for benchmarking the power of adversarial attacks against fault prediction systems in smart electrical grid," in *Proc. 31st ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2022, pp. 3817–3821.

[47] T. M. Chen and S. Abu-Nimeh, "Lessons from stuxnet," *Computer*, vol. 44, no. 4, pp. 91–93, Apr. 2011.

[48] J. E. Sullivan and D. Kamensky, "How cyber-attacks in Ukraine show the vulnerability of the U.S. power grid," *Electr. J.*, vol. 30, no. 3, pp. 30–35, Apr. 2017.

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

[50] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Proc. IEEE Symp. Secur. Privacy (SP)*. Los Alamitos, CA, USA: IEEE Comput. Soc. Press, May 2017, pp. 39–57.

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

[52] R. S. Zimmermann, "Comment on 'Adv-BNN: Improved adversarial defense through robust Bayesian neural network,'" 2019, *arXiv:1907.00895*.

[53] R. Krishnan, J.-L. Lin, M. Beale, M. Subedar, and P. E. J. van Amersfoort, ''IntelLabs/Bayesian-torch: Bayesian-torch 0.5.0,'' Zenodo, IntelLabs, Hillsboro, OR, USA, Jan. 2024, doi: 10.5281/zenodo.10452824.

[54] M. Noroozi, H. Mohammadi, E. Efatinasab, A. Lashgari, M. Eslami, and B. Khan, ''Golden search optimization algorithm,'' *IEEE Access*, vol. 10, pp. 37515–37532, 2022.

[55] S. Fort, H. Hu, and B. Lakshminarayanan, ''Deep ensembles: A loss landscape perspective,'' 2019, *arXiv:1912.02757*.

[56] L. Smith and Y. Gal, ''Understanding measures of uncertainty for adversarial example detection,'' 2018, *arXiv:1803.08533*.

[57] S. Chakraborty and S. Das, ''Application of smart meters in high impedance fault detection on distribution systems,'' *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 3465–3473, May 2019.

[58] H. Jiang, J. J. Zhang, W. Gao, and Z. Wu, ''Fault detection, identification, and location in smart grid based on data-driven computational methods,'' *IEEE Trans. Smart Grid*, vol. 5, no. 6, pp. 2947–2956, Nov. 2014.

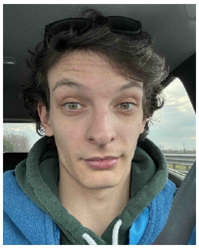
[59] X. Wang, J. Gao, X. Wei, G. Song, L. Wu, J. Liu, Z. Zeng, and M. Kheshti, ''High impedance fault detection method based on variational mode decomposition and Teager–Kaiser energy operators for distribution network,'' *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6041–6054, Nov. 2019.

[60] M. Shafiullah and M. A. Abido, ''S-transform based FFNN approach for distribution grids fault detection and classification,'' *IEEE Access*, vol. 6, pp. 8080–8088, 2018.

[61] M. He and J. Zhang, ''A dependency graph approach for fault detection and localization towards secure smart grid,'' *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 342–351, Jun. 2011.

[62] H. Kim, ''Torchattacks: A PyTorch repository for adversarial attacks,'' 2020, *arXiv:2010.01950*.

EMAD EFATINASAB received the B.Sc. degree in computer engineering from the University of Birjand, in 2021, and the M.Sc. degree in computer science from the University of Padova, in 2023, where he is currently pursuing the Ph.D. degree in information engineering. His current research interests include security, energy informatics, generative networks, adversarial machine learning, and optimization.



ALBERTO SINIGAGLIA received the B.Sc. degree (Hons.) in computer science and the M.Sc. degree (Hons.) in data science from the University of Padova, Italy, in 2021 and 2023, respectively, where he is currently pursuing the Ph.D. degree. His research interests include deep reinforcement learning, multi-agent reinforcement learning, fairness in machine learning, adversarial machine learning, and Bayesian machine learning.

NAHAL AZADI received the B.Sc. degree in computer engineering from the University of Birjand, in 2022. She is currently pursuing the M.Sc. degree in ICT for internet and multimedia with the University of Padova. Her current research interests include cybersecurity, computer networks, computer vision, and machine learning.

GIAN ANTONIO SUSTO (Senior Member, IEEE) received the M.S. degree (cum laude) in control systems engineering and the Ph.D. degree in information engineering from the University of Padova, Padua, Italy, in 2009 and 2013, respectively. He is currently an Associate Professor with the University of Padova. He held visiting positions at the University of California, San Diego; the National University of Ireland, Maynooth; and Infineon Technologies Austria AG, Villach, Austria. He is also one of the co-founders of the Machine Learning Company Statwolf. He is the author of more than 150 publications in peer-reviewed venues. During his career, he has been awarded with four best conference papers awards by various IEEE societies.

MIRCO RAMPAZZO (Member, IEEE) received the Laurea degree in electrical engineering and the Ph.D. degree in information and communication science and technologies from the University of Padova, Padua, Italy, in 2005 and 2010, respectively. In 2010, he joined the Department of Information Engineering, University of Padova, where he is currently an Associate Professor of control systems engineering. In addition, he has held visiting positions at Luleå University of Technology and ICE RISE SICS North Research Institute, Luleå, Sweden. He is actively involved in research projects, collaborating with both academic and industrial partners. He has authored several publications in journals and conference proceedings. He is a co-inventor of patents related to the use of advanced control techniques in various application areas. His research interests include advanced control applications with special emphasis on multiphysics systems and industrial processes. He is currently a member of the IFAC Industry Committee. In 2019, he received the IFAC-ACD Best Industry Paper Award.

Open Access funding provided by 'Università degli Studi di Padova' within the CRUI CARE Agreement