QNAD: Quantum Noise Injection for Adversarial Defense in Deep Neural Networks

Shamik Kundu*‡, Navnil Choudhury*‡, Sanjay Das*, Arnab Raha†, and Kanad Basu*
*University of Texas at Dallas, TX, †Intel Corporation, Santa Clara, CA

***Abstract*—Deep learning in quantum computing seeks to leverage the unique properties of quantum systems, such as superposition and entanglement, to enhance the performance of deep learning algorithms. Quantum neural networks (QNNs), which are designed to operate on quantum computers, have the potential to enable faster and more efficient inference execution. However, quantum computers are susceptible to noise, which can rapidly degrade the coherence of quantum states and lead to errors in quantum computations. As a result, deep neural networks (DNNs) that operate on quantum computers may experience degraded classification accuracy during inference. However, in this paper, we demonstrate that this intrinsic quantum noise can actually improve the robustness of DNNs against adversarial input attacks. The noisy behavior of quantum computers can reduce the impact of adversarial attacks, thereby improving the accuracy of the degraded DNNs. To further enhance DNN robustness, we perform am extensive exploration on the prowess of Quantum Noise injection for Adversarial Defense (QNAD), which induces carefully crafted crosstalk in the quantum computer. QNAD preselects a subset of pretrained network weights to be perturbed with injected crosstalk in the qubits, causing them to become entangled due to interactions between neighboring qubits. When evaluated on state-of-the-art network dataset configurations, the proposed QNAD approach provides up to 268% relative improvement in accuracy, against adversarial input attacks compared to conventional DNN implementations.**

***Index Terms*—Quantum Computing, Quantum Machine Learning, NISQ, Adversarial Attack, Deep Neural Networks.**

I. Introduction

Deep learning [1] has emerged as a powerful tool for solving complex problems in various domains, such as computer vision [2], natural language processing [3], and recommendation systems [4]. One of the key strengths of deep learning is its ability to automatically learn representations of data through hierarchical layers of neural networks. However, this comes at a significant computational cost. Deep learning models typically require large amounts of data for training, as well as high-performance computing resources to efficiently process the massive amounts of data. This has led researchers to explore the potential of quantum computers in accelerating the execution of deep neural networks (DNNs).

Quantum computers are computing systems that use quantum mechanics principles to perform computations [5]. They have the potential to solve certain problems with improved efficiency, that are difficult or impossible for classical computers to solve. For example, Shor's algorithm which factors numbers into their prime factors, leverages the principles of quantum computing to furnish tremendous speedup over its classical counterpart. Deep learning on quantum computers can also benefit from the quantum properties of superposition and entanglement to process information in parallel and perform computations more efficiently than classical computers [6]. One of the most promising approaches in this direction is the quantum neural network (QNN). QNNs are quantum analogues of classical neural networks and are designed to operate on quantum data. They use quantum gates to process quantum states and train quantum circuits to perform various tasks such as classification, regression, and clustering. QNNs have shown promise in applications such as quantum image processing and quantum machine learning [7].

However, quantum computers are susceptible to intrinsic noise, arising due to various physical sources of interference, such as temperature fluctuations, magnetic field noise, and imperfections in the hardware components [8], [9]. Quantum noise may lead to errors in quantum computations, which can cause deviations from the ideal output and subvert the accuracy of quantum algorithms. The impact of quantum noise is particularly significant for deep learning algorithms that rely on high-precision calculations and require a large number of qubits to perform complex computations. Noise can cause instability in quantum states, leading to erroneous calculations in deep learning models. As a result, deep learning algorithms, when executed on these noisy quantum computers suffer degradation in classification accuracy during inference.

On the flip side, even though noisy quantum computers furnish sub-par DNN performance, it can potentially lead to adversarially robust DNN implementations. Vulnerability of DNNs against adversarial attacks has been an important security challenge in classicial deep learning. Adversarial images are generated by estimating the gradients of the DNN with respect to its input, and carefully perturbing the images in the direction of maximum change in the classifier output. Prior works have shown that the performance of DNNs can be severely degraded by modifying the inputs of DNNs by a small amount using adversarial algorithms [10], [11]. Several recent works have proposed noise-injection techniques to defend against adversarial attacks [12]–[15]. Parametric noise injection involves trainable Gaussian noise into the activations or weights of each DNN layer to improve the adversarial robustness [12]. Furthermore, existing research have employed synthetic or Gaussian noise in In-Memory Computing (IMC)

‡Authors have equal contribution.
This research is supported by NSF grant #2228725.
Corresponding Author: Shamik Kundu (*shamik.kundu@utdallas.edu*).

979-8-3503-7994-3/24/$31.00 ©2024 IEEE

architectures to perturb the activations/weights of DNNs to improve robustness [16], [17]. However, improving adversarial robustness of a DNN involving quantum hardware has not yet been explored.

In this work, we propose QNAD (Quantum Noise Injection for Adversarial Defense), an extensive exploration that investigates the prowess of hardware noise from quantum computers towards enhancing the robustness of DNNs against adversarial input attacks. QNAD mapped the multiply-accumulate operations in convolution and fully-connected layers of pre-trained DNN models with quantum hardware designs for inference, and investigated the impact of existing intrinsic noise profiles predominant in state-of-the-art quantum computers on the adversarial robustness of the DNN. In order to further bolster the adversarial defense, QNAD leverages the inherent properties of quantum computing to introduce crosstalk in the qubits, that in turn modifies the behavior of the DNN. QNAD selects a specific subset of pre-trained network weights using a gradient-based approach, that has the highest impact on the output inference prediction of the DNN. The injection of crosstalk in the qubits causes them to become entangled through interactions with neighboring qubits, thereby perturbing those highly important weights to counteract the impact of adversarial attacks on the DNN. To the best of our knowledge, this is the first work, that leverages quantum hardware to bolster adversarial robustness in a DNN. The proposed QNAD is flexible to be applied to various quantum architectures (utilizing superconducting qubits) and DNN models. A point to note here is that, QNAD does not aim to supplant existing defense strategies against adversarial attacks. Instead, QNAD, for the first time ever, performs an extensive exploration to demonstrate the inherent capability of quantum circuit noise to enhance the resilience of DNN executions. This enhancement can serve as a supplementary approach alongside existing adversarial defense techniques like adversarial training and regularization, thereby adding an extra layer of security. We make the following key contributions in this paper:

- In this paper, we, for the first time, demonstrate that intrinsic hardware noise in quantum circuits can improve the robustness of DNNs against adversarial input attacks.
- We propose Quantum Noise injection study for Adversarial Defense (QNAD), which injects regulated crosstalk in the quantum computer to further improve the adversarial robustness of the DNN. QNAD pre-selects a subset of pretrained network weights to be perturbed with injected crosstalk in the qubits, causing them to become entangled due to interactions between neighboring qubits.
- When evaluated on state-of-the-art network-dataset configurations, our proposed QNAD approach achieves up to 183% relative improvement in the classification accuracy under adversarial attack, when exposed to intrinsic quantum noise in the circuit. We also demonstrate that introducing crosstalk into a conventionally trained DNN during inference leads to upto 268% relative improvement in accuracy of the DNN.

The rest of the paper is organized as follows. Section II presents background information on deep learning in quantum computing, quantum noise, and adversarial attacks and defense. The attack model, along with the proposed QNAD approach is delineated in Section III. Section IV demonstrates the prowess of the QNAD methodology in defending adversarial attacks. Finally, Section V concludes the paper.

II. Background and Related Work

A. Deep Learning in Quantum Computing

An area of active research involves adapting classical machine learning techniques for use in quantum systems. To this end, several architectures have emerged, including Quantum Convolutional Neural Networks, QuantumFlow , QuGAN, and QuClassi. Quantum Convolutional Neural Networks (QCNN) adapt the classical concept of spatial data encoding to quantum machine learning techniques [18]. QCNN uses dual qubit unitaries and mid-circuit measurement to perform information down-pooling, allowing for decision-making. This contrasts with an opposite direction traversal of a MERA network [19]. QuClassi proposes a state-based detection scheme inspired by classical machine learning methods [20]. It trains"weights" to represent classifier states, where each state represents a probability of belonging to a specific class. The resulting output layers are similar to those produced by classical classification machine learning networks.

QuantumFlow seeks to replicate the transformations that occur in classical neural networks and achieve a similar transformation as the classical equation, $y = f(x^T w + b)$ [21]. It uses phase flips, accumulation via a Hadamard gate, and an entanglement operation to achieve this transformation. QuantumFlow demonstrates the benefits of batch normalization, revealing significant performance improvements when normalizing quantum data to reside around the XY plane instead of clustering around either the $|1\rangle$ or $|0\rangle$ point. Additionally, QuantumFlow illustrates the reduced parameter potential of quantum machine learning, highlighting a quantum advantage.

Recently, researchers have explored the use of a quantum convolutional network for high energy physics data analysis [20]. A framework for encoding localized classical data is presented, followed by a fully entangled parameterized layer for spatial data analysis. The promise of quantum convolutional networks is demonstrated by their numerical analysis. It is notable that all of these works have taken a classical machine learning technique and modified it to adapt to the quantum setting in different forms.

Despite such progress, deep learning algorithms, being executed on quantum computers are susceptible to noise, that leads to degradation in inference classification accuracy of the network. Quantum circuits can be affected by various types of intrinsic noise, such as thermal noise and shot noise. Thermal noise arises due to fluctuations in the temperature of the system, leading to random fluctuations in the energy levels of the qubits. Shot noise, on the other hand, arises due to the probabilistic nature of quantum measurements, leading to random fluctuations in the number of particles passing through the system. Both types of noise can cause errors in quantum

gate operations and can limit the accuracy and reliability of quantum computations. Not only noise, but these quantum computers are also prone to crosstalk. In the context of quantum computing, crosstalk refers to unwanted interactions between qubits in a quantum circuit. These interactions can arise due to coupling between different qubits, resulting in the leakage of information from one qubit to another. Crosstalk can lead to significant errors in quantum gate operations and can limit the scalability of quantum computing systems. In this paper, we leverage this inherent noise in quantum hardware, coupled with regulated crosstalk to improve the adversarial robustness in deep learning algorithms.

B. Adversarial Attacks and Defenses

Adversarial attacks and defenses are a growing area of research in the field of machine learning. An adversarial attack is an attempt to manipulate input data in order to cause a machine learning model to produce incorrect or misleading results. This can be achieved in a variety of ways, including by adding noise to the input, modifying individual features of the input, or by crafting adversarial examples that are specifically designed to deceive the model. Adversarial attacks have been shown to be effective against a wide range of machine learning models, including deep neural networks, support vector machines, and decision trees.

One of the most common types of adversarial attacks is the so-called "Fast Gradient Sign Method (FGSM)" attack. This involves adding a small amount of noise to an input image, in order to cause a deep neural network to misclassify the image. The attack is relatively simple to implement, but can be highly effective. For example, Goodfellow et al. demonstrated that an FGSM attack could be used to cause a state-of-the-art deep neural network to misclassify an image of a panda as a gibbon, simply by adding a small amount of noise to the input [11]. Similarly, effective adversarial samples can be generated by utilizing a multi-step iterative optimization-based method (unlike "FGSM"), which is known as "Projected Gradient Descent (PGD)" [22].

Adversarial attacks are a growing concern because they can have serious real-world consequences. For example, an attacker could use an adversarial attack to cause a self-driving car to misclassify a stop sign as a yield sign, potentially leading to a dangerous traffic situation. Similarly, an attacker could use an adversarial attack to manipulate medical images, leading to incorrect diagnoses and potentially harming patients. Because of these potential consequences, there is a growing need for effective defenses against adversarial attacks.

There are a number of different strategies that can be used to defend against adversarial attacks. One approach is to use adversarial training, in which the machine learning model is trained using both clean and adversarial examples. This can improve the model's ability to resist adversarial attacks, by making it more robust to small perturbations in the input. For example, Madry et al. demonstrated that adversarial training could be used to improve the robustness of deep neural networks against a range of adversarial attacks [10].

Another approach is to use input preprocessing techniques, such as denoising or randomization, to make it more difficult for attackers to craft adversarial examples. For example, Xie et al. proposed a method called "randomization smoothing", in which the input is randomized using a stochastic function before being fed into the machine learning model [23]. This makes it more difficult for attackers to craft targeted adversarial examples, because the exact input that will be fed into the model is not known in advance. Recently, a method to improve the robustness of deep neural networks against adversarial attacks has been proposed by injecting parametric noise during training [12]. The proposed method, called "Parametric Noise Injection," adds random noise to the input data of the neural network in a controllable and trainable manner. This technique helps to improve the performance of the neural network in the presence of adversarial attacks, while also improving its accuracy on clean data.

Adversarial attacks and defenses have important implications for a wide range of mission critical applications, and hence, an active area of research. To the best of our knowledge, this is the first paper to utilize the noise in quantum hardware as a means of enhancing the adversarial robustness of deep learning algorithms, thereby improving the safety and security of such systems.

III. Proposed QNAD Approach

In this section, we first delineate our adversarial attack model. Next, we outline our defense against such attacks by leveraging the effect of inherent noise present in quantum circuits. We proceed to examine the effects of the inherent noise present in the quantum circuits and its effects on the robustness of the mapped quantum neural network model (QNN). This is accomplished under the presumption of a preexisting mapping strategy, which possesses the capability to effectuate the translation of a neural-network architecture into a gate-based quantum circuit. An example of such a mapping strategy is denoted in Figure 1.

A. Attack Model

In this section, we define a threat model where the adversary has a knowledge of the target DNN architecture. We define a DNN model, $f_\theta : \chi \rightarrow \mathbb{R}^k$, where, θ represents the model parameters, χ is the input space and k is the number of output classes. Let $x_i \in \chi$ be an input instance, $y_i \in \mathbb{R}^k$ be the true label for that input and α be the adversarial perturbation. The perturbation α is usually constrained to belong to an allowable set Δ. We call the perturbed input $\hat{x} = x + \alpha$, an adversarial example if it satisfies the relation $f_\theta(x)_i \neq (x_i + \alpha), \alpha \in \Delta$. Now, if $l(f_\theta(x_i), y_i)$ is a loss function, the adversarial perturbation α can be computed by solving the optimization problem $max_{(\alpha \in \Delta)} l(f_\theta(x + \alpha), y)$. Various adversarial attacks can be constructed by modifying the optimization method as well as the constraint Δ. Two of the most extensively studied and effective methods for generating adversarial examples, utilized in this paper are Fast Gradient Sign Method (FGSM) [11] and Projected Gradient Descent (PGD) [22].

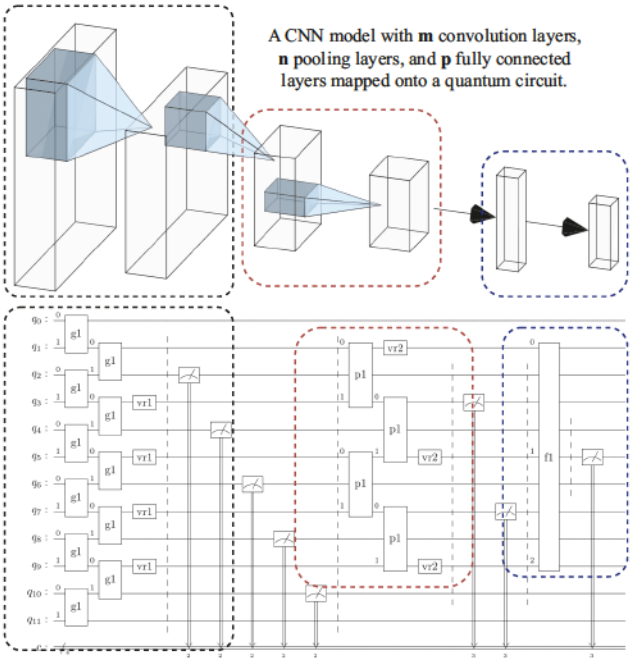

Fig. 1: A potential mapping of the layers of a traditional CNN model onto a quantum circuit.

FGSM Method: In this paper, we first utilize the FGSM method to generate adversarial attacks. FGSM is a popular and straightforward method for generating adversarial examples. It leverages the gradient information of the loss function with respect to the input to craft perturbations that can lead to misclassifications or incorrect predictions. FGSM is known for its simplicity and efficiency, making it a widely used technique for adversarial attacks. The key idea behind FGSM is to compute the gradient of the loss function with respect to the input data and then perturb the input by taking a small step in the direction of the sign of the gradient. By using the sign of the gradient, FGSM determines the direction that increases the loss function the most, allowing for targeted perturbations towards a specific class or untargeted perturbations to induce misclassification. The equation for calculating the adversarial attack can be represented as :

$$\hat{x} = x + \epsilon * sign(\nabla l(\theta, x, y)) \tag{1}$$

where, ϵ is a small number that limits the amount of perturbation, ∇ is the gradient of the loss function with respect to input x. This method adds a perturbation whose direction is the same as the gradient of the cost function, thereby increasing the value of the cost function. This method is invariant of the magnitude of the gradient. The simplicity and efficiency of FGSM make it an attractive choice for adversaries seeking to generate adversarial examples quickly. It requires only a single forward and backward pass through the network to compute the gradient, making it computationally inexpensive compared to more iterative methods like PGD.

PGD Method: Apart from the FGSM method, we utilize the Projected Gradient Descent (PGD) method, which is an iterative optimization-based method for generating adversarial examples. It aims to find the perturbation that maximizes the loss function while staying within a predefined epsilon-bound region around the original input. The basic idea behind PGD is to perform multiple iterations of gradient ascent on the loss function with respect to the input, while projecting the perturbed input back onto the allowed region at each iteration. The number of iterations is a crucial parameter in PGD. More iterations generally lead to better approximations of the optimal perturbation but also increase the computational cost. In each iteration of PGD, the gradient of the loss function with respect to the input is computed. This gradient represents the direction in which the loss function increases the most. The perturbed input is then updated by taking a small step in the direction of the gradient. The step size determines the magnitude of the perturbation at each iteration and is usually chosen carefully to balance the convergence speed and the likelihood of staying within the epsilon-bound region. After each update, the perturbed input is projected back onto the allowed region to enforce the constraints. The projection step ensures that the perturbed input remains within predefined boundaries, even if the gradient ascent step takes it outside of those boundaries. In case of a PGD attack, the original input ($x_0 = x$) is iteratively updated as follows:

$$x_{k+1} = clip(x_k + \alpha.sign(\Delta_{x_k} Loss(\theta, x_k, y_{x_k}))), \quad for\ k = 0, ..., K-1 \tag{2}$$

where α is the step size, K is the number of iterations and the clip$(\cdot)$ function applies element-wise clipping such that $\|x_k - x\|_\infty \epsilon, \epsilon \geq 0 \in \mathbb{R}$. We take x_K as the final perturbed spectral feature as an input for adversarial attack.

B. Translating a DNN model onto a Quantum Circuit

To enhance the security of our Quantum Neural Network (QNN) model against adversarial attacks, which are FGSM and PGD in the context of this paper, we propose an exploratory approach, Quantum Noise-Assisted Defense (QNAD). QNAD leverages both inherent and injected quantum noise to provide robustness against adversarial design. To achieve this, we begin by exploring the implementation of a classical Deep Neural Network (DNN) architecture on a quantum circuit.

To realize this implementation, it is crucial to convert the classical properties of the models, such as weights and activations, into the quantum domain. This process involves mapping each layer of the neural network onto a quantum circuit. To facilitate this, we divide the mapping process in a quantum circuit into three distinct stages. The first stage involves encoding the inputs by transforming classical data into quantum variables. We achieve this by applying rotation (R_X, R_Z) gates on qubits, representing the classical information in a quantum form. The next stage focuses on constructing the learnable quantum circuit. Here, we translate each operation present in the classical model, such as convolution and pooling, into

the corresponding set of operations in the quantum circuit. In the last stage, we obtain the classification accuracy and fine-tune the quantum circuit based on the evaluated results. This conversion process guarantees the precise preservation of the functionality of classical operations within the quantum framework. By formalizing the process and breaking it down into these stages, we simplify the understanding and implementation of translating a classical neural network model into a quantum circuit. The objective of our mapping approach is to offer an intuitive means of comprehending the conversion of a Deep Neural Network (DNN) architecture into a quantum circuit. It should however be noted that the influence of noise in quantum circuits on the mapped DNN architecture can vary contingent upon the selected mapping strategy.

C. Analyzing General Effect of Inherent Quantum Noise

After the successful translation of our classical Deep Neural Network (DNN) model into a Quantum Neural Network (QNN) model on a quantum circuit, we proceed to analyze the impact of noise on the QNN model within the quantum circuit. Noise is an inherent characteristic of quantum computers, stemming from the probabilistic nature of quantum measurements and the interactions between quantum systems and their surrounding environment [24].

Given the inherent nature of noise in quantum circuits, it uniformly affects every layer that is mapped within the quantum circuit. The presence of this inherent noise enhances the performance of the model, by counteracting the impact of adversarial input perturbations. To elaborate, we observe that this noise predominantly impacts the Multiply-Accumulate (MAC) operations of each layer. It can also be observed that, since the presence of adversarial inputs also induces alterations in the MAC operations, resulting in a reduction in model performance, and the inference classification accuracy. Hence, it becomes crucial to evaluate the impact of this inherent noise on the Multiply-Accumulate (MAC) operations of each layer within a Quantum Neural Network (QNN) model, as demonstrated below:

$$\mathbf{F} = \sum_{j=1}^{n} (x_j \pm \delta x_j) \times (w_j \pm \delta w_j) \tag{3}$$

Equation 1 illustrates the MAC operation within a neural network model, incorporating noise perturbation. This is represented by δx_j, which denotes the change in input caused by noise, and δw_j, which indicates the change in weight attributed to the same. The equation describes how a unit MAC operation is affected by the presence of noise. Considering that adversarial attacks perturb the input (x_j), it is reasonable to infer that a change (δx_j) in the perturbed input due to inherent noise leads to a more resilient model. The effect of the noise subverts the adverse impact of input alterations, thereby enhancing the model's robustness. Subsequently, we confirm our observations, noticing that the presence of noise actually enhances the accuracy of the Quantum Neural Network (QNN) model under adversarial attacks. This improvement can be attributed to the impact of noisy MAC operations on activations, which in turn directly influence the output accuracy of the model. Therefore, the robustness of MAC operations directly contributes to an increase in the overall model accuracy. Subsequently, we undertake an evaluation of the transferability of the insights obtained from our model by applying them to different DNN architectures. While achieving the optimal reduction of adversarial noise effects may necessitate individual tuning of each noise source, we observe that the values derived from one model yield similar outcomes when applied to a different model. The varied models share the same architecture but undergo different training process. This observation can be attributed to the fact that the impact of induced noise is largely dependent upon the architecture of the circuit being considered. While the outputs of a circuit are influenced by the provided inputs, we note that the influence of inputs on the noise within a circuit is negligible.

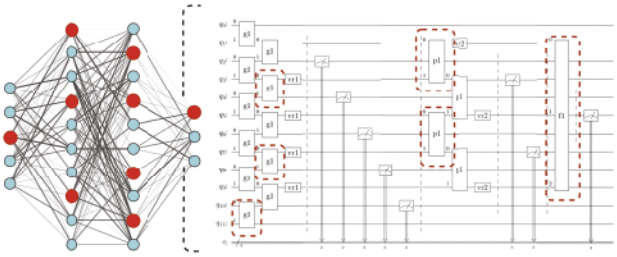

Fig. 2: Injecting crosstalk to improve adversarial robustness.

D. Injecting Crosstalk Noise in Specific Localities

While inherent noise in a quantum system offers a certain resilience against adversarial inputs, it does not restore the model's accuracy to a significant extent. Consequently, motivated by the robustness provided by inherent noise, within our approach, we propose investigating the effects of deliberate noise injection into the Quantum Neural Network (QNN) model. This is elucidated in Figure 2, where the highlighted nodes are susceptible to adversarial attack, and consequently their corresponding mapped counterparts on the quantum circuits are also susceptible to attack.

Given our knowledge of the QNN model's architecture following the mapping process, we propose the controlled introduction of regulated crosstalk noise into the circuit. The rationale behind this approach stems from the fact that introducing regulated crosstalk necessitates knowledge of the specific region within the quantum circuit where it should be injected. This is because it is important to exercise caution when introducing higher levels of crosstalk, as it can potentially have adverse effects on the model's performance. To circumvent the potential adverse effects of crosstalk, it is crucial to precisely localize the operations in the QNN model where crosstalk needs to be inserted. Referring to equation (1), we note that MAC operations depend on both the weights (w_j) of a neural network layer and its inputs (x_j). Existing

research has also demonstrated the importance of weight perturbation for defense against adversarial attacks [25], [26]. Therefore, we suggest the incorporation of crosstalk noise into the phase-gates of qubits within the quantum computer, which correspond to weights in the classical model.

To further localize the amount of crosstalk, in this approach, we focus on the weights that have the greatest impact on the model's performance. To achieve this, we suggest identifying the weights within our neural network model that carry the highest significance in influencing the output accuracy of the NN model. By selectively targeting these influential weights, we can effectively limit the impact of crosstalk while maximizing its potential benefits in improving the overall performance of the QNN model. Following this, we describe our critical weight selection approach.

1) Critical Weight selection: A traditional DNN architecture is composed of several hidden layers primarily comprising convolution, pooling, batch normalization and fully connected layers etc. An exhaustive search for finding vulnerable weights in these layers becomes computationally intensive, and thus we consider it an impractical endeavour. To circumvent this obstacle, we utilize a gradient-based important weight selection approach. In this method, we apply a set of inputs to the model for feed forward inference and then perform a backward gradient calculation using the model's loss. Thereafter, we evaluate the weights with the highest absolute gradient values in this process, and these weights are considered to contribute the most to the output of the DNN model. Elaborating on this, let us assume, $DNN_{W,b}(X)$ is a $d-$ layer neural network with c-dimensional output and σ_i are the corresponding activation functions for the connection between layer i and $i+1$. Assuming y_i^r as the one-hot coded vectors, and $\hat{y}_i^r = p_{ir}$ as the softmax probability for the i^{th} observation, falling in r^{th} class depending on the feature values X and weights W :

$$\hat{y}_i^r = p_{ir} = \mathbb{P}(Y_i = r) = \frac{\exp \beta_r' X_i}{\sum_{s=1}^{c} \exp(\beta_s' X_i)} \tag{4}$$

Here, X_i is the feature vector. We replace the simplistic multinomial logit structure $\beta_S' X_i$ using feature vectors by a suitable neural network structure. In order to calculate the gradient of the network loss L for an input sample, let us consider a particular weight w. Then,

$$\frac{\delta L}{\delta z_r} = y_r - \hat{y}_r \tag{5}$$

where $z_r = (DNN_{W,b}(X))_r$. By chain rule, we can easily compute the gradients $\frac{\delta L}{\delta w}$ as following:

$$\frac{\delta L}{\delta w} = \sum_{t=1}^{c} \frac{\delta L}{\delta z_t} \frac{\delta z_t}{\delta w} \tag{6}$$

Here, $\frac{\delta z_t}{\delta w}$ heavily depends on network structure involving the specific weight w. By analyzing the gradients ($\frac{\delta L}{\delta w}$), we identify the critical weights from the set of all weights W using the absolute gradient values. Next, we proceed to determine the specific quantum phase gates and corresponding qubits where these weights have been mapped in the circuit. This is accomplished under the assumption of the existence of a viable mapping strategy which could be utilized to map the DNN architecture onto the quantum circuit, as shown in Figure 2. Upon identifying the qubits and gates, we can attain the regions within the quantum circuit where crosstalk can be induced. This localization allows us to minimize the adverse effects associated with crosstalk. Subsequently, we select suitable qubits within the localized region for the insertion of crosstalk. These qubits are chosen from the physical layout of the quantum backend, and they should be neighboring qubits to our target qubit, where we intend to induce crosstalk.

2) Noise insertion: In the final step of our approach, we proceed to introduce crosstalk into the identified qubits, precisely regulating the amount to be inserted within each designated region. Throughout this process, we meticulously evaluate the effects of the introduced crosstalk and fine-tune its magnitude and parameters to attain optimal outcomes. This iterative refinement allows us to achieve the desired balance between the beneficial effects of crosstalk and any potential detrimental impact, ensuring optimal operation for our QNN model. The induction of crosstalk noise in a quantum circuit primarily involves the utilization of two-qubit Controlled-NOT (CNOT) gates. The tuning process of the crosstalk entails regulating the number of CNOT gates to be inserted into the quantum circuit. Subsequently, we analyze the observed enhancement in model accuracy resulting from our proposed approach. By systematically introducing and fine-tuning the crosstalk noise within the quantum circuit, we can assess the positive impact on the accuracy of the model. Moreover, CNOT gates are introduced in pairs. This is done to prevent the possibility of affecting the forward function, since an even number of CNOT gates results in an identity operation. However, the crosstalk effect due to the CNOT gates can still be perceived. This evaluation demonstrates the effectiveness of our approach in improving the overall performance of the Quantum Neural Network (QNN) model.

Algorithm 1 Quantum Crosstalk noise Injection

Input: QNN **after** Adversarial attack
Output: Noise Injected QNN

```
W_i = Set of important weights obtained from Eqs. [2-4]
Sort(List[W_i])
for W_i in List[W_i] do
    Extract Q_i, Op_i ← W_i
    List [Qu_imp].append[Q_i], List [Op_imp].append[Op_i]
end for
for qu in List [Qu_imp] do
    Extract(List AV_Q [Available neighboring qubits(qu)])
    if AV_Q not empty then
        op ← Op_imp corresponding to Qu_imp
        X_talk ← Evaluate crosstalk to be inserted into operation
        Inject_Noise (qu, op, X_talk)
    end if
end for
```

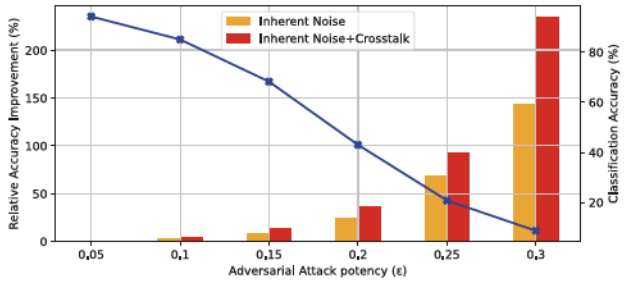

Fig. 3: Impact of FGSM attack for LeNet on MNIST and corresponding improvement in accuracy under QNAD.

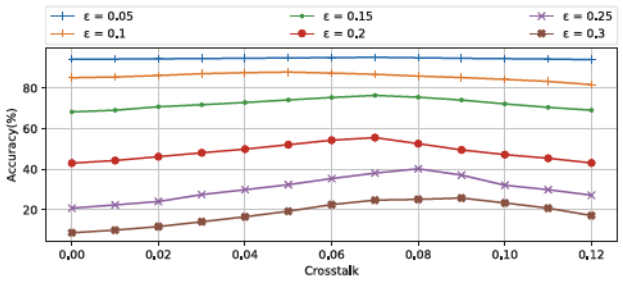

Fig. 4: Variation in classification accuracy for varying crosstalk levels under FGSM attack on LeNet-MNIST configuration.

Our approach for inserting crosstalk is presented in Algorithm 1. In this algorithm, we aim to defend against adversarial attacks by determining the location and quantity of crosstalk to be inserted into the QNN model. The algorithm begins by obtaining the QNN model after the adversarial attack has been performed *(line 1)*. We then identify the critical weights in the model using their absolute gradients ($\frac{\delta L}{\delta w}$), as described in Equations 2 to 4 *(line 2)*. These weights are sorted in a list based on their importance, which indicates their influence on the model's accuracy. Next, we determine the corresponding qubit and the rotation operation on that qubit for each weight gradient in the classical model. We add the qubits and operations to individual lists, denoted as Qu_{imp} and Op_{imp}, respectively *(lines 3-5)*. For each qubit in the Qu_{imp} list, we identify the neighboring qubits and assess their availability in the quantum circuit *(lines 7 and 8)*. Availability is determined based on whether the qubit is undergoing any operations in conjunction with the QNN model. Neighboring qubits refer to one-hop and two-hop qubits in the physical architecture that are adjacent to the qubit under consideration. If there are no available neighboring qubits, it implies that crosstalk cannot be introduced via noise injection. However, if neighboring qubits are available, we obtain the operation corresponding to the qubit *(lines 9-10)*. Subsequently, we evaluate the amount of crosstalk to be inserted and proceed with the insertion into the qubit *(lines 11-12)*. This process is repeated for each weight in the list of important weights. Finally, we obtain the QNN model with the injected noise after completing the injection process for all important weights.

The process of inserting noise through crosstalk offers advantages over general noise injection. It provides greater control over the noise injection procedure, thereby enhancing the feasibility of mitigating the effects of adversarial attacks.

IV. Evaluation

A. Experimental setup

In order to evaluate the efficiency of our proposed QNAD approach in enhancing the adversarial robustness, we utilize three network dataset configurations – LeNet on MNIST, AlexNet on CIFAR 10 and VGG-16 on CIFAR 100 datasets. While these models provide substantial depth and complexity in terms of network dimensions and computations, datasets CIFAR 10 and CIFAR 100, with 10 and 100 output classes respectively, are being traditionally used in existing research to evaluate computer vision workloads. The networks are developed using PyTorch framework, and are trained on cloud, following which, they are deployed in the quantum computer for inference. The baseline classification accuracies for LeNet on MNIST, AlexNet on CIFAR 10 and VGG-16 on CIFAR 100 are 98.2%, 73.7% and 64.3% respectively. Subsequently, we execute both the FGSM and PGD adversarial attacks on all the network-dataset configurations and evulate the efficiency of QNAD in defending such attacks. To counteract the potential attacks, we employ a defensive strategy by extracting the intrinsic noise characteristics of the IBM Quantum's physical backend, namely ***ibm_nairobi***. Subsequently, we conduct an estimation of the noise resulting from crosstalk on the same quantum backend. The outcomes of these analyses are presented in the subsequent section.

B. Experimental Results

In this section, we study the effect of varying amounts of inherent noise present in quantum systems, as well as the effects of induced crosstalk on different DNN models mapped on QNN's. For this purpose, present three case studies.

*1) **Case 1 - LeNet on MNIST**:* We first assess the performance of QNAD by employing a LeNet model trained on an MNIST dataset for two different attack scenarios.

Efficiency on FGSM Attack: In this experiment, we conduct an analysis where we apply the FGSM attack to the incoming input vectors. We increase the perturbation magnitude, denoted by epsilon (ϵ), and observe the resulting degradation in the classification accuracy of the network. The experimental results, depicted in Figure 3, demonstrate how the classification accuracy changes with varying epsilon values. Subsequently, we utilize a quantum circuit to map the Deep Neural Network (DNN) model and evaluate the accuracy of the newly established Quantum Neural Network (QNN) model. The figure also showcases the improvement in classification accuracy achieved by QNAD. Based on the results depicted in Figure 3, it is evident that the inherent noise inherent in the quantum system plays a crucial role in enhancing the accuracy of the model following an adversarial attack. The figure clearly demonstrates that as the perturbation in the input increases, the

effectiveness of the intrinsic noise in improving robustness also increases, leading to a significant improvement in accuracy of up to 144%.

We conducted a study to analyze the impact of crosstalk injection on model accuracy. Following Algorithm 1 described earlier, we selectively chose important weights for crosstalk insertion. By regulating and localizing the amount of crosstalk injection, we observed the change in accuracy as the level of crosstalk varied. The results, depicted in Figure 4, illustrate the relationship between crosstalk amount and accuracy. The degree of crosstalk is measured by the ratio of the number of weights in the Quantum Neural Network (QNN) model affected by crosstalk, to the total number of weights.

Figure 4 demonstrates that the model's performance improves with increasing crosstalk up to a certain threshold. However, exceeding this threshold leads to a gradual decrease in accuracy. We further examined the influence of crosstalk on QNN models under various levels of adversarial attacks. Interestingly, we found that the optimal crosstalk level for achieving maximum accuracy varied depending on the intensity of the attack. Additionally, we observed that there is a threshold specific to each attack level, beyond which accuracy starts to decline. Moreover, incorporating both crosstalk and inherent noise significantly enhances the accuracy of the QNN model. The combined approach leads to a remarkable increase in model accuracy, reaching up to 235%, as shown in Figure 3.

Efficiency on PGD Attack: In this experiment, we evaluate the efficiency of QNAD under the PGD attack for LeNet-MNIST configuration. The corresponding degradation in classification accuracy for varying attack intensities is represented in Figure 5. As shown in the figure, the iterative nature of the PGD attack leads to a slightly higher degradation in classification accuracy compared to the FGSM attack. The PGD attack, with its multiple iterations, results in a more pronounced impact on the model's accuracy. This degradation in accuracy is mitigated by up to 183%, with the aid of inherent noise in the quantum circuit.

We conducted a study to investigate the effect of crosstalk injection on model accuracy. By selectively inserting crosstalk into key weights using Algorithm 1, we controlled its level. The resulting accuracy was observed as we varied the crosstalk amount. Figure 6 illustrates the relationship between crosstalk and accuracy, where crosstalk is measured by the fraction of affected weights in the QNN model. Figure 6 shows that increasing crosstalk initially improves model performance, but surpassing a threshold leads to a decline in accuracy. We examined crosstalk's impact on QNN models under different adversarial attack levels and found that the optimal crosstalk level varied based on attack intensity. Each attack level had a specific threshold where accuracy started to decline. Moreover, incorporating both crosstalk and inherent noise significantly boosted the QNN model's accuracy by up to 285%, as demonstrated in Figure 5.

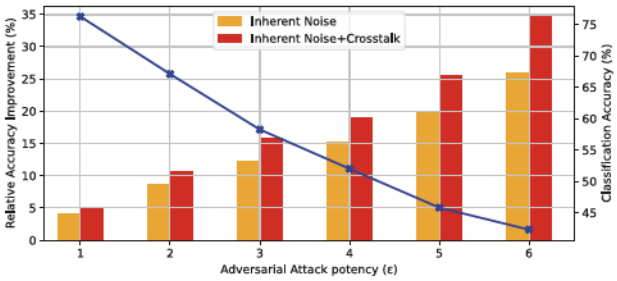

Fig. 7: Impact of FGSM attack on AlexNet-CIFAR 10 configuration and corresponding improvement in classification accuracy by QNAD.

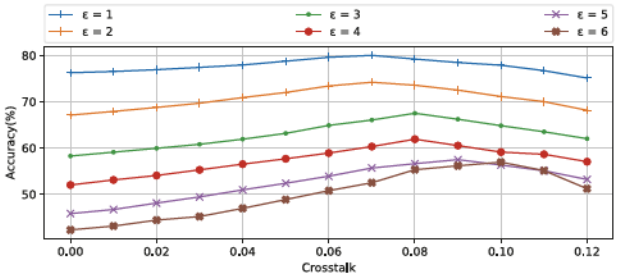

Fig. 8: Variation in classification accuracy for varying crosstalk levels under FGSM attack with differing intensity on AlexNet-CIFAR 10 configuration.

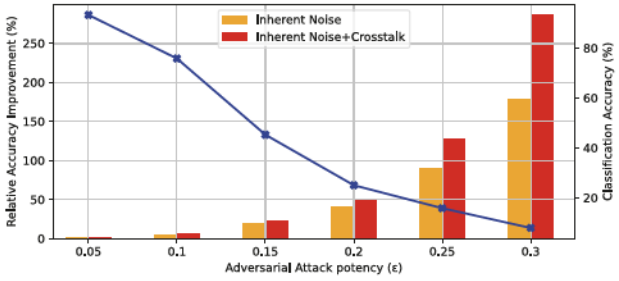

Fig. 5: Impact of PGD attack on LeNet-MNIST configuration and corresponding improvement in classification accuracy furnished by QNAD.

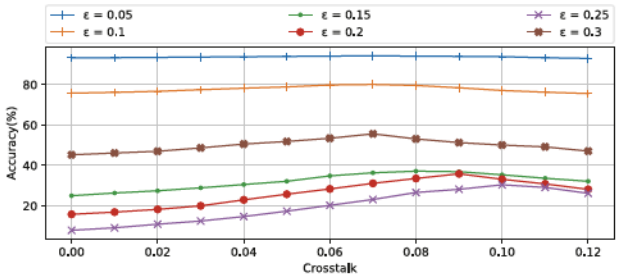

Fig. 6: Variation in classification accuracy for varying crosstalk levels under PGD attack on LeNet-MNIST configuration.

2) Case 2 - AlexNet on CIFAR 10: In this experiment, to evaluate the performance of QNAD, we utilize an AlexNet model trained on an CIFAR 10 dataset and examine its effectiveness under both the attack scenarios.

Efficiency on FGSM Attack: In this experiment, we assess the effectiveness of QNAD against the FGSM attack on the AlexNet-CIFAR 10 configuration. We analyze the resulting degradation in classification accuracy for different attack intensities, as depicted in Figure 7. The CIFAR-10 dataset on AlexNet exhibits similar trends as MNIST, albeit with a higher value of epsilon (ϵ). Nonetheless, the integration of inherent noise in the quantum circuit aids in mitigating the accuracy degradation, contributing to improved performance by up to 26%, even at higher ϵ values.

We conducted a study on the effect of crosstalk injection on model accuracy. By selectively introducing crosstalk into key weights, we controlled its level. Figure 8 shows the relationship between crosstalk and accuracy, with crosstalk measured by the fraction of affected weights in the QNN model. Increasing crosstalk initially improves performance, but surpassing a threshold leads to a decline in accuracy. Crosstalk's impact on QNN models varied based on attack intensity, with each level having a specific accuracy threshold. Furthermore, incorporating both crosstalk and inherent noise significantly boosted model accuracy by 34%, as shown in Figure 7. It is important to note that the observed improvement

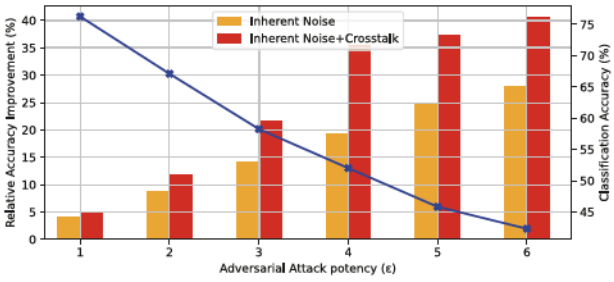

Fig. 9: Impact of PGD attack for AlexNet-CIFAR 10 configuration and corresponding improvement in classification accuracy furnished by QNAD.

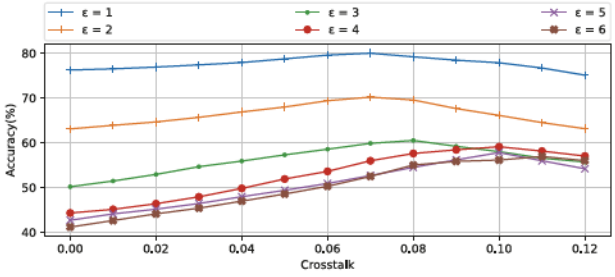

Fig. 10: Variation in classification accuracy for varying crosstalk levels under PGD attack of differing intensity on AlexNet-CIFAR 10 configuration.

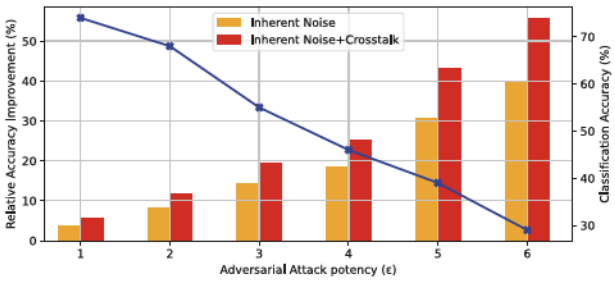

Fig. 11: Impact of FGSM attack for VGG16-CIFAR 100 configuration and corresponding improvement in accuracy under QNAD.

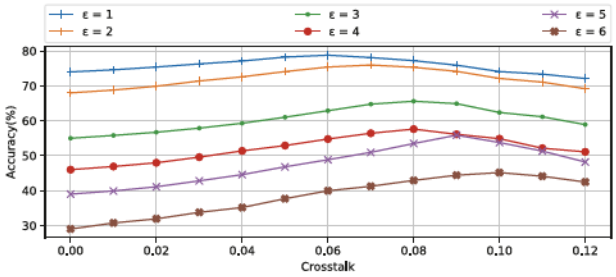

Fig. 12: Variation in classification accuracy for varying crosstalk levels under FGSM attack of differing intensity on VGG16-CIFAR 100 configuration.

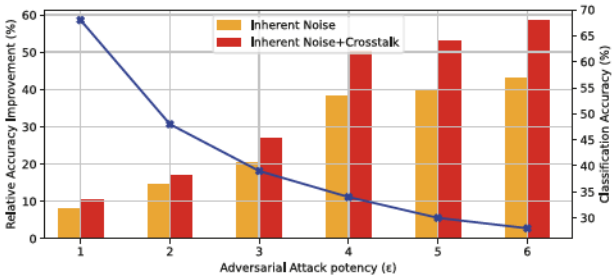

Fig. 13: Impact of PGD attack for VGG16-CIFAR 100 configuration and corresponding improvement in classification accuracy by QNAD.

in accuracy for this case is relatively low compared to Case 1. This can be attributed to the fact that the initial network accuracy is less affected by the attack in this scenario. In cases where the reduction in accuracy is more significant, our proposed QNAD approach has demonstrated a higher improvement in accuracy, thus enhancing the adversarial robustness of the network. The effectiveness of QNAD is more pronounced when the initial impact of the attack on the classification accuracy is greater, as shown in Case 1.

Efficiency on PGD Attack: In this experiment, we evaluate our QNAD's effectiveness against the PGD attack on the AlexNet-CIFAR 10 configuration. We examine the impact on

TABLE I: Summary of results.

Adversarial attack	Model-dataset	Optimal ϵ	Relative accuracy improvement (%)
FGSM [11]	LeNet-MNIST	0.3	235.17
	AlexNet-CIFAR-10	6	34.61
	VGG16-CIFAR-100	6	58.07
PGD [22]	LeNet-MNIST	0.3	268.28
	AlexNet-CIFAR-10	6	42.16
	VGG16-CIFAR-100	6	59.25

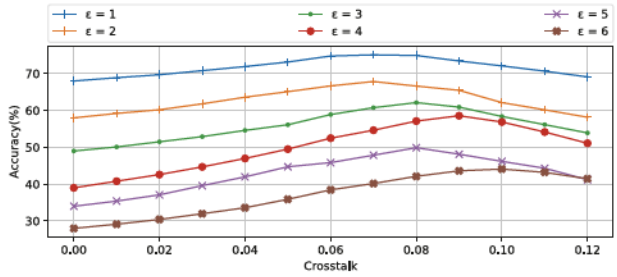

Fig. 14: Variation in classification accuracy for varying crosstalk levels under PGD attack of varying intensity on VGG16-CIFAR 100 configuration.

classification accuracy at various attack intensities, shown in Figure 9. The CIFAR-10 dataset on AlexNet shows similar patterns to MNIST, but with a larger epsilon (ϵ) value. However, incorporating inherent noise in the quantum circuit helps reduce accuracy degradation and enhances performance by up to 28%, even at higher ϵ values.

We investigated the impact of crosstalk injection on model accuracy by selectively introducing it into specific weights. Figure 10 demonstrates the relationship between crosstalk and accuracy, measured as the fraction of affected weights in the QNN model. Increasing crosstalk initially enhances performance, but exceeding a threshold results in decreased accuracy. Crosstalk's effect on QNN models varies with attack intensity, each having a specific accuracy threshold. Furthermore, incorporating both crosstalk and inherent noise significantly improves model accuracy by 42% (Figure 9).

3) Case 3 - VGG16 on CIFAR 100: In this experiment, we assess the performance of QNAD by employing an VGG16 model trained on the CIFAR 100 dataset. We evaluate its effectiveness in two attack scenarios, FGSM and PGD.

Efficiency on FGSM Attack: In this experiment, we evaluate our QNAD approach against the FGSM attack on VGG16-CIFAR 100 configuration. Figure 11 illustrates the impact on classification accuracy at various attack intensities. Incorporating inherent noise in the quantum circuit reduces accuracy degradation and improves performance by up to 43%, even as intensity of adversarial attack, *i.e.,* ϵ, increases.

We studied crosstalk injection's impact on model accuracy by selectively introducing it into specific weights. Figure 12 shows the relationship between crosstalk and accuracy, measured as the fraction of affected weights in the QNN model. Increasing crosstalk initially improves performance, but exceeding a threshold decreases accuracy. Crosstalk's effect varies with attack intensity, each with a specific accuracy threshold. Additionally, incorporating crosstalk and inherent noise significantly boosts model accuracy by 58% (Figure 11).

Efficiency on PGD Attack: We test our QNAD approach against the PGD attack on VGG16-CIFAR 100 configuration. Figure 13 shows the impact on classification accuracy at varying attack intensities. Incorporating inherent noise in the quantum circuit minimizes accuracy degradation and boosts performance by up to 44%, even at higher ϵ values. Moreover, we examined the impact of selectively introducing crosstalk into specific weights on model accuracy. Figure 14 illustrates the relationship between crosstalk and model accuracy, measured as the fraction of affected weights in the QNN model. Initially, increasing crosstalk improves performance, but surpassing a threshold decreases accuracy. Crosstalk's effect varies with attack intensity, each with its own accuracy threshold. Moreover, the incorporation of crosstalk and inherent noise significantly enhances model accuracy by 59%, as demonstrated in Figure 13.

Summary of results: The results obtained by our experiments are summarized in Table I. The first column indicates the attack model considered for our experiments. Following this, the second column describes the dataset on which the experiments were evaluated. The third column depicts the ϵ or attack intensity, for which our defense provides the highest robustness. Finally, columns four and five show the accuracy of the QNN models before and after the operation of our proposed defense strategy. We observe relative improvement in accuracy of up to 268.28%, furnished by QNAD, for PGD attack performed on LeNet-MNIST configuration, with intensity ϵ of 0.3. The results from Table I underscore the efficacy of QNAD in enhancing robustness against attacks driven by adversarial inputs. Moreover, since it utilizes inherent properties of quantum circuits, QNAD preserves the extensibility of its integration into pre-existing adversarial defense frameworks, particularly when such frameworks are implemented within a quantum circuitry paradigm.

V. Conclusion

In conclusion, this paper shows that intrinsic quantum noise can improve the robustness of DNNs against adversarial input attacks. The noisy behavior of quantum computers reduces the impact of attacks, enhancing the accuracy of compromised DNNs. To further enhance DNN robustness, we propose QNAD, that induces carefully crafted crosstalk in the quantum computer by perturbing a subset of pretrained network weights, causing entanglement among neighboring qubits. Experimental evaluations demonstrate that QNAD achieves up to 268% relative improvement against adversarial input attacks compared to conventional DNN implementations on state-of-the-art network dataset configurations. These findings highlight the potential of leveraging quantum noise and crosstalk injection techniques for enhancing DNN security in adversarial scenarios. Further exploration is imperative to substantiate the validity of these intuitions and their supposed effects on the resilience of DNN models mapped onto a quantum computer.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis *et al.*, "Deep learning for computer vision: A brief review," *Computational intelligence and neuroscience*, vol. 2018, 2018.

[3] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *ieee Computational intelligenCe magazine*, vol. 13, no. 3, pp. 55–75, 2018.

[4] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir *et al.*, "Wide & deep learning for recommender systems," in *Proceedings of the 1st workshop on deep learning for recommender systems*, 2016, pp. 7–10.

[5] A. Steane, "Quantum computing," *Reports on Progress in Physics*, vol. 61, no. 2, p. 117, 1998.

[6] V. Dunjko and H. J. Briegel, "Machine learning & artificial intelligence in the quantum domain: a review of recent progress," *Reports on Progress in Physics*, vol. 81, no. 7, p. 074001, 2018.

[7] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017.

[8] A. A. Clerk, M. H. Devoret, S. M. Girvin, F. Marquardt, and R. J. Schoelkopf, "Introduction to quantum noise, measurement, and amplification," *Reviews of Modern Physics*, vol. 82, no. 2, p. 1155, 2010.

[9] C. Gardiner and P. Zoller, *Quantum noise: a handbook of Markovian and non-Markovian quantum stochastic methods with applications to quantum optics*. Springer Science & Business Media, 2004.

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

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

[12] Z. He, A. S. Rakin, and D. Fan, "Parametric noise injection: Trainable randomness to improve deep neural network robustness against adversarial attack," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 588–597.

[13] X. Liu, M. Cheng, H. Zhang, and C.-J. Hsieh, "Towards robust neural networks via random self-ensemble," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 369–385.

[14] A. Jeddi, M. J. Shafiee, M. Karg, C. Scharfenberger, and A. Wong, "Learn2perturb: an end-to-end feature perturbation learning to improve adversarial robustness," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1241–1250.

[15] C. Xie, M. Tan, B. Gong, A. Yuille, and Q. V. Le, "Smooth adversarial training," *arXiv preprint arXiv:2006.14536*, 2020.

[16] D. Roy, I. Chakraborty, T. Ibrayev, and K. Roy, "On the intrinsic robustness of nvm crossbars against adversarial attacks," in *2021 58th ACM/IEEE Design Automation Conference (DAC)*. IEEE, 2021, pp. 565–570.

[17] S. K. Cherupally, A. S. Rakin, S. Yin, M. Seok, D. Fan, and J.-s. Seo, "Leveraging noise and aggressive quantization of in-memory computing for robust dnn hardware against adversarial input and weight attacks," in *2021 58th ACM/IEEE Design Automation Conference (DAC)*. IEEE, 2021, pp. 559–564.

[18] I. Cong, S. Choi, and M. D. Lukin, "Quantum convolutional neural networks," *Nature Physics*, vol. 15, no. 12, pp. 1273–1278, 2019.

[19] S. Oh, J. Choi, and J. Kim, "A tutorial on quantum convolutional neural networks (qcnn)," in *2020 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2020, pp. 236–239.

[20] S. A. Stein, B. Baheri, D. Chen, Y. Mao, Q. Guan, A. Li, S. Xu, and C. Ding, "Quclassi: A hybrid deep neural network architecture based on quantum state fidelity," *Proceedings of Machine Learning and Systems*, vol. 4, pp. 251–264, 2022.

[21] W. Jiang, J. Xiong, and Y. Shi, "A co-design framework of neural networks and quantum circuits towards quantum advantage," *Nature communications*, vol. 12, no. 1, p. 579, 2021.

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

[23] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille, "Mitigating adversarial effects through randomization," *arXiv preprint arXiv:1711.01991*, 2017.

[24] S. Resch and U. R. Karpuzcu, "Benchmarking quantum computers and the impact of quantum noise," *ACM Comput. Surv.*, vol. 54, no. 7, jul 2021. [Online]. Available: https://doi.org/10.1145/3464420

[25] D. Wu, Y. Wang, and S. Xia, "Revisiting loss landscape for adversarial robustness," *CoRR*, vol. abs/2004.05884, 2020. [Online]. Available: https://arxiv.org/abs/2004.05884

[26] J. Xu, L. Li, J. Zhang, X. Zheng, K.-W. Chang, C.-J. Hsieh, and X. Huang, "Weight perturbation as defense against adversarial word substitutions," in *Findings of the Association for Computational Linguistics: EMNLP 2022*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 7054–7063. [Online]. Available: https://aclanthology.org/2022.findings-emnlp.523