

Article

Cybersecurity Monitoring of Quantum Cyber-Physical Systems Using Artificial Intelligence: Detection of False Data Injection Cyber-Attacks in Photonics-Driven Quantum Information

Mohammad Reza Habibi

Department of the Built Environment, Aalborg University, 2450 København SV, Denmark; mre@build.aau.dk

Abstract

Quantum cyber-physical systems can involve computation-based strategies like quantum algorithms, physical parts, and communication-based infrastructures such as physical quantum bits (qubits), measurement units, and communication links. A photonics-driven quantum cyber-physical system can rely on physical parameters of the environment or communication links such as the refractive index. The injection of false data into the actual value of the refractive index can cause the use of an incorrect value of the refractive index in the system if needed, and as a result, an incorrect interpretation of information or even access to non-real information instead of the actual information. The first attempt to avoid this issue can be the detection of the existence of the cyber-attacks in the system. This paper will address this challenge using artificial intelligence for binary classification to detect the cyber-attacks in the system. For the proof of concept, the proposed strategy is examined deploying the real and imaginary parts of the refractive index value corresponding to a semiconductor, i.e., GaAs. Different scenarios are performed, including a single evaluation, an analysis of several training runs, a comparison considering two types of normalization techniques, variations in the size of artificial intelligence, and a comparison among different machine learning techniques, including artificial neural networks, decision tree models, a logistic regression model, and support vector machine classifiers. Based on the obtained results, for the single evaluation, the class of 95.24 % of the testing samples could be classified successfully. In addition, for the case of the analysis of several training runs, a total of 9000 runs were run for 60 shallow artificial neural networks with different sizes. For 58 neural networks, the maximum achieved accuracy was 100 %. Besides, for the case of the comparison between the normalization techniques, two methods were evaluated, i.e., min-max and z-score normalization. The results were very close to each other, but, more accurately, z-score normalization indicated a better performance and a higher accuracy. Finally, among the mentioned machine learning models, artificial neural networks mostly showed higher accuracies.



Academic Editor: Jinliang Liu

Received: 22 June 2026

Revised: 21 July 2026

Accepted: 24 July 2026

Published: 30 July 2026

Copyright: © 2026 by the author.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: quantum cyber-physical systems; quantum science; artificial intelligence; artificial neural networks; photon energy; refractive index; cybersecurity; false data injection cyber-attacks; photonics-driven quantum information

1. Introduction

Artificial intelligence (AI) can successfully address many challenges in engineering and science, and more specifically cyber-physical systems, because of its high interaction

with data and its ability to provide an interface between cyber and physical parts. For example, AI can be implemented to predict various variables and physical parameters in cyber-physical energy and power systems such as electrical loads [1,2], renewable energy outputs, e.g., photovoltaic systems [3,4], wind power [5,6], and marine-based energy [7,8], as well as, electricity price [9,10]. In addition to forecasting-based applications, AI can be implemented in sensoreless applications for cyber-physical systems, such as energy systems [11,12]. Besides, as another application, AI can support energy storage-based cyber-physical systems. In [13], AI has been employed for energy storage applications, i.e., for the estimation of the surface temperature of lithium-ion batteries. In addition, AI has been employed in digital twin-based applications [14,15]. Also, AI has been deployed in different applications connected to the buildings such as indoor temperature modeling [16]. Furthermore, AI-based methods can address cybersecurity and cyber-resiliency challenges in classical systems. For example, [17] implemented AI for cybersecurity monitoring of power electronics, and [18,19] proposed AI-based strategies against cyber-attacks in cyber-physical DC microgrids.

In this study, inspired by the efficient operation of AI for cybersecurity improvement of classical cyber-physical systems, an AI-based strategy is proposed to detect the existence of false data injection attacks (FDIAs) in quantum cyber-physical systems. Previously, researchers attempted to integrate AI and quantum science across different research lines and applications. For example, one of the research lines is the study of estimation or forecasting-based methods using quantum computing and AI, which can be deployed in many applications, e.g., demand forecasting [20], weather and environmental-based forecasting [21–24], and carbon price forecasting [25]. As another collaborative example of quantum science and AI, the implementation of AI for quantum error correction and quantum noise estimation can be mentioned [26,27]. In addition, another attractive research line is the quantum-inspired optimization algorithms, which have been implemented to overcome many challenges in different systems, such as quantum-inspired particle swarm optimization for unit commitment [28] and economic load dispatch [29] in power systems, electromagnetic applications [30], photovoltaic systems [31], energy consumption of Internet of Things networks [32], and load demand forecasting [33]. More studies related to the quantum-inspired optimization can be found in [34–36].

As mentioned earlier, numerous studies have been conducted on collaborative applications deploying quantum science and AI. Although both possible directions of collaboration between quantum science and AI, namely, AI for quantum science and quantum science for AI, have gained the attention of researchers, the cybersecurity of quantum cyber-physical systems remains an important challenge that needs to be addressed. Given the potential of AI to address various challenges in classical cyber-physical systems, its role in solving cyber-related problems in quantum cyber-physical systems should be evaluated. Therefore, this study will investigate how a positive collaboration between AI and quantum cyber-physical systems can be provided to address cybersecurity challenges in quantum-based systems. In this study, photonics-driven quantum information is considered the system, while the cyber-attack that may compromise this system is the challenge, and AI is considered as the potential solution to address this issue, with the detection of the cyber-attacks serving as the starting point.

Photonics-driven quantum information systems rely on light, and the refractive index is a key parameter for analyzing light propagation. Therefore, this study investigates cyber-attacks that target measured, stored, or transmitted refractive index data. Here, the type of cyber-attack is FDIA. In other words, false data on the refractive index are considered the studied cyber-attack. The refractive index is a vital parameter that affects different optical parameters and phenomena, e.g., the speed of light, reflection, and absorption of light when

interacting with different materials, etc. [37]. This parameter can have a complex value and may vary based on different parameters or situations, such as environmental conditions. Knowledge about deviations or errors in the measured refractive index can help a quantum cyber-physical system avoid the incorrect interpretation of information. If, instead of the actual value of the refractive index, a quantum technology-based system receives an incorrect value, it can lead to an incorrect interpretation of information. Therefore, in the case of FDIAs on the value of the refractive index, it is very important to have a plan against this type of cyber-attacks in the system.

The first step toward solving this issue is to identify the existence of false data in the system. This paper proposes a method to detect the existence of FDIAs in the value of the refractive index. Here, AI will be used for the identification of FDIAs targeting the refractive index. Basically, an AI-based strategy is a data-driven approach, and it works based on data in the system. It can be independent of the physical or mathematical information of the quantum cyber-physical system and can treat the system as a black-box. Before the implementation of an AI-based model, it should be trained to calculate the parameters of the model. A crucial step toward the preparation of an AI-based model is access to a dataset for the training phase. Using that dataset, the parameters of the model can be determined. After tuning the parameters of the AI-based model, the tuned model can be deployed for testing purposes related to the desired material for which it has been trained for, i.e., the identification of FDIAs targeting the value of the refractive index of the studied material. In this study, in order to evaluate the proposed approach for the detection of false data in the refractive value, a comprehensive analysis of the results is conducted with the following features:

- The proposed cyber-attack detection approach for quantum cyber-physical systems is based on AI. The method is data-driven, and as a result, there is no need to have additional physical or mathematical information about the quantum-based system.
- This paper offers a solution to detect the existence of cyber-attacks in quantum cyber-physical systems using a binary classification strategy by classifying two types of data, i.e., healthy data and attacked data.
- This study will use a lightweight AI model, which can make the proposed FDIA detection strategy for the quantum cyber-physical systems easy to use.
- Typically, AI-based models require optimization in the training phase, while this process may rely on randomly initialized parameters, which can result in variations in accuracy by repeating the training. Therefore, this study will provide several training runs analysis to perform the training phase several times to obtain a more reliable interpretation of the results.
- AI-based approaches use datasets for the training, and dataset normalization is one of the initial steps before the training. Normalization techniques may affect the results. Therefore, this study will investigate the effects of two normalization methods on the proposed approach, i.e., min-max and z-score normalization.
- There are many types of modeling approaches that can be used to learn the classification model described by the samples from the original model. Therefore, to have an in-depth analysis of the cyber-attack detection approach, different models are examined, i.e., artificial neural networks (ANNs), decision trees, logistic regression, and support vector machines (SVMs).

For the rest of this study, Section 2 will study the effect of FDIAs on the values of the refractive index in quantum cyber-physical systems. Then, in Section 3, the proposed AI-based approach for the detection of FDIAs in photonics-driven quantum information-based cyber-physical systems will be introduced. Section 4 provides comprehensive results for the detection of FDIAs targeting the refractive index values. In Section 4, a case study

will be evaluated. An analysis of several training runs is provided in Section 4 after the case study evaluation. In addition, two different normalization techniques, and an AI model with different sizes will be examined in Section 4. In Section 5, different models for the FDIA framework will be evaluated. Section 6 will discuss the results of this study, and Section 7 will provide the conclusion of this work. In addition, suggestions for continuing this research line to increase the cyber-resiliency of quantum cyber-physical systems are provided at the end of this study in Section 8.

2. Effect of False Data on Refractive Index in Quantum Cyber-Physical Systems

The implementation of a wrong refractive index value may result in a wrong interpretation of the information. To encode, transfer, process, or decode information, time or phase can play important roles. Considering that the refractive index can affect the wave traveling time or the phase, the implementation of a wrong refractive index may result in a non-real information. For example, in the case of time-bine encoding, the time window plays an important role, i.e., for early arrival photons and late arrival photons, the information is encoded to $|0\rangle$ and $|1\rangle$, respectively.

By repeating the measurement for identical photons, a quantum state $\sqrt{\frac{q_0}{q_0+q_1}}|0\rangle + \sqrt{\frac{q_1}{q_0+q_1}}|1\rangle$ can be obtained, where q_0 and q_1 are measured counts for $|0\rangle$ (i.e., early arrival photons) and $|1\rangle$ (i.e., late arrival photons), receptively. For the very small or negligible $\lambda \times \frac{dn}{d\lambda}$ (where, λ is the wavelength value), the required time to pass a medium by a photon plays an important role and in a fixed pass length, it is proportional to the speed of the wave, i.e.,

$$\Delta t \propto \frac{1}{v_0} \times n \quad (1)$$

where n is related to the refractive index of the medium, and v_0 is the speed of light at vacuum. Implementation a wrong refractive value can result into calculating a wrong time-window and as a result wrong interpretation of information. In other words, if non-real value $n_F = n + f_D$ is used in the calculations,

$$\Delta t_F \propto \frac{1}{v_0} \times n_F. \quad (2)$$

Then,

$$\Delta t_F \propto \frac{1}{v_0} \times (n + f_D). \quad (3)$$

Therefore, if Δt_F will be used in the system, a non-real term δt is appeared in the time calculation that is proportional to the false data f_D as follows:

$$\Delta t_F = \Delta t + \delta t, \quad (4)$$

and,

$$\delta t \propto \frac{1}{v_0} f_D. \quad (5)$$

If the false data f_D is a positive real number, then the calculation of the time duration will be more than the real time duration. In addition, if f_D has a negative real value, the calculated time will be shorter than the real value of that. Therefore,

$$\begin{cases} \Delta t_F > \Delta t & \text{for } f_D > 0 \\ \Delta t_F = \Delta t & \text{for } f_D = 0 \\ \Delta t_F < \Delta t & \text{for } f_D < 0 \end{cases} \quad (6)$$

As another example, the phase accumulation/differences is proportional to the $n \times R$ (where R is the length of the path). In other words,

$$\phi \propto n \times R. \quad (7)$$

So, by injection the false data f_D into the refractive index value, if n_F is used for the phase calculation, then a wrong value of phase can be appeared in the system, i.e., ϕ_F as follows,

$$\phi_F = \phi + \delta\phi, \quad (8)$$

where,

$$\delta\phi \propto f_D \times R. \quad (9)$$

If $\phi_1 < \phi < \phi_2$, for sufficiently small value of $|f_D|$,

$$\begin{cases} \phi_2 > \phi_F > \phi & \text{for } f_D > 0 \\ \phi_F = \phi & \text{for } f_D = 0 \\ \phi_1 < \phi_F < \phi & \text{for } f_D < 0 \end{cases}. \quad (10)$$

Therefore, non-accurate and false refractive index can result in a difference in the phase calculations, i.e., $\delta\phi$. To conclude this part, the deployment of the photons to construct the state of the qubit under the cyber-attack is shown by Figure 1. In addition, Figure 2 depicts the wrong calculations corresponding to the time and phase, under an FDIA with a same situation but using different representations.

This section presented a simplified conceptual framework to show the possible consequences of falsified refractive index data. However, the goal of this study is to detect FDIAs targeting refractive index values by providing the data as inputs to an AI model, rather than to model their detailed physical effects.

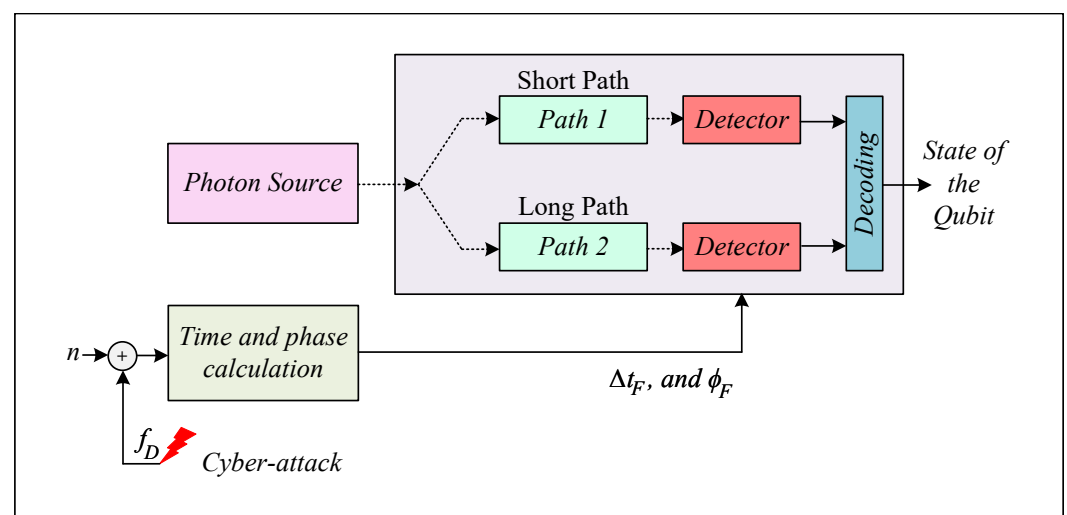


Figure 1. Implementation of photons to construct the state of the qubit under a cyber-attack. In the case of cyber-attack, i.e., $f_D \neq 0$, the state of the qubit may be reconstructed incorrectly, and as a result, an incorrect interpretation of information carried by the qubit.

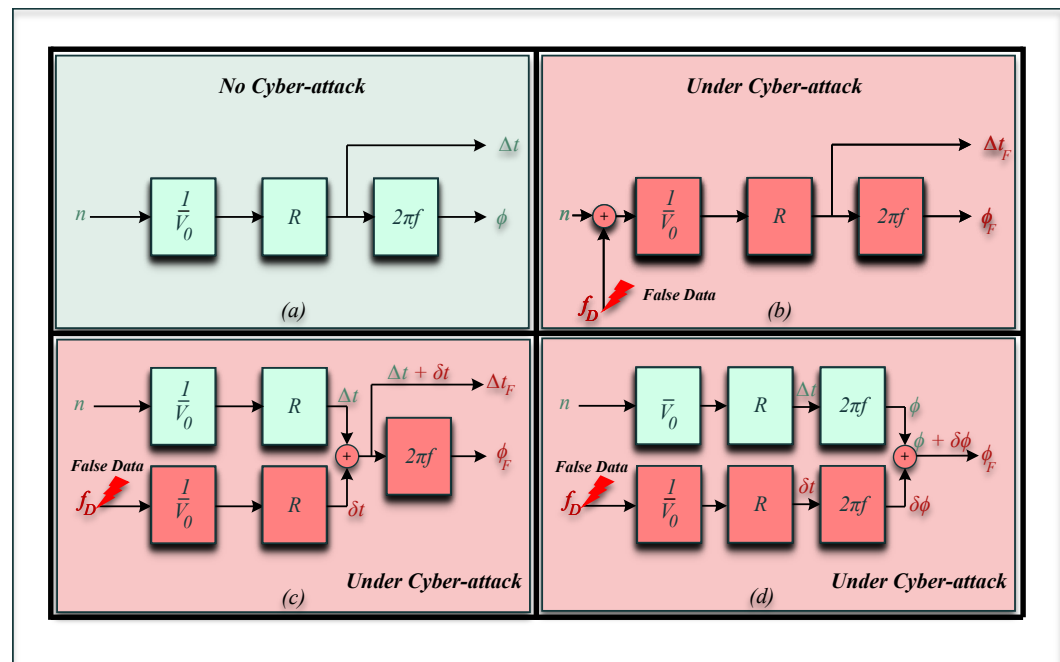


Figure 2. Time and phase calculation framework illustrating the effect of false data on the calculated time delay and phase. In case (a), there is no cyber-attack on the system. Cases (b–d) show different representations of an FDIA with the same value of false data value f_D . In addition, f is the frequency of the photon.

3. Proposed Strategy

This section includes four parts. For the first part, the inputs data preparation will be explained. In addition, for the second part, the output dataset will be discussed. For the third part, the calculations using the ANN for the detection of cyber-attack will be talked about. Finally, in part four, all steps are summarized by an algorithm.

3.1. Part 1: Inputs Dataset Preparation

Here, an ANN is implemented for a binary classification goal, that is, to detect FDIAs. The first step is to define the classes set for the problem that plays the role of the output of the ANN. The output of the ANN includes two values or classes 0 and 1. When the data are not under cyber-attack, the value of the data is actual, and the class of data is labeled 0. In addition, when the data are under cyber-attack, false data is injected into the actual value of the refractive index, and the corresponding class is labeled 1. In other words, the output of the ANN is as follows:

$$Out_C = Class \quad \text{and} \quad Class \in \{0, 1\}. \quad (11)$$

In (11), 0 shows an actual or healthy data, and 1 indicates that the data is under the injection of false data. The second step is to define the input vector of the ANN. Here, the input vector of the ANN is as follows:

$$Inp = [en \quad rp_x \quad ip_x], \quad (12)$$

where, en is the energy of light. Also, rp_x and ip_x are the relevant real part and imaginary part of the refractive index of material x related to en . The third step toward the detection of FDIAs on the refractive index using ANNs is preparing a dataset that can be used for training phase of the ANN and deployed for the operation phase. In Figure 3, the dataset preparation technique is shown. The dataset should include two types of data, i.e., actual

data and wrong data caused by the injection of the false data into the system by cyber-attack. The final input dataset I is defined as follows:

$$I^T = \begin{bmatrix} I_{act}^T & I_{W1}^T & \cdots & I_{Wn-1}^T & I_{Wn}^T \end{bmatrix}, \quad (13)$$

where, I_{act} belongs to the actual input dataset corresponding to the healthy data. In addition, in (13), for $1 \leq v \leq n$, I_{Wv} represents v th wrong dataset that is under cyber-attack. If each input dataset includes κ samples, I_{act} and I_{Wv} are as follows:

$$I_{act} = \begin{bmatrix} en_1 & rp_{x,1} & ip_{x,1} \\ en_2 & rp_{x,2} & ip_{x,2} \\ \vdots & \vdots & \vdots \\ en_{\kappa-1} & rp_{x,\kappa-1} & ip_{x,\kappa-1} \\ en_{\kappa} & rp_{x,\kappa} & ip_{x,\kappa} \end{bmatrix}, \quad (14)$$

and,

$$I_{Wv} = \begin{bmatrix} en_{Wv1} & rp_{x,Wv1} & ip_{x,Wv1} \\ en_{Wv2} & rp_{x,Wv2} & ip_{x,Wv2} \\ \vdots & \vdots & \vdots \\ en_{Wv\kappa-1} & rp_{x,Wv\kappa-1} & ip_{x,Wv\kappa-1} \\ en_{Wv\kappa} & rp_{x,Wv\kappa} & ip_{x,Wv\kappa} \end{bmatrix}. \quad (15)$$

where, in (14), for $1 \leq \zeta \leq \kappa$, en_{ζ} , $rp_{x,\zeta}$ and $ip_{x,\zeta}$ are the ζ th sample of energy of light, the actual real value of the refractive index and the actual imaginary part of the refractive index of the material x , respectively. Also, in (15), $en_{Wv\zeta}$, $rp_{x,Wv\zeta}$ and $ip_{x,Wv\zeta}$ are the energy of light, real part of refractive index, and the imaginary part of the refractive index, respectively, and belongs to the ζ th sample of v th wrong dataset. Therefore, based on (13)–(15), the final inputs dataset is as follows:

$$I = \begin{bmatrix} en_1 & rp_{x,1} & ip_{x,1} \\ en_2 & rp_{x,2} & ip_{x,2} \\ \vdots & \vdots & \vdots \\ en_{\kappa-1} & rp_{x,\kappa-1} & ip_{x,\kappa-1} \\ en_{\kappa} & rp_{x,\kappa} & ip_{x,\kappa} \\ en_{W11} & rp_{x,W11} & ip_{x,W11} \\ en_{W12} & rp_{x,W12} & ip_{x,W12} \\ \vdots & \vdots & \vdots \\ en_{W1\kappa-1} & rp_{x,W1\kappa-1} & ip_{x,W1\kappa-1} \\ en_{W1\kappa} & rp_{x,W1\kappa} & ip_{x,W1\kappa} \\ \vdots & \vdots & \vdots \\ en_{Wn1} & rp_{x,Wn1} & ip_{x,Wn1} \\ en_{Wn2} & rp_{x,Wn2} & ip_{x,Wn2} \\ \vdots & \vdots & \vdots \\ en_{Wn\kappa-1} & rp_{x,Wn\kappa-1} & ip_{x,Wn\kappa-1} \\ en_{Wn\kappa} & rp_{x,Wn\kappa} & ip_{x,Wn\kappa} \end{bmatrix}. \quad (16)$$

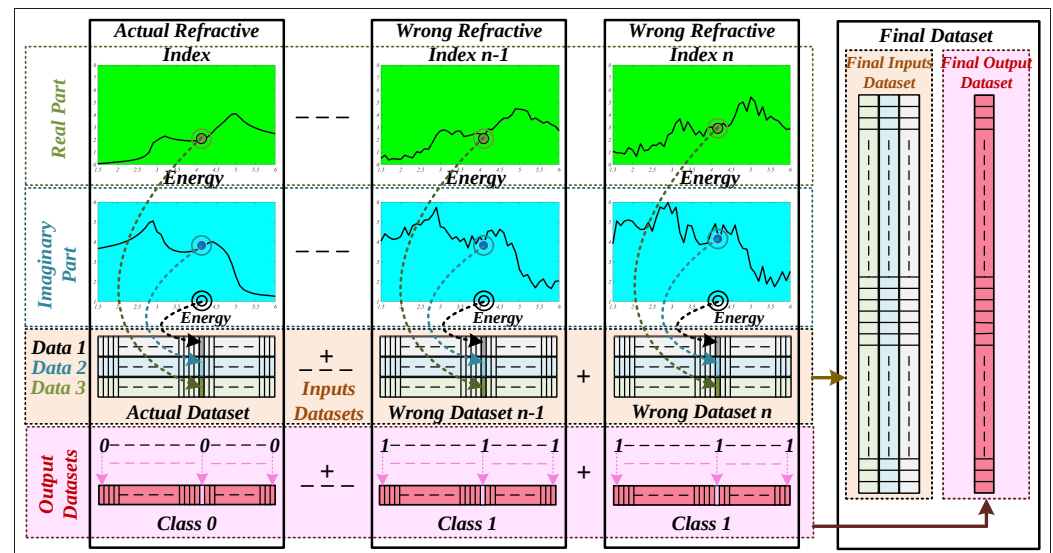


Figure 3. Material dataset preparation using a datasets related to the actual value of the refractive index, and n datasets corresponding to wrong refractive indices.

Taking into account one healthy dataset and n wrong datasets in I , and considering that the number of inputs is 3, the size of matrix I is $(\kappa + \kappa n) \times 3$ including $\kappa + \kappa n$ samples of 3 inputs data.

3.2. Part 2: Output Dataset Preparation

After the preparation of the final inputs dataset, the next step is to define the output dataset. Considering that j th sample for $j \leq \kappa$ belongs to the class of actual data (i.e., $Class = 0$), and for $\kappa + 1 \leq j \leq \kappa + \kappa n$ corresponds to the class of wrong data under the cyber-attack (i.e., $Class = 1$), the final output dataset can be defined as follows:

$$O^T = [O_{act}^T \quad O_{W1}^T \quad \cdots \quad O_{Wn-1}^T \quad O_{Wn}^T]. \quad (17)$$

where,

$$O_{act}^T = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \end{bmatrix}. \quad (18)$$

Besides, for $1 \leq v \leq n$, O_{Wv} is as follows:

$$O_{Wv}^T = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \end{bmatrix}. \quad (19)$$

Therefore, based on (17), (18), and (19), the final output dataset O is calculated as follows:

$$O^T = \begin{bmatrix} 0 & \cdots & 0 & 1 & 1 & 1 & \cdots & 1 \end{bmatrix}. \quad (20)$$

3.3. Part 3: Identification of Cyber-Attacks Using ANN

The goal of this part is to show the calculations of the ANN to identify FDIAs. To implement an ANN for the purpose of the detection of FDIAs, it should be trained. A part of the prepared dataset including the inputs and the output can be used to train the ANN, and the trained ANN can be deployed for examination in the rest of the dataset. In Figure 4, the training and operation (testing unemployed data) phases for the ANN is depicted. Each ANN contains parameters, i.e., weighting co-efficients and bias factors, and to obtain the optimized value of them, a training phase is required. For the training process, a cost

function is required to be used and the goal of the training phase of ANN to detect the cyber-attacks is to find the values of the weighting co-efficients and bias factors in a way to minimize the value of the cost function. If the functionality or operator of the ANN is defined by a function f , the output of that can be 0 or 1 as follows:

$$f(Inp) = Out_C, \quad (21)$$

and in other words,

$$f(en, rp_x, ip_x) = \begin{cases} 0 & \text{for healthy data} \\ 1 & \text{for data under FDIAs} \end{cases}. \quad (22)$$

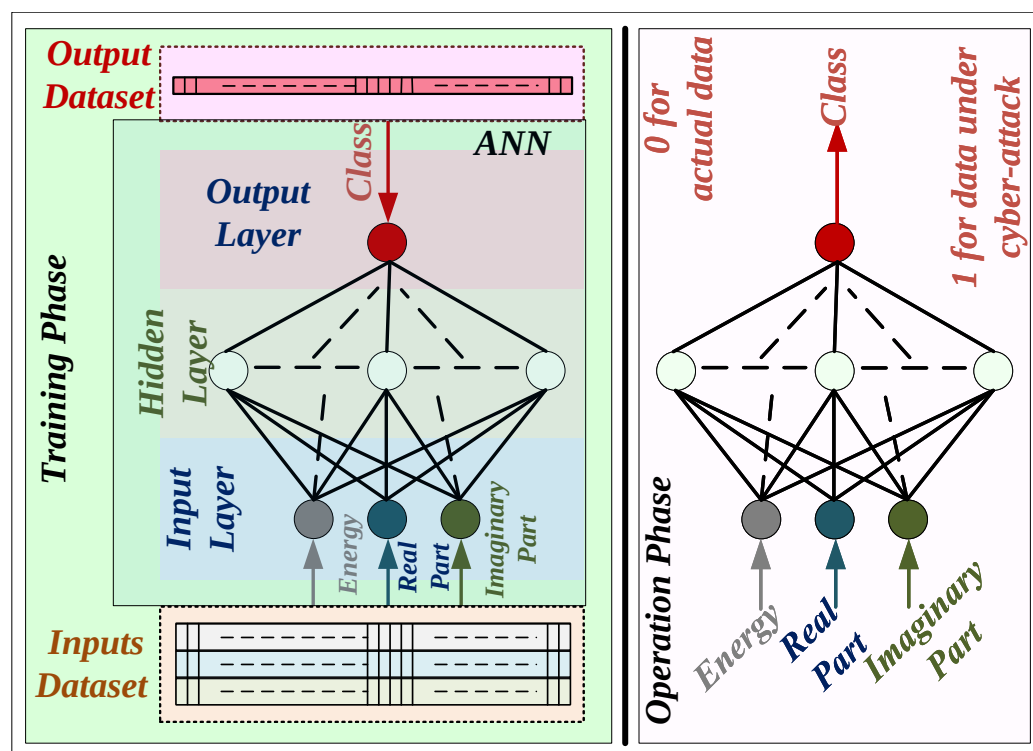


Figure 4. Training and operation phases for the ANN to identify the existence of FDIAs.

In this part, the mathematical expression of $f(en, rp_x, ip_x)$ will be presented. Here, an ANN with only one hidden layer is implemented and, as it will be shown in the next section, this ANN can provide very good results. Therefore, in this part, to avoid issues such as complexity and overfitting, $f(\cdot)$ representing the ANN with one hidden layer is discussed. The input signals of the ANN are the normalized values of the energy of light, the real part of the refractive index, and the imaginary part of the refractive index. Therefore, ANN input signals can be obtained by normalization of the elements of Inp in (12). The input signals of the ANN are presented by Λ as follows:

$$\Lambda = \begin{bmatrix} \lambda_1 & \lambda_2 & \lambda_3 \end{bmatrix}, \quad (23)$$

where, λ_1 is the normalized value of the energy of light. In addition, λ_2 and λ_3 are the normalized values of the real part and the imaginary part of the refractive index. Also, the output signals of the hidden layer can be defined as follows:

$$\Xi = \begin{bmatrix} \xi_1 & \xi_2 & \cdots & \xi_h \end{bmatrix}. \quad (24)$$

In (24), h is the number of neurons in the hidden layer. Furthermore, ξ_i is the output signal of the i th neuron of the hidden layer. Each neuron of the hidden layer receives

the input signals and, using weighting co-efficients and bias factors, calculates the output signal as follows:

$$\xi_i = \Omega\left(\sum_{\alpha=1}^3 (\lambda_{\alpha} \times \iota_{\alpha,i}) + \gamma_i\right). \quad (25)$$

In (25), $\iota_{\alpha,i}$ represents the weighting co-efficient between the α th input of the ANN and the i th neuron of the hidden layer. In addition, γ_i is the bias factor of the i th neuron in the hidden layer. Furthermore, Ω is the activation function of the hidden layer. The neurons of the hidden layer are connected to the output layer of the ANN that has only one neuron. If the activation function of the output layer is θ , the ANN output signal is as follows:

$$Out_C = \theta\left(\sum_{i=1}^h (\xi_i \times \sigma_i) + \mu\right). \quad (26)$$

where, σ_i is the weighting co-efficient between the i th neuron of the hidden layer and the output layer. In addition, μ is the bias factor of the output layer neuron. To detect cyber-attack on refractive index values, the activation function θ returns a value between 0 and 1. Therefore, based on (21), (25) and (26), the functionality of the ANN (i.e., f in (22)) can be calculated as follows:

$$f(en, rp_x, ip_x) = \theta\left(\sum_{i=1}^h \left(\Omega\left(\sum_{\alpha=1}^3 (\lambda_{\alpha} \times \iota_{\alpha,i}) + \gamma_i\right) \times \sigma_i\right) + \mu\right), \quad (27)$$

where, λ_1 , λ_2 , and λ_3 are the normalized values of en , rp_x , and ip_x , respectively. Therefore, the output of the ANN is calculated based on (27). If the calculated value $f(en, rp_x, ip_x)$ is more than 0.5, it returns 1 (the case where cyber-attack is in the system), and if the value is less than 0.5, it returns 0 (the case without cyber-attack). In the next part, the AI-based strategy is summarized.

3.4. Proposed Strategy in Nutshell

Algorithm 1 shows the proposed strategy in summary. The first and the second steps are related to the definition of the inputs vector and the output class of the ANN. Considering that the goal is to detect the cyber-attack, i.e., distinguishing between actual data and the data under cyber-attack, there are two output classes 0 or 1. Where, 0 belongs to actual data, and 1 corresponds to data that is under cyber-attack. In addition, in Step 3, the ANN is trained. After the training phase, the normalized inputs in Step 5 can be implemented to estimate the output class of the ANN. Finally, in Step 6, based on the obtained class, the status of the existence of the cyber-attack can be detected.

Algorithm 1: FDIAs detection for refractive index using AI.

- Step 1. Inputs/Output Definition:** Defining the output classes and the inputs vector based on (11) and (12).
 - Step 2. Inputs Dataset Preparation:** Preparing inputs dataset based on (13).
 - Step 3. Output Dataset Preparation:** Preparing output dataset based on (17).
 - Step 3. Training the ANN:** A part of datasets (i.e., (16) and (20)) is implemented to train the model.
 - Step 4. Normalizing the Inputs:** Calculation of λ_1 , λ_2 , and λ_3 .
 - Step 5. Class Estimation:** Calculation the output of the trained ANN based on (27). If $f(en, rp_x, ip_x)$ is less than 0.5, the estimated class is 0, and if the value of $f(en, rp_x, ip_x)$ is equal or more than 0.5, the estimated class is 1.
 - Step 6. Cyber-attack Detection:** If class is 0, there is no cyber-attack. If the class is 1, there is cyber-attack in the system.
-

4. Results

4.1. GaAs Case Study

In this part, the obtained results related to the identification of FDIAs on dataset of the refractive index of a material, i.e., GaAs, will be presented. The value of the refractive index of the material can be found in [38]. In addition, $n = 2$ and $\kappa = 46$. Also, the number of the hidden layer and the neurons of that are 1 and 7, respectively. For the training, validation, and examination of the proposed strategy, 70%, 15% and 15% of the dataset are implemented, respectively. To evaluate the performance of the ANN, cross-entropy is calculated. Also, to have a better understanding about the results, index $e\%$ as the percentage of the error to calculate the class of the system is used. It is important to note that, the hidden layer implemented the hyperbolic tangent sigmoid activation function, and the neural network is trained using scaled conjugate gradient backpropagation. Training is limited to 1000 epochs and stopped when the gradient is less than 10^{-6} or validation performance failed to improve for six consecutive epochs. The studies are performed in MATLAB R2026a.

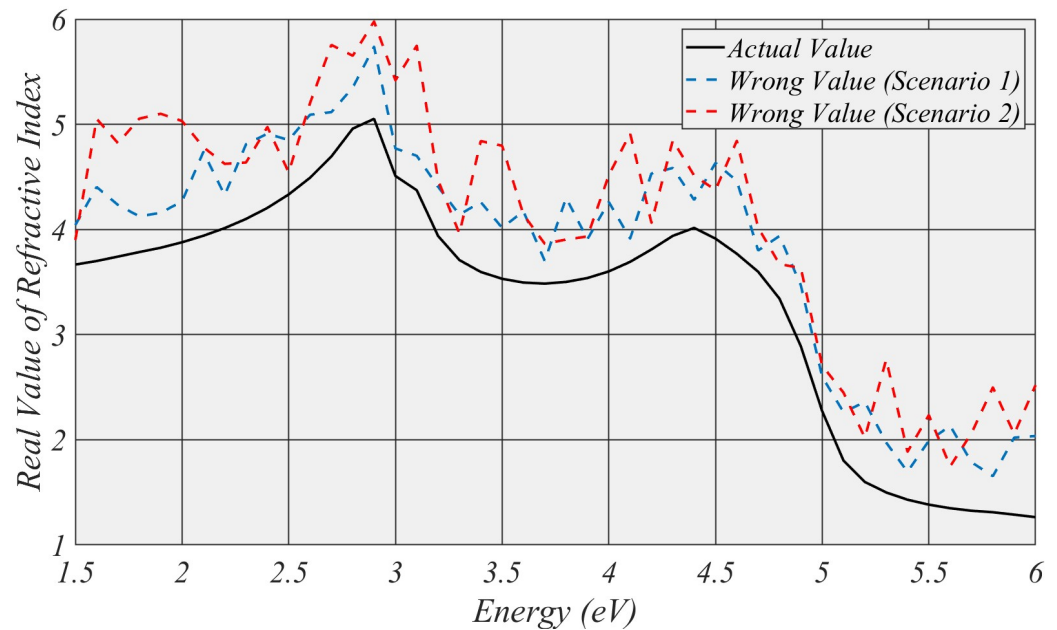
In Figure 5, the actual and wrong values of the refractive index of GaAs are depicted as the function of energy of light. The wrong values are under the cyber-attack by injection false data to them, that made the actual data wrong. To create the wrong datasets, i.e., Scenario 1 and Scenario 2, uniform random numbers with different lower and upper bounds are added into the actual values of the refractive index.

To model the cyber-attacks (i.e., creating the erroneous dataset), two various non-real datasets are constructed using two FDIAs scenarios, i.e., Scenario 1 and Scenario 2. For Scenario 1, two vectors (for the real and the imaginary parts of the refractive index) of 46 elements containing false data using two various uniform distributions with the minimum and the maximum of 0.2 and 0.8 have been implemented. In addition, for Scenario 2, two vectors (one belongs to the real part of the refractive index, and the other one is related to the imaginary part of the refractive index) including 46 elements of false data using two different uniform distributions between 0.2 and 1.4 have been deployed. Then, the created vectors have been added to the refractive index values to construct final version of datasets corresponding to Scenario 1 and Scenario 2.

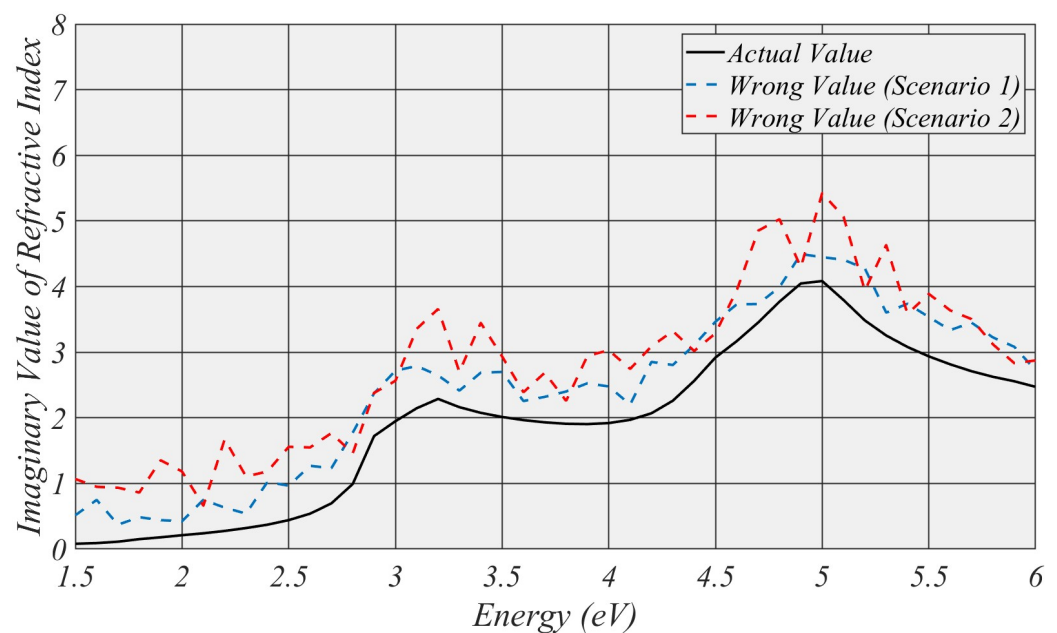
The performance of the ANN for the training, validation, and testing at each epoch is shown in Figure 6. Based on Figure 6, the performance of the validation dataset starts to be increased at epoch number 30, while it is decreasing for the training dataset. Therefore, to avoid overfitting issue, the training is stopped at epoch number 36, and the weighting co-efficients and the bias factors corresponding to epoch 30 are selected for the ANN model. In Figure 7, the confusion matrices are shown. Based on Figure 7, for the training dataset including 96 samples, 93 samples including 0 and 1 are estimated successfully. Also, for 21 samples of the validation dataset, the ANN successfully estimates the class of all samples. Furthermore, for the 21 implemented samples of the examination phase, the AI-based strategy estimate the class of 20 samples effectively. For all samples including testing, validation, and testing data, the error of the calculation is 2.9%. In Table 1, the values of the performance and e corresponding to the training, validation, and testing samples are shown. Based on Table 1, the percentage of the error of the calculation, i.e., e , for the testing phase is 4.76%.

In Table 2, more indicators related to the examination of the ANN on the testing dataset is shown. Based on Table 2, the accuracy is 0.9524, and in other words, the ANN could detect the class of 95.24% of testing samples correctly. Based on Table 2, the value of recall is 1. Therefore, the trained ANN could find 100% of the attacked testing samples successfully. As another parameter, the value of precision is involved in Table 2 with the achieved value of 0.9444. Based on this indicator, among all cyber-attack detections on

the testing dataset, 94.44% of the detections are correct. Based on the obtained results, the proposed AI-based strategy can detect the existence of the FDIAs successfully.



(a) The real part of the refractive index of GaAs.



(b) The imaginary part of the refractive index of GaAs.

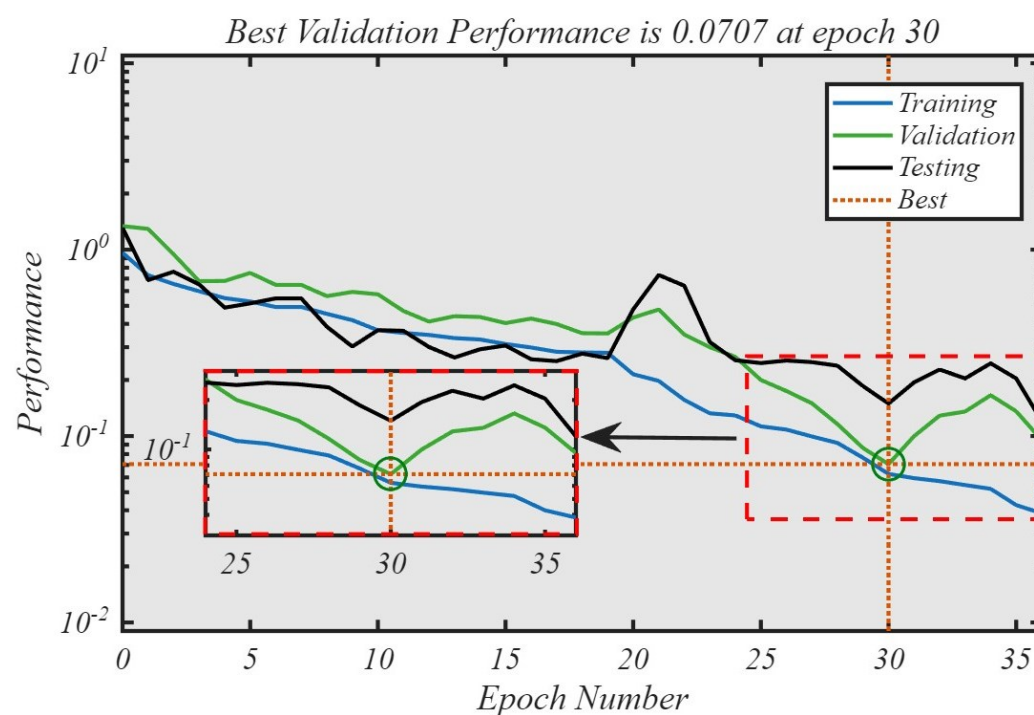
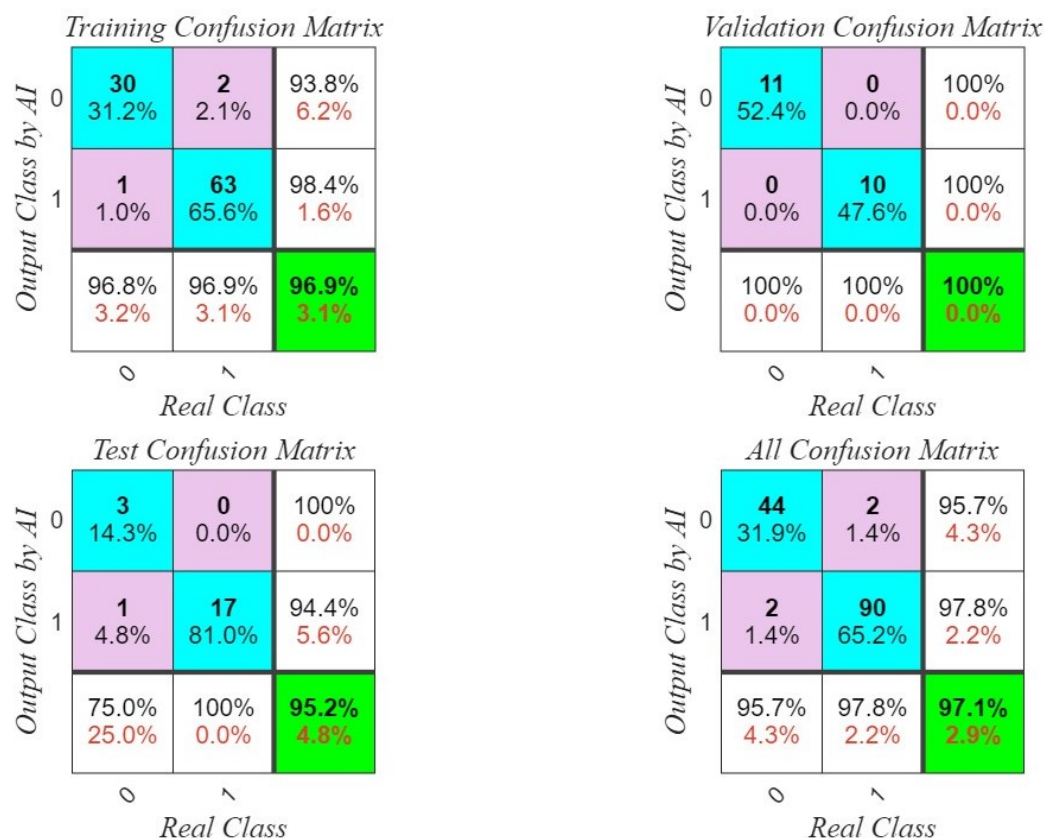
Figure 5. The representation of: (a) dataset related to the real value of the refractive index of GaAs, and (b) dataset related to the imaginary value of the refractive index of GaAs.

Table 1. Information of training, validation, and testing phases for the ANN.

	<i>e</i> % (Error of the Calculations)	Performance
Training	3.12	0.0627
Validation	0	0.0707
Testing	4.76	0.1494

Table 2. Values of different indicators related to the examination of the trained ANN on testing dataset.

Indicator	F1 Score	Recall	Accuracy	Precision
Value	0.9714	1.0000	0.9524	0.9444

**Figure 6.** The performance of the ANN for the training, validation, and testing stages for different epochs.**Figure 7.** The confusion matrices related to the ANN to classify data.

4.2. Several Experiments Analysis

In this part, different runs for ANNs with a hidden layer are made, calling here an experiment for each single run to evaluate an ANN. The studied ANNs have various numbers of hidden neurons from 1 for the simplest ANN to 60 neurons for the largest ANN. Therefore, totally 60 ANNs are studied. In addition, for each ANN, 150 experiments are done, and as a result, totally $60 \times 150 = 9000$ experiments are made. Table 3 includes the values of the average of the performances for the testing phases, the best testing performance, the average of testing phases accuracy, and the best accuracy. Based on Table 3, the best values of the testing performance and accuracy among all 9000 experiments are 2.6445×10^{-8} and 1, respectively. In addition, the average of the testing performances and testing accuracies of all experiments are 0.2903 and 0.8802, respectively.

Table 3. Average of testing performance, the best performance, average of testing accuracy, and the best accuracy for 9000 experiments (150 experiments for ANNs with different hidden neurons from 1 to 60 neurons, i.e., $150 \times 60 = 9000$ experiments).

Indicator	Value
Average of Testing Performances	0.2903
Best Testing Performance	2.6545×10^{-8}
Average of Testing Accuracies	0.8802
Best Accuracy	1

In Figure 8, the performance-based values of the experiments are shown. The minimum value of the testing performance across 150 experiments for the ANNs with 6 or more hidden neurons is very small and close to zero. In addition, the maximum value of the testing performance is occurred for the ANN with 27 hidden neurons. The second highest testing performance belongs to the one of the experiments related to the ANN with 18 hidden neurons. The highest average value of the testing performances is obtained by the ANN with only one hidden neurons.

The values related to the testing F1 score of the experiments for the ANNs with different numbers of the hidden neurons are depicted by Figure 9. Based on Figure 9, the maximum F1 score across the 150 experiments of all ANNs is 1 except two ANNs with 1 and 3 hidden neurons. In other words, at least for one of experiment among all 150 experiments to evaluate the ANN with l hidden neurons for $l \in L$ and $L = \{2\} \cup \{4, 5, 6, \dots, 57, 58, 59, 60\}$, the existence of the cyber-attacks on all attacked samples of the testing dataset have been detected correctly. In addition, the average of F1 score for the ANNs with hidden neurons of less than 13 is smaller than 0.9. Also, for the ANNs with 13 and 14 hidden neurons, the average of F1 score is more than 90%, for the ANN with 15 hidden neuron is less than 90%, and for the ANNs with the numbers of hidden neurons more than 15 is larger than 90%. The minimum value of F1 score has been happened for the ANN with 47 hidden neurons.

The minimum, maximum, average, and standard deviation of recall indicator during the 150 experiments for different numbers of neurons are shown by Figure 10. For all case studied, i.e., different numbers of neurons, the maximum value of recall is 1. In other words, for each ANN, at least one experiment with 100% successful detection of the attacked samples of the testing dataset has been happened. Also, the average value of recall for all ANNs is larger than 0.8, and for the ANNs with more than 3 hidden neurons is larger than 0.9. The average value of recall for the ANNs with more than 12 hidden neurons is larger than 0.93. Besides, the minimum value of recall was happened on one of experiments related to the ANN with 47 hidden neurons.

Finally, Figure 11 illustrates the minimum, maximum, average, and standard deviation of the testing phase accuracy across 150 experiments for each ANN. Based on Figure 11,

for all the ANNs except the ANNs with 1 and 3 hidden neurons, the maximum accuracy indicator is 1, and in other words, the accuracy of 100 % was happened during the experiments. In addition, for the ANNs with 1 and 3 hidden neurons, the maximum testing accuracy is more than 0.95. Further, the average of the testing accuracy has the maximum value for the ANN with 43 hidden neurons.

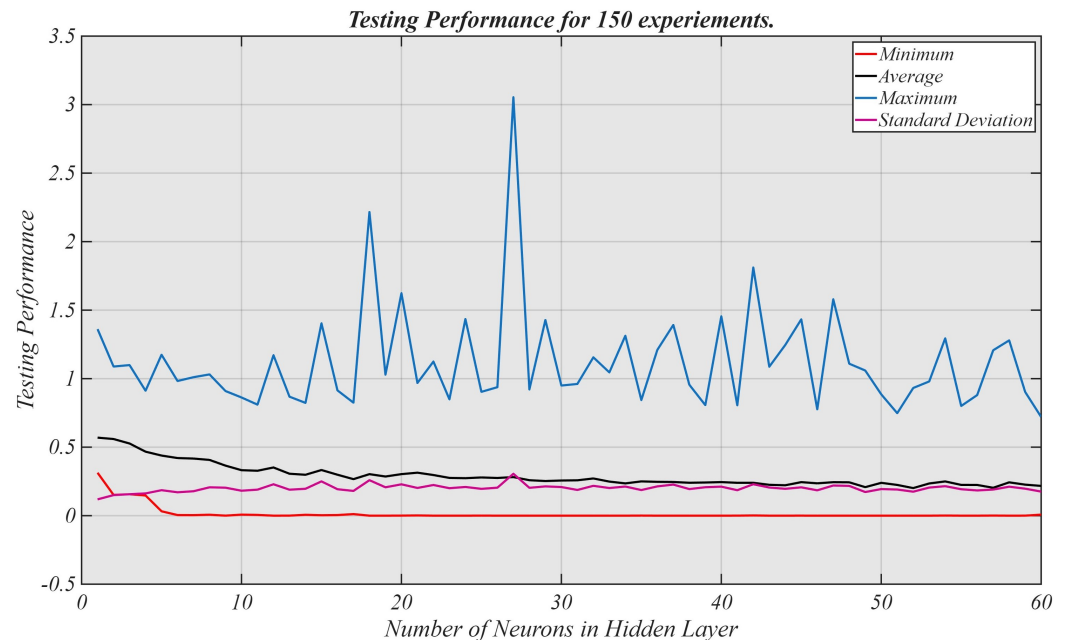


Figure 8. Testing performance visualization across 150 independent experiments for varying numbers of hidden neurons from 1 to 60 (i.e., totally $150 \times 60 = 9000$ experiments). Here, the minimum, the maximum, the average, and the standard deviation of the testing performance value of ANNs with different sizes (i.e., numbers of neurons in the hidden layer) are shown.

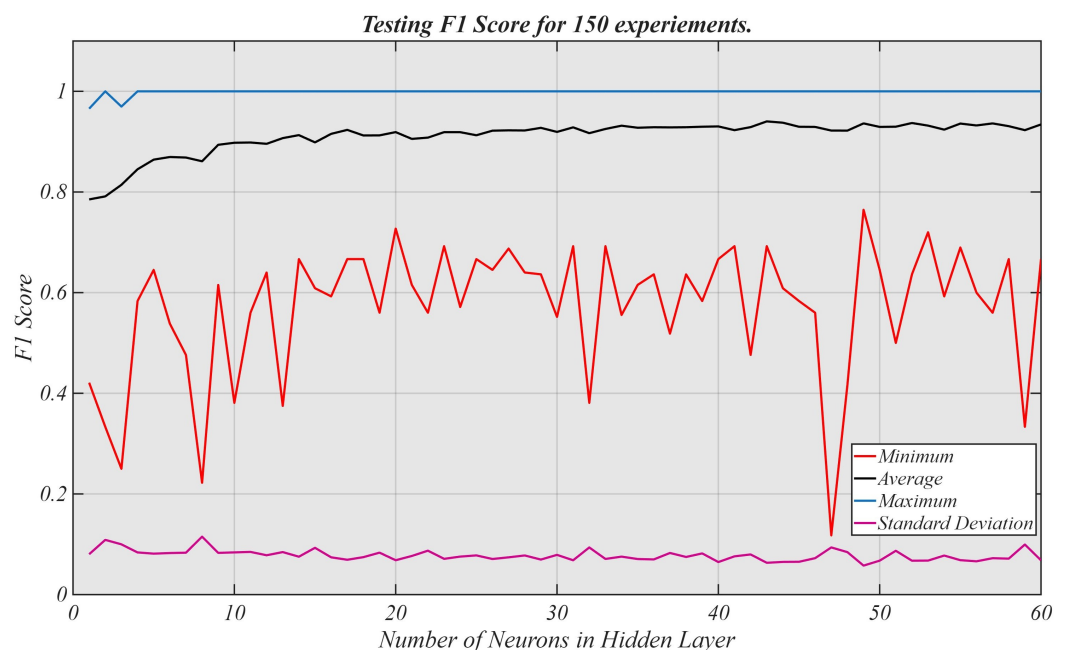


Figure 9. Testing F1 score visualization across 150 independent experiments for varying numbers of hidden neurons from 1 to 60 (i.e., totally $150 \times 60 = 9000$ experiments). Here, the minimum, the maximum, the average, and the standard deviation of the testing F1 score value of ANNs with different sizes (i.e., numbers of neurons in the hidden layer) are shown.

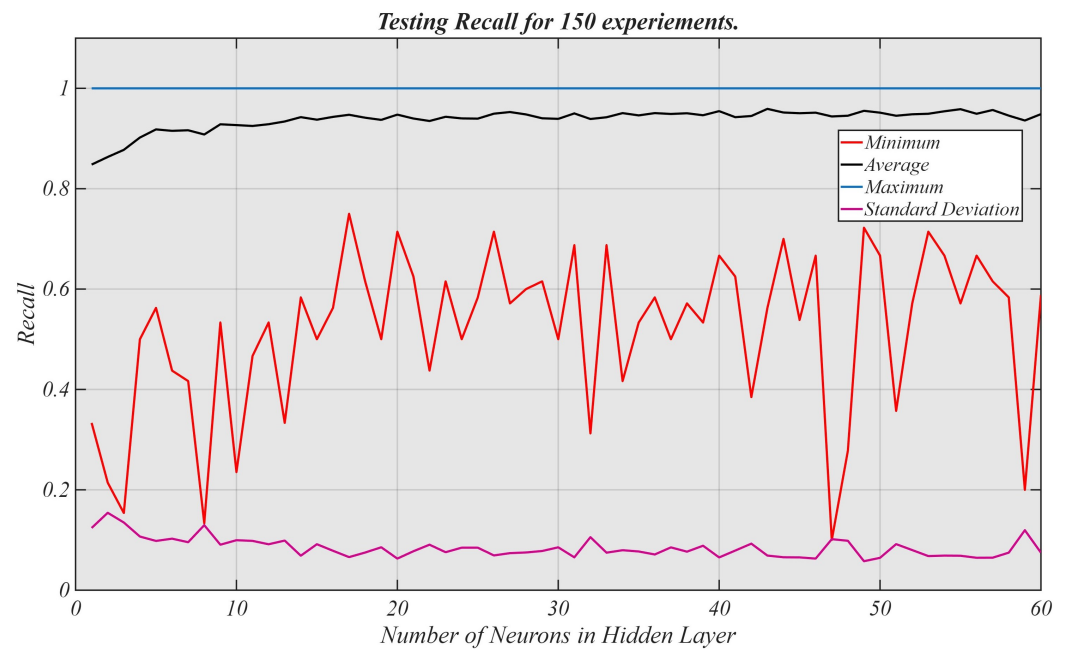


Figure 10. Testing recall visualization across 150 independent experiments for varying numbers of hidden neurons from 1 to 60 (i.e., totally $150 \times 60 = 9000$ experiments). Here, the minimum, the maximum, the average, and the standard deviation of the testing recall value of ANNs with different sizes (i.e., numbers of neurons in the hidden layer) are shown.

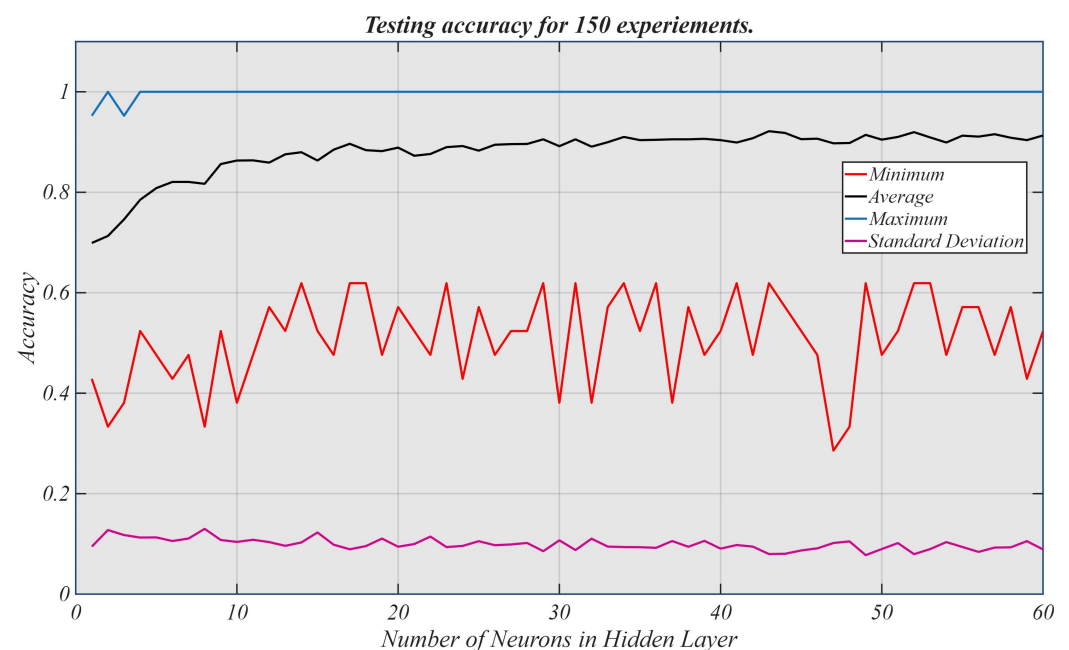


Figure 11. Testing accuracy visualization across 150 independent experiments for varying numbers of hidden neurons from 1 to 60 (i.e., totally $150 \times 60 = 9000$ experiments). Here, the minimum, the maximum, the average, and the standard deviation of the testing accuracy value of ANNs with different sizes (i.e., numbers of neurons in the hidden layer) are shown.

4.3. Effect of Normalization

For this part, the effect of the normalization on the proposed cyber-attack detection approach is investigated. Here, two normalization techniques, i.e., min-max and z-score, are implemented. In addition, 20 ANNs with various numbers of hidden neurons from 1 to 20 are studied. Furthermore, for each ANN using each normalization technique, 100 independent experiments (i.e., evaluating the training, validation, and testing phases)

is done. Therefore, totally $2 \times 100 \times 20 = 4000$ experiments are done, where 2000 of them belongs to the min-max normalization, and the other 2000 experiments are related to the z-score normalization.

In Table 4, information regarding the findings of this case study are summarized, i.e., our findings about the averages of the testing performance and accuracy, best value of the testing performance, and the maximum obtained accuracy under each of the normalization techniques. As it is mentioned in Table 4, the average values related to the testing performance for min-max and z-score cases are 0.3798 and 0.3737, respectively. In addition, the best achieved testing performance for the case of min-max normalization is 1.7678×10^{-6} , and for the case of z-score normalization is 1.5986×10^{-6} . The results show that in the case of the performance of the ANNs under the testing phase, z-score could reach out to better results. However, the achieved results are very close to each other, and as a result, it can be concluded that both methods may have very similar performances during several experiments under the testing phase.

For the case of accuracy indicator, the average of this indicator for the cases of min-max and z-score normalizations are 83.55 % and 84.27 %, respectively. therefore, the average of the accuracy achieved under the z-score normalization is about 0.72 % more than the average of the accuracy belongs to the testing under min-max score normalization. For both normalization techniques, they could achieved at least one time during testing phase experiments to the accuracy of 100 %. Based on the achieved average and best accuracy indicators, z-score-based normalization has shown more accurate results, but, it should be consider that the results are very close, i.e., same best accuracy 100 %, and only 0.72% difference in the average of the accuracies.

Table 4. Average of testing performance, the best testing performance, average of testing accuracy, and the best testing accuracy of 2000 experiments (100 experiments for ANNs with different hidden neurons from 1 to 20 neurons, i.e., $100 \times 20 = 2000$ experiments) for each normalization technique. Totally, $2 \times 2000 = 4000$ experiments are deployed.

	Min-Max	Z-Score
Average of Testing Performances	0.3798	0.3737
Best Testing Performance	1.7678×10^{-6}	1.5986×10^{-6}
Average of Testing Accuracies	0.8355	0.8427
Best Accuracy	1	1

More detailed results of testing corresponding to performance, F1 score, recall, and accuracy related to the two mentioned normalization techniques are depicted by Figure 12, Figure 13, Figure 14 and Figure 15, respectively.

Based on Figure 12, both methods have shown very close behavior corresponding to the standard deviation and average of testing performance results for various numbers of hidden neurons. In addition, for the min-max normalization scenario, the highest testing performance has been occurred during an experiment belong to the ANN with 18 hidden neuron. The highest testing performance corresponding to the z-score normalization technique is happened for the ANN with 6 hidden neurons. For both normalization scenarios, the achieved minimum performance is very close to zero value for the ANN with more than 4 hidden neurons.

In the case of F1 score indicator, for the maximum achieved value based on Figure 13, min-max normalization-based ANNs could achieved to the value of 1 for all cases except the ANNs with 1 and 3 hidden neurons. In addition, all ANNs under the z-score could achieved F1 score with value of 1 excluding the ANNs with 1, 2, and 3 hidden neurons. Based on Figure 13, for the obtained average F1 score and the standard deviation of that,

both normalizations have very close values for the ANNs with different numbers of the hidden neurons. For both normalization techniques, the achieved minimum F1 score values are occurred for the ANNs with 8 hidden neurons.

In Figure 14, the results corresponding to the recall indicator show the maximum recall value of 1 for all the ANNs and for both the normalization techniques. Considering Figure 14, both normalization techniques could achieved to very close values for the average values and the standard deviation values of recalls. The minimum recall values for the both normalization techniques is happening for the ANN with 8 hidden neurons.

Finally, based on Figure 15, for the testing accuracy indicator, both normalization techniques could achieved to very close average and standard deviation values. In addition, min-max normalization technique could achieved to the maximum best accuracy value 100% for all the ANNs excluding the ANNs with 1 and 3 hidden neurons, and the z-score normalization techniques could have the maximum accuracy 100% for the ANNs with 4 or more hidden neurons. Furthermore, both normalization techniques have the minimum accuracy during the experiments under 8 hidden neurons.

To conclude this part, for the testing phase, the average values and standard deviation values of all the depicted indicators (performance, F1 score, recall, and accuracy) are very close to each other. In addition, for the case of F1 score, recall, and accuracy indicators, the minimum values happens for both normalization techniques for the ANN with 8 hidden neurons. Besides, the achieved maximum recall for all the ANNs is 1. The maximum F1 score value under the min-max normalization technique with 2, 4 or more hidden neurons is 1, and this value for the z-score normalization technique is one if the ANN has 4 or more hidden neurons. Finally, for both min-max and z-score normalization techniques, the ANNs with 4 or more hidden neurons could achieved the 100 % accuracy value during the experiments. Also, for the case of min-max normalization technique, during the experiments of the ANN with 2 hidden neurons the achieved maximum accuracy is 100%.

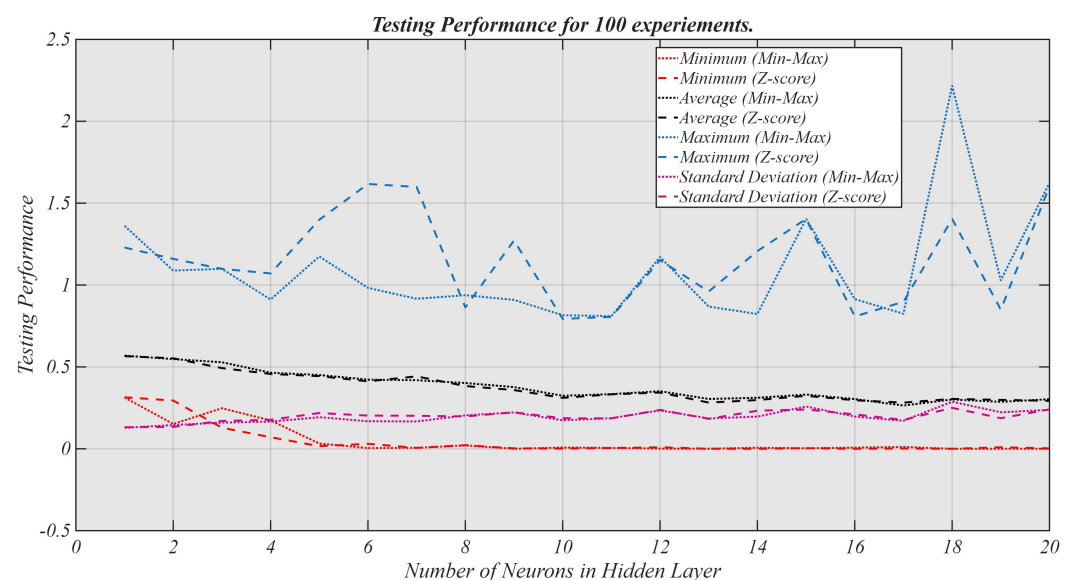


Figure 12. Testing performance visualization based on two types of normalization techniques. For each normalization technique, 100 independent experiments for varying numbers of hidden neurons from 1 to 20 have been approached (totally $2 \times 1000 \times 20 = 4000$ experiments). Here, the minimum, the maximum, the average, and the standard deviation of the testing performance value of ANNs with different sizes (i.e., numbers of neurons in the hidden layer) for both normalization techniques are shown.

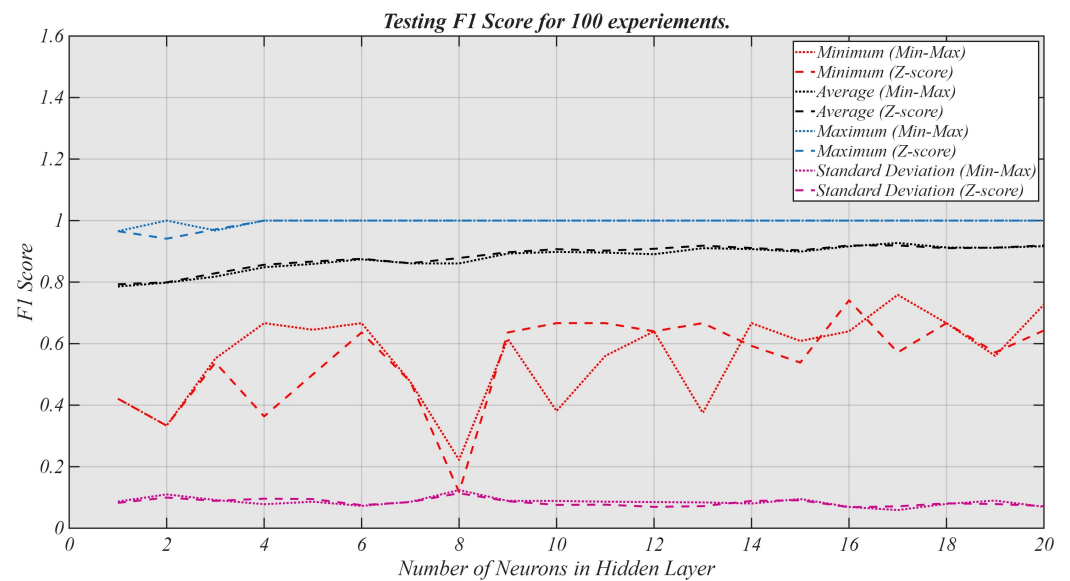


Figure 13. Testing F1 score visualization based on two types of normalization techniques. For each normalization technique, 100 independent experiments for varying numbers of hidden neurons from 1 to 20 have been approached (totally $2 \times 1000 \times 20 = 4000$ experiments). Here, the minimum, the maximum, the average, and the standard deviation of the testing F1 score value of ANNs with different sizes (i.e., numbers of neurons in the hidden layer) for both normalization techniques are shown.

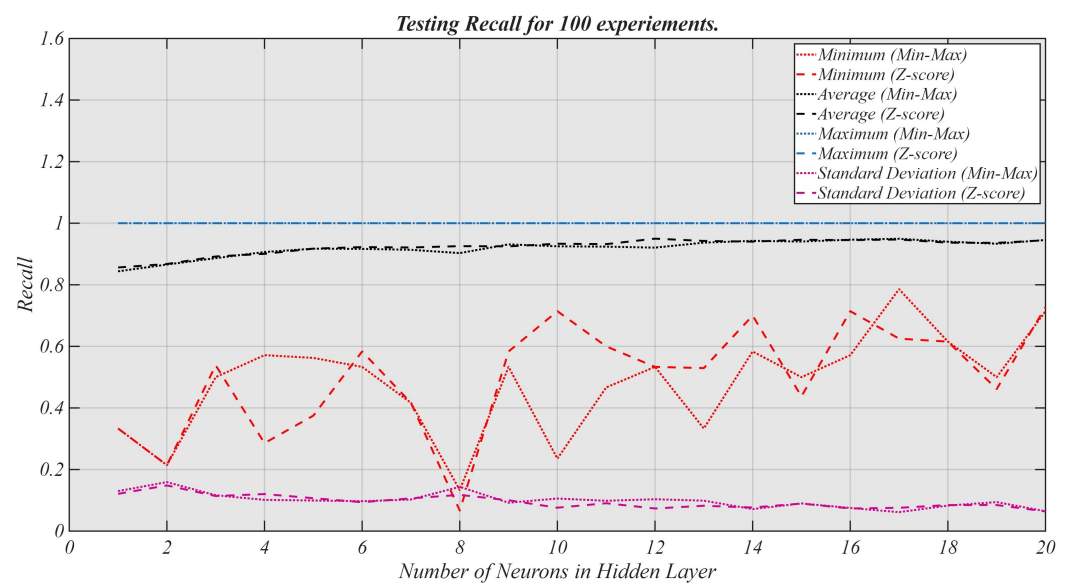


Figure 14. Testing recall visualization based on two types of normalization techniques. For each normalization technique, 100 independent experiments for varying numbers of hidden neurons from 1 to 20 have been approached (totally $2 \times 1000 \times 20 = 4000$ experiments). Here, the minimum, the maximum, the average, and the standard deviation of the testing recall value of ANNs with different sizes (i.e., numbers of neurons in the hidden layer) for both normalization techniques are shown.

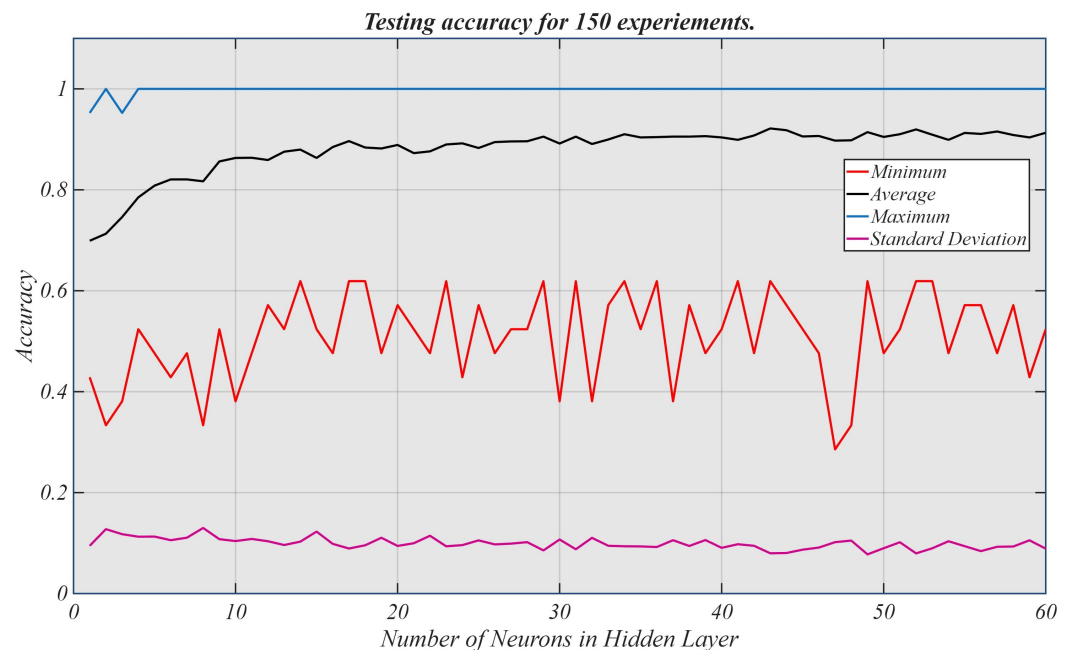


Figure 15. Testing accuracy visualization based on two types of normalization techniques. For each normalization technique, 100 independent experiments for varying numbers of hidden neurons from 1 to 20 have been approached (totally $2 \times 1000 \times 20 = 4000$ experiments). Here, the minimum, the maximum, the average, and the standard deviation of the testing accuracy value of ANNs with different sizes (i.e., numbers of neurons in the hidden layer) for both normalization techniques are shown.

4.4. Comparison: Variations in the Number of Hidden Neurons

The goal of this part is to show the effect of the number of hidden neurons to identify the cyber-attacks. In this part, same training, validation, and testing datasets are used for the cyber-attack detection purpose, using different ANNs with various number of hidden neurons. Here, ANNs with 1, 2, 4, 8, 12, 16, and 20 hidden neurons are evaluated. To compare the ANNs, the performance and $e\%$ of the training, validation, and testing phases for all ANNs are calculated. Here, cross-entropy function is used to indicate the performance of the ANNs.

The performance and $e\%$ for the ANNs during the training, validation, and testing phases are included in Table 5. Based on Table 5, the ANN with 4 neurons has the best training, validation, and testing performances, i.e., 0.3188, 0.2576, and 0.3787, respectively. In addition, for the training and validation phases, the best values of $e\%$ belongs to the ANN with 4 hidden neurons. About the testing phase, the best value of $e\%$ is achieved by the ANNs with 4 and 12 hidden neurons. For the case of precision, the highest value is occurred when the ANN with the 4 hidden neuron is used. The ANN with 12 hidden neurons has the second highest precision value, i.e., 0.8750.

Table 6, involved the calculations of different indicators to evaluate the trained ANNs corresponding to the testing dataset, i.e., F1 score, recall, accuracy, and precision. The highest recall value 0.8750 is achieved by the ANNs with 8, 10, and 12 hidden neurons. In the case of precision, the ANNs with 4 and 12 hidden neurons have the first rank value 0.9286 and the second rank value 0.8750, respectively. The ANNs with 4 and 12 hidden neurons also have the highest accuracy value that is 0.8095. The first highest F1 score is captured by the ANN with 12 hidden neurons. In addition, the second and the third highest F1 score belong to the ANN with 4 and 8 hidden neurons with values of 0.8667 and 0.8485, respectively.

To conclude this part, when considering only the performance, the ANN with 4 hidden neurons is the best choice when that is compared to other ANNs with 2, 6, 8, 10, 12, 14, 16, and 18 hidden neurons. But, if only the testing $e\%$ is considered, the ANNs with 4 and 12 hidden neurons can be the proper choices. In addition, if F1 score is selected as the only selected indicator, the ANNs with 12 hidden neurons can be the first selection and the second and the third choices can be the ANNs with 4 and 8 hidden neurons, respectively.

Table 5. Information (performance and $e\%$) of training, validation, and testing phases for the ANNs with one hidden layer including different numbers of hidden neurons, i.e., h .

h	Training		Validation		Testing	
	Performance	$e\%$	Performance	$e\%$	Performance	$e\%$
2	0.5161	29.17	0.5196	23.81	0.5945	38.10
4	0.3188	11.46	0.2576	9.52	0.3787	19.05
6	0.3731	19.79	0.4972	14.29	0.4905	28.57
8	0.4214	17.71	0.5059	23.81	0.4710	23.81
10	0.4902	27.08	0.5753	33.33	0.5654	28.57
12	0.4054	18.75	0.4942	19.05	0.5224	19.05
14	0.4414	22.92	0.5010	19.05	0.5559	33.33
16	0.3576	14.58	0.4829	23.81	0.4310	23.81
18	0.4486	20.83	0.4168	19.05	0.5038	23.81

Table 6. Values of different indicators related to the examination of the trained ANNs with different numbers of hidden neurons on testing dataset.

h	F1 Score	Recall	Accuracy	Precision
2	0.7500	0.7500	0.6190	0.7500
4	0.8667	0.8125	0.8095	0.9286
6	0.8125	0.8125	0.7143	0.8125
8	0.8485	0.8750	0.7619	0.8235
10	0.8235	0.8750	0.7143	0.7778
12	0.8750	0.8750	0.8095	0.8750
14	0.7586	0.6875	0.6667	0.8462
16	0.8387	0.8125	0.7619	0.8667
18	0.8387	0.8125	0.7619	0.8667

5. Comparison of Different Machine Learning Models

In this part, a comparison among different machine learning models will be done. Four various models are considered for the evolution, i.e., decision tree, logistic regression, support vector machine, and ANNs. Here, for the validation, k-fold cross-validation ($k = 6$) is implemented. In addition, 20% of dataset is used for the testing phase. The training and testing phases include 111 and 27 samples, respectively. Besides, three decision trees, a logistic regression, four support vector machines, and 5 ANNs models with different hyperparameters are deployed.

In Table 7, the information of the implemented decision tree models is shown. Tables 8 and 9 include the information of the deployed logistic regression model, and the four implemented support vector machines, respectively. Based on Table 7, three decision tree models namely DT_1 , DT_2 , and DT_3 with different maximum number of splits (i.e., 60, 20, and 9, respectively) are implemented. For the logistic regression model, as it is shown by Table 8, the binary generalized model is used and the regression model is called LR . Also, considering Table 9, four support vector machines are deployed, which are called SVM_1 , SVM_2 , SVM_3 , and SVM_4 . The Kernel functions of SVM_1 , SVM_2 , SVM_3 , and SVM_4 are linear, quadratic, cubic, and gaussian, respectively. In addition, five ANN models namely

ANN_1 , ANN_2 , ANN_3 , ANN_4 , and ANN_5 are deployed in this part. The information of the implemented ANNs of this part can be found in Table 10. For ANN_1 , ANN_2 , and ANN_3 , only one hidden layer is implemented with 9, 30, and 60 hidden neuron numbers, respectively. In addition, ANN_4 includes two hidden layers with 9 hidden neurons for each hidden layer, i.e., {9, 9}. Finally, ANN_5 is made using three hidden layers with the set of {9, 9, 9} for the hidden neurons.

Therefore, totally 13 machine learning models are evaluated here as follows:

- Three Decision Trees: Namely DT_1 , DT_2 , and DT_3 .
- One Logistic Regression: Namely LR .
- Four Support Vector Machines: Namely SVM_1 , SVM_2 , SVM_3 , and SVM_4 .
- Five ANNs: Namely ANN_1 , ANN_2 , ANN_3 , ANN_4 , and ANN_5 .

In Figure 16, the percentage of the accuracy for the testing phase of the implemented machine learning models is shown. Based on Figure 16, ANN_4 has the maximum value of the accuracy, i.e., 100% of accuracy for the testing dataset. The second highest accuracy is achieved while SVM_3 is examined on the testing dataset. The third highest accuracy percentage belongs to SVM_4 , ANN_1 , ANN_2 , and ANN_3 . In addition, Figure 17 illustrates the accuracy percentage for the validation of the mentioned machine learning models of this part. Based on Figure 17, the highest accuracy is happened on ANN_3 . Besides, the minimum of accuracy is achieved by LR model.

Table 7. The information of the implemented decision tree models, i.e., their names, and the splits maximum number of each model.

Machine Learning Model	Name	Splits Maximum Number
Decision Tree	DT_1	60
Decision Tree	DT_2	20
Decision Tree	DT_3	9

Table 8. The information of the implemented logistic regression model, i.e., the name, and the type of the model.

Machine Learning Model	Name	Type
Logistic Regression	LR	Binary Generalized Model

Table 9. The information of the implemented support vector machine models, i.e., their names, and the Kernel function of each model.

Machine Learning Model	Name	Kernel Function
Support Vector Machine	SVM_1	Linear
Support Vector Machine	SVM_2	Quadratic
Support Vector Machine	SVM_3	Cubic
Support Vector Machine	SVM_4	Gaussian

Table 10. The information of the implemented ANN models, i.e., their names, numbers of hidden layers, and the numbers of hidden neurons for the hidden layers of each model.

Machine Learning Model	Name	Numbers of Hidden Layers	Numbers of Hidden Neurons
Artificial Neural Network	ANN_1	1	{9}
Artificial Neural Network	ANN_2	1	{30}

Table 10. Cont.

Machine Learning Model	Name	Numbers of Hidden Layers	Numbers of Hidden Neurons
Artificial Neural Network	ANN_3	1	{60}
Artificial Neural Network	ANN_4	2	{9,9}
Artificial Neural Network	ANN_5	3	{9,9,9}

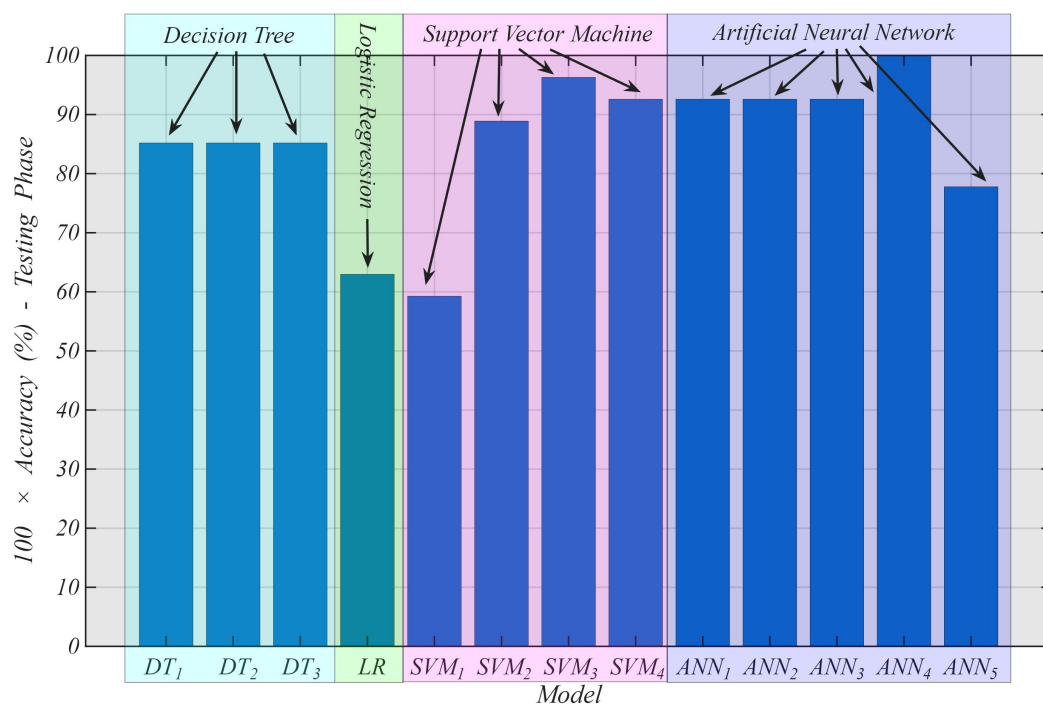


Figure 16. The percentage value of the accuracy (i.e., $100 \times \text{Accuracy}$) related to the testing phase of different implemented machine learning models.

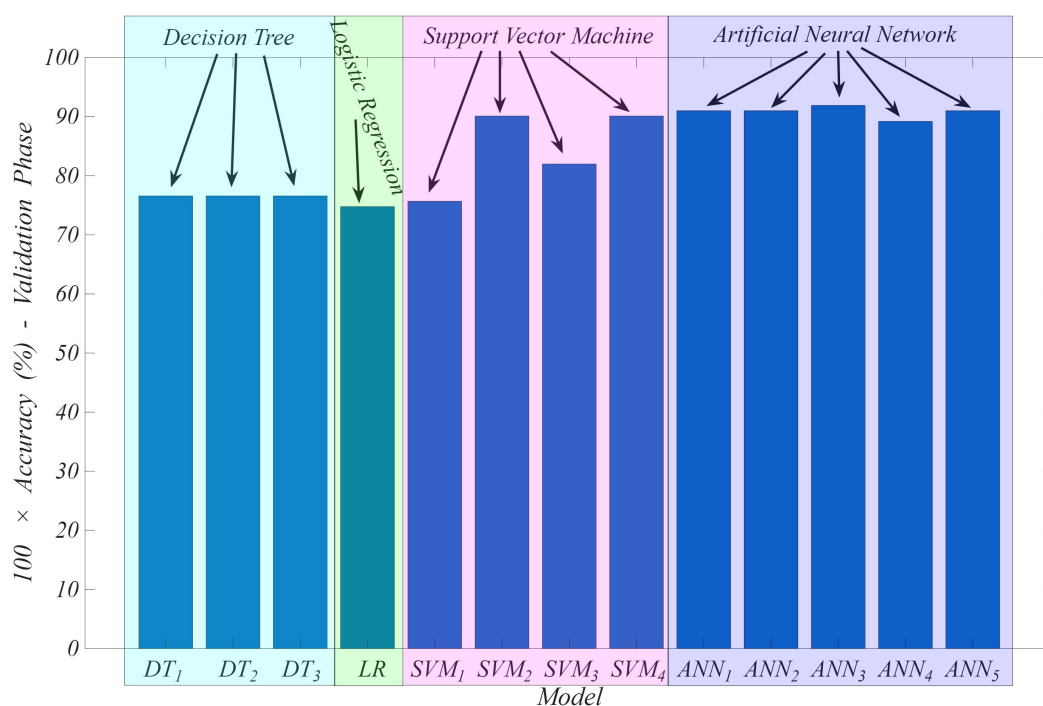


Figure 17. The percentage value of the accuracy (i.e., $100 \times \text{Accuracy}$) related to the validation phase of different implemented machine learning models.

Table 11, includes the quantified information of Figures 16 and 17. Based on Table 11, ANN_3 has the best validation accuracy with value of 0.9189. In addition, the second rank of validation accuracy is 0.9099 belongs to ANN_1 , ANN_2 , and ANN_5 . The lowest validation accuracy is captured by LR model with value of 0.7477. Furthermore, the second lowest validation accuracy is 0.7658, which is for all decision tree models, i.e., DT_1 , DT_2 , and DT_3 . For the case of testing accuracy, ANN_4 has the highest testing accuracy with value of 1, and SVM_3 has the second best value, i.e., 0.9630. Also, the third highest testing accuracy is achieved by SVM_4 , ANN_1 , ANN_2 , and ANN_3 with the value of 0.9259.

Table 11. The results obtained with different types of machine learning models.

Model Name	Validation Accuracy	Testing Accuracy	Validation Total Cost	Testing Total Cost
DT_1	0.7658	0.8519	26	4
DT_2	0.7658	0.8519	26	4
DT_3	0.7658	0.8519	26	4
LR	0.7477	0.6296	28	10
SVM_1	0.7568	0.5926	27	11
SVM_2	0.9009	0.8889	11	3
SVM_3	0.8198	0.9630	20	1
SVM_4	0.9009	0.9259	11	2
ANN_1	0.9099	0.9259	10	2
ANN_2	0.9099	0.9259	10	2
ANN_3	0.9189	0.9259	9	2
ANN_4	0.8919	1	12	0
ANN_5	0.9099	0.7778	10	6

To conclude this part, 13 machine learning models including three decision trees, a logistics regression, four support vector machines, and 5 ANNs with different hyperparameters have been examined. For the validation, SVM_2 , SVM_4 , ANN_1 , ANN_2 , ANN_3 , and ANN_5 have shown the accuracies with values of higher than 0.9 or in other words more than 90%. In addition, for the testing phase of the machine learning models, the accuracy of SVM_3 , SVM_4 , ANN_1 , ANN_2 , ANN_3 , and ANN_4 were more than 0.9. Therefore, SVM_4 , ANN_1 , ANN_2 , and ANN_3 achieved the accuracies which are more than 0.9 for both validation and testing phases. So, the set of $\{SVM_4, ANN_1, ANN_2, ANN_3\}$ can involve the best candidates. Considering both testing and validation accuracy values, ANN_3 can be the best choice due to having the minimum deviation between the validation accuracy and testing accuracy values that can make it a more reliable machine learning model for the selection in this scenario. Based on the obtained results, in this scenario all the evaluated ANNs with only one hidden layer (i.e., ANN_1 , ANN_2 , and ANN_3) were among the best candidates for the cyber-attack detection. Also, ANN_3 with one hidden layer have shown the most reliable results considering the rank of the accuracies and the difference between the validation and testing accuracies. In addition, SVM_4 , ANN_1 , and ANN_2 have shown a small difference between the validation and testing accuracies that can suggest them as reliable models as well.

6. Discussion

This study has addressed the cybersecurity challenge in quantum cyber-physical systems by harnessing the capability of AI to model the system as a classification framework

for distinguishing between healthy and attacked data. In this study, a binary strategy has been deployed for the classification purposes to detect cyber-attacks targeting refractive index values, which can play an important role in quantum cyber-physical systems. Here, the real and imaginary parts of the refractive index of GaAs are investigated as case studies.

In this investigation, first, the proposed method was applied in a single training phase as a single experiment. Then, the trained ANN was used for a testing phase, achieving an accuracy of 95.24%.

Then, to obtain more reliable results, an analysis of several experiments, i.e., several independent training runs, was conducted. For this analysis, ANNs with one hidden layer but various numbers of hidden neurons (from 1 to 60 hidden neurons) were deployed. For each architecture, 150 training runs were performed. Therefore, for this analysis, a total of $60 \times 150 = 9000$ experiments were conducted. Based on the results, for all ANNs except those with 1 and 3 hidden neurons, an accuracy of 100% was achieved in at least one experiment. Also, for the ANNs with 1 and 3 hidden neurons, the maximum accuracy achieved during the experiments was more than 95%. Therefore, the average maximum accuracy achieved by ANNs with different sizes was more than 99.8%.

In addition, as another scenario, the effect of normalization on the dataset has been examined. Two normalization techniques for the refractive index dataset have been used, i.e., the min-max and z-score methods. To provide a more efficient comparison, ANNs with different numbers of hidden neurons from 1 to 20 neurons have been implemented. In addition, for each ANN, 100 experiments have been conducted, i.e., 100 independent training and testing phases. Considering the two mentioned normalization techniques, in this scenario, a total of $2 \times 20 \times 100 = 4000$ experiments have been conducted. Based on the results, both normalization techniques achieved very similar average results, i.e., only a 0.72% difference in the average accuracy. Here, the z-score achieved slightly better results than the min-max technique.

To compare different machine learning models, various methods have been deployed within the proposed cyber-attack detection framework. In total, 13 models, including decision trees, logistic regression, SVM models, and ANNs, were implemented. Here, three decision trees with different maximum numbers of splits were used. Also, a logistic regression model for the binary classification was used. For the case of SVMs, four models with different kernel functions, i.e., linear, quadratic, cubic, and Gaussian kernels, were implemented. In addition, five ANNs with different architectures were implemented. For the first, second, and third ANNs, one hidden layer with different different numbers of hidden neurons was considered. For the fourth and fifth ANNs, variations in the number of hidden layers were considered. In this scenario, three ANNs and one of the SVM models have showed high accuracies for both the validation and training dataset, and one of the ANNs was selected as the most reliable model due to its smallest difference between the results obtained for the validation and testing datasets.

The proposed method has three technical features that can make it a feasible solution for addressing the cybersecurity problem in quantum cyber-physical systems, as follows:

- The method relies only on data from the system, and it does not need to know the physical or mathematical relationships within the system to detect cyber-attacks in quantum cyber-physical systems. The only point that should be taken into account is access to a ground-truth dataset to train the AI model correctly.
- The method uses a lightweight AI model, i.e., a shallow ANN, which can result in low computational costs and complexity.
- The use of simple AI architectures and independence from the physical and mathematical information about the system can result in ease of use.

Here, numerous studies considering the architecture of ANNs, different machine learning models, and normalization techniques were conducted. Based on the obtained results, the introduced AI-based framework can be a reliable solution for addressing the cybersecurity challenge in quantum cyber-physical systems, i.e., the detection of cyber-attacks.

7. Conclusions

In this study, an AI-based approach has been introduced to detect the existence of cyber-attacks in quantum cyber-physical systems. Here, the type of cyber-attacks was an FDIA, and the system compromised by the cyber-attacks was a photonics-driven quantum information system. In other words, the existence of false data in the refractive index value, as a key parameter for studying the interaction of light with materials, was investigated. The proposed strategy implemented ANNs for binary classification to detect FDIAs targeting refractive index values. The method was examined using the refractive index values of a semiconductor, GaAs. Based on the obtained results, the proposed strategy can effectively detect the existence of cyber-attacks.

8. Future Works

For future work, investigations corresponding to several research directions are suggested as follows:

- The detection of cyber-attacks targeting other parameters is suggested, e.g., temperature.
- The use of AI to mitigate false data values in the system is suggested. An incorrect refractive index value may affect the system by leading to an incorrect interpenetration of information, and the analyzing of data may result to a solution to remove the false data or modify the interpenetration of information before its use, and as a result, increasing the cyber-resiliency of quantum cyber-physical systems against cyber-attacks.
- Studies of different cyber-attacks, including negative attacks, structured attacks, energy-dependent attacks, and attacks with amplitudes close to the measurement noise, can be conducted.
- The implementation of experimental setups using different materials, as well as additional measurements to obtain a larger dataset, can be considered one of the future directions.
- Modifying data splitting to reduce the risk of data leakage.
- Other physical or statistical methods, such as detection based on deviations from the reference refractive index curve and linear classifiers, can be investigated to compare their performance with AI-based methods to detect cyber-attacks in quantum cyber-physical systems.

Funding: This research received no external funding.

Data Availability Statement: The optical parameters of GaAs can be found in [38]. The generated dataset during this study is available from the corresponding author upon reasonable request.

Conflicts of Interest: The author declare no conflict of interest.

References

1. Al-Suod, M.; Victor, B.; Valerii, T.; Olha, C.; Galina, S.; Zannon, M. Forecasting Energy Consumption of a Mining Plant Using Artificial Neural Networks. *IEEE Access* **2025**, *13*, 63237–63247. [[CrossRef](#)]
2. Karima, N.N.; Ahmad, S.; Islam, A.; Huda, A.S.N.; Awal, L.J.; Halim, S.A.; Mokhlis, H.; Ali, M.S.; Mubarak, S. Enhancing Short-Term Load Forecasting Using Hyperparameter-Optimized Deep Learning Approaches. *Energies* **2026**, *19*, 705. [[CrossRef](#)]
3. Asghar, R.; Fulginei, F.R.; Quercio, M.; Mahrouch, A. Artificial Neural Networks for Photovoltaic Power Forecasting: A Review of Five Promising Models. *IEEE Access* **2024**, *12*, 90461–90485. [[CrossRef](#)]

4. Zhou, Y.; Liu, Z.; Cheng, Z.; Mi, H.; Qin, Z.; Cao, K. A Photovoltaic Power Prediction Method Based on Wavelet Convolutional Neural Networks and Improved Transformer. *Energies* **2026**, *19*, 2040. [\[CrossRef\]](#)
5. Gijón, A.; Pujana-Goitia, A.; Perea, E.; Molina-Solana, M.; Gómez-Romero, J. Prediction of wind turbines power with physics-informed neural networks and evidential uncertainty quantification. *Eng. Appl. Artif. Intell.* **2026**, *164*, 113331. [\[CrossRef\]](#)
6. Hussain, A.; Kumar, K.; Ravikumar, G. Bayesian-optimized bidirectional long-short-term memory network for wind power forecasting with uncertainty quantification. *Electr. Power Syst. Res.* **2026**, *251*, 112261. [\[CrossRef\]](#)
7. Omar, M.B.; Rozihan, M.R.B.; Wishnuwardana, R.J.; Jusoh, W.N.L.B.W.; Bingi, K.; Ibrahim, R. AI-Driven Forecasting of Wave Energy Potential in Terengganu, Malaysia. In Proceedings of the 2025 International Conference on Advances in Machine Intelligence, and Cybersecurity Technologies (AMICT), Kota Kinabalu, Malaysia, 20–21 November 2025; pp. 19–24. [\[CrossRef\]](#)
8. Dai, C.; Li, X.; Qin, Y.; Jiang, S.; Yan, T.; Luo, M.; Jin, X. Forecasting the Wave Energy Flux in the China Sea Using a Blend of Physics and Artificial Intelligence. *Renew. Energy* **2026**, *272*, 126084. [\[CrossRef\]](#)
9. Amor, S.B.; Ziel, F. Recurrent neural networks with linear structures for electricity price forecasting. *Renew. Sustain. Energy Rev.* **2026**, *231*, 116773. [\[CrossRef\]](#)
10. Jia, H.; Guo, Y.; Zhang, X.; Ma, Q.; Yang, Z.; Zheng, Y.; Zeng, D.; Liu, D. Forecasting electricity prices in the spot market utilizing wavelet packet decomposition integrated with a hybrid deep neural network. *Glob. Energy Interconnect.* **2025**, *8*, 874–890. [\[CrossRef\]](#)
11. Alsokhry, F. Sensorless estimation of irradiance and temperature for renewable energy applications: An experimental examination. *Comput. Electr. Eng.* **2025**, *124*, 110329. [\[CrossRef\]](#)
12. Ahmed, A.; Rehman, H.U.; Ahmad, I.; Hasan, A. Encoder Decoder architecture with Bi-LSTM for model free sensorless control of induction motor. *Results Eng.* **2026**, *30*, 109962. [\[CrossRef\]](#)
13. Naguib, M.; Kollmeyer, P.; Emadi, A. Application of Deep Neural Networks for Lithium-Ion Battery Surface Temperature Estimation Under Driving and Fast Charge Conditions. *IEEE Trans. Transp. Electr.* **2023**, *9*, 1153–1165. [\[CrossRef\]](#)
14. Palacios, R.A.; Mauricio Villanueva, J.M.; Morales-Villanueva, A.; Aranda, C.B.; Waters Torres, O.A.; García, J.E.V. Development of a Digital Twin Model for a Hydrogen Production Plant Based on Artificial Intelligence Techniques. *IEEE Access* **2026**, *14*, 46371–46391. [\[CrossRef\]](#)
15. Akhter, M.N.; Ali, S.; Latif, A.; Rizwan, M.; Shehzad, M.F.; Ali, Y.; Muneeb, S.M.; Kamyab, H. Digital twin and artificial intelligence-driven optimization for renewable-dominated energy systems: A system-level integration framework. *Energy Convers. Manag. X* **2026**, *327*, 102010.
16. Semeraro, S.; Vecchi, F.; Stasi, R.; Berardi, U. Physics-Informed Neural Networks for predicting indoor temperature and cooling demand in historic buildings. *J. Build. Eng.* **2025**, *115*, 114392. [\[CrossRef\]](#)
17. Habibi, M.R.; Guerrero, J.M.; Vasquez, J.C. Artificial intelligence for cybersecurity monitoring of cyber-physical power electronic converters: A DC/DC power converter case study. *Sci. Rep.* **2024**, *14*, 22072. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Habibi, M.R.; Baghaee, H.R.; Dragičević, T.; Blaabjerg, F. Detection of False Data Injection Cyber-Attacks in DC Microgrids Based on Recurrent Neural Networks. *IEEE J. Emerg. Sel. Top. Power Electron.* **2021**, *9*, 5294–5310. [\[CrossRef\]](#)
19. Habibi, M.R.; Baghaee, H.R.; Dragičević, T.; Blaabjerg, F. False Data Injection Cyber-Attacks Mitigation in Parallel DC/DC Converters Based on Artificial Neural Networks. *IEEE Trans. Circuits Syst. II Express Briefs* **2021**, *68*, 717–721. [\[CrossRef\]](#)
20. de Jesus, G.F.; da Silva, M.H.F.; Pires, O.M.; da Silva, L.C.; dos Santos Cruz, C.; da Silva, V.L. Exploring Quantum Neural Networks for Demand Forecasting. *Entropy* **2025**, *27*, 490. [\[CrossRef\]](#) [\[PubMed\]](#)
21. da Silva, M.H.F.; de Jesus, G.F.; Nascimento, C.M.S.; da Silva, V.L.; Cruz, C.S. Exploring Quantum Machine Learning for Weather Forecasting. *Braz. J. Phys.* **2026**, *56*, 22. [\[CrossRef\]](#)
22. Peksa, J.; Perekrest, A.; Vadurin, K.; Mamchur, D. A Quantum-Hybrid Framework for Urban Environmental Forecasting Integrating Advanced AI and Geospatial Simulation. *Sensors* **2025**, *25*, 7422. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Pires, O.M.; Nascimento, E.G.S.; Moret, M.A. A quantum neural network model for short term wind speed forecasting using weather data. *Energy AI* **2025**, *231*, 100588. [\[CrossRef\]](#)
24. Zhen, L.; Bărbulescu, A. Quantum Neural Networks Approach for Water Discharge Forecast. *Appl. Sci.* **2025**, *15*, 4119. [\[CrossRef\]](#)
25. Cao, Y.; Zhou, X.; Fei, X.; Zhao, H.; Liu, W.; Zhao, J. Linear-layer-enhanced quantum long short-term memory for carbon price forecasting. *Quantum Mach. Intell.* **2023**, *5*, 26. [\[CrossRef\]](#)
26. Wang, Z.; Tang, H. Artificial intelligence for quantum error correction: A comprehensive review. *arXiv* **2024**, arXiv:2412.20380.
27. Gardeazabal-Gutierrez, J.; Terres-Escudero, E.B.; Bringas, P.G. Machine learning methods as robust quantum noise estimators. In *International Conference on Hybrid Artificial Intelligence Systems*; Springer Nature: Cham, Switzerland, 2024; pp. 238–247.
28. Jeong, Y.W.; Park, J.B.; Jang, S.H.; Lee, K.Y. A new quantum-inspired binary PSO: Application to unit commitment problems for power systems. *IEEE Trans. Power Syst.* **2010**, *25*, 1486–1495. [\[CrossRef\]](#)
29. Meng, K.; Wang, H.G.; Dong, Z.; Wong, K.P. Quantum-inspired particle swarm optimization for valve-point economic load dispatch. *IEEE Trans. Power Syst.* **2009**, *25*, 215–222.

30. Tu, S.; Rehman, O.U.; Rehman, S.U.; Ullah, S.; Waqas, M.; Zhu, R. A novel quantum inspired particle swarm optimization algorithm for electromagnetic applications. *IEEE Access* **2020**, *8*, 21909–21916. [[CrossRef](#)]
31. Rehman, Z.U.; Rehman, O.U.; Munshi, A.; Saeed, M.A.; Rehman, S.U. Parameters optimization of photovoltaic systems using modified quantum inspired particle swarm method. *Sci. Rep.* **2026**, *16*, 7864. [[CrossRef](#)] [[PubMed](#)]
32. Mahmoudi, Y.; Zioui, N.; Belbachir, H. An improved quantum-inspired particle swarm optimisation approach to reduce energy consumption in IoT networks. *Int. J. Cogn. Comput. Eng.* **2025**, *6*, 313–322. [[CrossRef](#)]
33. Khan, S.; Mazhar, T.; Shahzad, T.; Ali, T.; Ayaz, M.; Ghadi, Y.Y.; Aggoune, E.-H.M.; Hamam, H. Optimizing load demand forecasting in educational buildings using quantum-inspired particle swarm optimization (QPSO) with recurrent neural networks (RNNs): A seasonal approach. *Sci. Rep.* **2025**, *15*, 19349. [[CrossRef](#)] [[PubMed](#)]
34. Zouache, D.; Nouioua, F.; Moussaoui, A. Quantum-inspired firefly algorithm with particle swarm optimization for discrete optimization problems. *Soft Comput.* **2016**, *20*, 2781–2799. [[CrossRef](#)]
35. Gharehchopogh, F.S. Quantum-inspired metaheuristic algorithms: Comprehensive survey and classification. *Artif. Intell. Rev.* **2023**, *56*, 5479–5543. [[CrossRef](#)]
36. da Cruz, A.V.A.; Vellasco, M.M.; Pacheco, M.A.C. Quantum-inspired evolutionary algorithms applied to numerical optimization problems. In Proceedings of the IEEE Congress on Evolutionary Computation, Barcelona, Spain, 18–23 July 2010; pp. 1–6.
37. Chang, T.; Kim, J.U.; Kang, S.K.; Kim, H.; Kim, D.K.; Lee, Y.H.; Shin, J. Broadband giant-refractive-index material based on mesoscopic space-filling curves. *Nat. Commun.* **2016**, *7*, 12661. [[CrossRef](#)] [[PubMed](#)]
38. Aspnes, D.E.; Studna, A. Dielectric functions and optical parameters of si, ge, gap, gaas, gasb, inp, inas, and insb from 1.5 to 6.0 ev. *Phys. Rev. B* **1983**, *27*, 985. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.