Hindawi
IET Signal Processing
Volume 2024, Article ID 8884688, 13 pages
https://doi.org/10.1049/2024/8884688

Research Article

Enhancing Industrial Wireless Communication Security Using Deep Learning Architecture-Based Channel Frequency Response

Lamia Alhoraibi,[1] **Daniyal Alghazzawi**,[1] **Reemah Alhebshi**,[1] **Liqaa F. Nawaf**,[2] **and Fiona Carroll**[2]

[1]*Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia*
[2]*Cardiff School of Technologies, Cardiff Metropolitan University, Cardiff, UK*

Correspondence should be addressed to Lamia Alhoraibi; lsalehalhoraibi@stu.kau.edu.sa

Received 15 June 2023; Revised 12 December 2023; Accepted 7 March 2024; Published 28 March 2024

Academic Editor: Jung-Chieh Chen

Wireless communication plays a crucial role in the automation process in the industrial environment. However, the open nature of wireless communication renders industrial wireless sensor networks susceptible to malicious attacks that impersonate authorized nodes. The heterogeneity of the wireless transmission channel, coupled with hardware and software limitations, further complicates the issue of secure authentication. This form of communication urgently requires a lightweight authentication technique characterized by low complexity and high security, as inadequately secure communication could jeopardize the evolution of industrial devices. These requirements are met through the introduction of physical layer authentication. This article proposes novel deep learning (DL) models designed to enhance physical layer authentication by autonomously learning from the frequency domain without relying on expert features. Experimental results demonstrate the effectiveness of the proposed models, showcasing a significant enhancement in authentication accuracy. Furthermore, the study explores the efficacy of various DL architecture settings and traditional machine learning approaches through a comprehensive comparative analysis.

1. Introduction

Industrial wireless sensor networks (IWSN) have received increased attention in most industries in recent years thanks to their flexible nature and ability to work in challenging environments. IWSN can be used in various manufacturing applications, such as industrial control and process automation. However, the accessibility of IWSN raises concerns about security and privacy. Consequently, vulnerabilities may arise, primarily linked to the potential for adversaries to eavesdrop through the unauthorized interception of communications between industrial nodes, jam communications between nodes by flooding a channel with noise, or spoof communications by transmitting a forged signal between two authorized nodes. The wireless network must implement security procedures for access control and prevent unauthorized users. Every connection between nodes, whether involving a limited-access device or a smart device, is significant in terms of security. In the field of IWSN, addressing confidentiality, integrity, and availability is crucial as the foremost requirement for information security [1].

Authentication is an essential challenge in IWSN, where node identification requires safeguarding wireless communications to determine nodes' authenticity and grant them access while blocking unauthorized devices [2]. Wireless communication systems execute authentication using upper-layer authentication methods, which generally employ cryptographic algorithms [2]. Traditional authentication methods rely on vulnerable addresses, especially IP and MAC addresses. Nevertheless, upper-layer authentication methods that use traditional cryptography algorithms are insufficient for advanced wireless communication technologies [3], such as the Internet of Things (IoT) and the Internet of Vehicles. As artificial intelligence (AI) technologies progress, various cybersecurity techniques explore methods to leverage the capabilities offered by AI. Machine learning (ML) and deep learning (DL) enhancements have recently strengthened classification reliability within authentication systems.

Due to the open nature of wireless communication, adversaries that imitate authorized users can compromise IWSN. Therefore, the heterogeneous nature of the wireless transmission channel and the limited hardware and software capabilities of the nodes make secure authentication in an IWSN more challenging. Consequently, determining the source of unknown wireless transmitters and mitigating security dangers raised by adversaries requires a simple and lightweight authentication solution.

In this article, we present an uncomplicated authentication model. The model is formulated to enact lightweight authentication by leveraging the unique characteristics of the physical layer within the transmission medium and is integrated with DL approaches.

The main contributions of this article are summarized as follows:

(i) This article introduces the wireless physical layer authentication (WPLA) model designed for wireless industrial device communications. The model employs DL architectures incorporating autonomous parameter optimization, thereby serving as a substitution for traditional ML algorithms.

(ii) We have developed a representative sequential architecture consisting of two layers of convolutional neural networks (CNNs), two layers of long short-term memory (LSTM) architecture, and an architecture based on restricted Boltzmann machines (RBMs). This composite architecture is systematically designed to ascertain the optimal model for physical layer authentication.

(iii) The conducted experiments utilized the publicly available dataset provided by the National Institute of Standards and Technology (NIST) [4]. We propose a model that systematically explores channel impulse response (CIR) and transitions to channel frequency response (CFR) as a strategic approach to attaining reliable performance.

(iv) Furthermore, a comparative analysis has been undertaken to scrutinize the impact of varied DL structural configurations and traditional ML algorithms on classification performance.

The remainder of this article is organized as follows: In Section 2, an overview of the background of DL architectures. Related works are discussed in Section 3. In Section 4, the system model is briefly described. Then, in Section 5, a physical layer authentication model is proposed. Section 6 presents the result and discusses the proposed models' performance. Finally, the conclusion is presented in Section 7.

2. Background

DL computing has emerged as the predominant paradigm within ML, exhibiting remarkable efficacy in diverse, intricate cognitive investigations. Notably, DL has outperformed established ML methodologies across multiple domains, which can be attributed to its enhanced capabilities in data analysis. The categorization of DL approaches encompasses four principal classifications, as delineated below:

(i) Deep supervised learning: This approach engages with labeled data, where the classifier continuously refines network parameters to understand desired outputs better. Various classes of supervised learning, such as recurrent neural networks (RNNs) and CNNs [5], fall under this category. Additionally, RNN architectures encompass gated recurrent units and LSTM architectures [6].

(ii) Deep unsupervised learning: This approach facilitates learning without the availability of labeled data. The classifier autonomously discerns significant features to reveal latent structures or relationships within the input data. Recently developed frameworks within the DL family, such as RBMs [7], autoencoders, and generative adversarial networks, have demonstrated proficiency in addressing nonlinear dimensionality reduction and clustering problems.

(iii) Deep reinforcement learning: Combines reinforcement learning and DL, allowing the classifier to make decisions from unstructured input data without manual engineering. This technique was developed in 2013 with Google Deep Mind [8].

(iv) Deep semisupervised learning: This learning process relies on semi-labeled datasets to facilitate the learning process.

2.1. CNN. The CNN is typically designed for processing data with a known grid-like structure. The convolutional layer plays a crucial role in the CNN architecture, where each layer comprises multiple filters working collaboratively to conduct extensive convolutions on the input, resulting in the creation of feature maps. The learning algorithm trains these filters through a backpropagation mechanism, often represented as a multidimensional array of parameters [5]. To achieve an appropriate level of abstraction at the correct scale, filter sizes are selected based on the dimensions of the input data.

For instance, a convolutional layer convolves over the 2D input x using 2D filters k to extract features, represented as follows:

$$(x \times k)_{m,n} = x[m,n] \times k[m,n] = \sum_i \sum_j x[i,j] \cdot k[m-i][n-j]. \tag{1}$$

After the convolution, a bias term b and a point-wise nonlinearity f are utilized to create a feature map at the filter output. The feature map is created as follows using filters based on the input x and weights W:

$$k_{m,n} = f((x \times W)_{mn} + b). \tag{2}$$

A pooling layer is applied after a convolutional layer to perform a down-sampling operation, reducing the in-plane

dimensionality of the feature maps. This operation introduces translation invariance to minor shifts and distortions while also reducing the number of subsequent learnable parameters. The fully connected layer, constituting fully linked neurons with all feature maps from the last layer, represents a fundamental component of a CNN architecture.

2.2. LSTM. LSTMs are a distinct type of RNN designed to retain information in an internal state or memory, facilitating the creation of long-term dependencies. The two states—the cell state and the hidden state—are transmitted to the subsequent cell. LSTMs utilize gates to regulate the memorization process, preventing long-term dependence. The three gates comprising an LSTM are the input gate (I_t), the forget gate (F_t), and the output gate (O_t). Equations (3)–(9) delineate the fundamental structure of the LSTM model [6]. This learning process relies on semilabeled datasets to facilitate the learning process.

The information from the new input, X_t, is decided upon and stored in the cell state, C_t, by the first gate, the input gate. This phase comprises two components: the sigmoid and tanh functions. The sigmoid function determines whether to update the data based on the new information, while the tanh function weighs the values passed to assess their relative importance. Conversely, according to the sigmoid function, the forget gate is employed to determine unnecessary information that will be excluded from the cell. The last gate is an output gate, where a filtered version of the output cell state generates the output values, H_t. A sigmoid function selects the components of the cell state sent to the output, and the new values produced by the tan h from the cell state, C_t, are multiplied by the outcome of the sigmoid function.

Sigmoid function:

$$\sigma(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{1+e^x}. \tag{3}$$

Gate:

$$F_t = \sigma\left(W_f H_{t-1} + W_f X_t\right) + b_f, \tag{4}$$

$$I_t = \sigma(W_i H_{t-1} + W_i X_t) + b_i, \tag{5}$$

$$O_t = \sigma(W_o H_{t-1} + W_o X_t) + b_o. \tag{6}$$

Input transform:

$$N_t = \tanh(W_n H_{t-1} + W_n X_t) + b_n. \tag{7}$$

State update:

$$C_t = F_t \cdot C_{t-1} + I_t \cdot N_t, \tag{8}$$

$$H_t = \tanh(C_t) \cdot O_t. \tag{9}$$

Here, σ represents the sigmoid function, H_{t-1} is the output of the last LSTM unit at time $t-1$, X_t is the current input at time t, and W and b are the weight matrices and bias, respectively. Additionally, C_{t-1} and C_t denote the cell states at $t-1$ and t.

2.3. RBMs. RBMs are undirected probabilistic models with two layers: visible and hidden units [7]. To optimize the likelihood function, one must identify the joint probability distribution. RBMs develop neural network topologies for unsupervised data modeling using the concept of energy minimization [9]. The term "restricted" refers to the absence of intralayer connections between the visible and hidden units. However, connections between visible units and hidden units are still allowed.

There are two essential stages in the operation of RBMs: the feed-forward pass and the feed-backward pass. During the feed-forward pass, the inputs are multiplied by the weights, and the bias is applied. The result is then fed into a sigmoid activation function, with the function's output determining the activation of the hidden state. Conversely, the feed-backward pass reconstructs the functionality of the input units using the activated hidden units. The dual connection structure assumes conditional distributions of visible units on all hidden units, as well as distributions of hidden units on all visible units [5, 10], which are defined as follows:

$$p(v|h) = \prod_{i=1}^{m} p(v_i|h), p(h|v) = \prod_{j=1}^{n} p\left(h_j|v\right), \tag{10}$$

$$p(v_i|h) = \sigma\left(b_i + \sum_{j=1}^{n} h_j \cdot W_{ij}\right), \tag{11}$$

$$p\left(h_j|v\right) = \sigma\left(b_j + \sum_{i=1}^{m} v_i \cdot W_{ij}\right). \tag{12}$$

Originally, RBMs were proposed for use with binary visible units ($v \in \{0, 1\}$) and hidden units ($h \in \{0, 1\}$). The bias vector is denoted by the letter b, and the weight of the link between the ith visible unit and the jth hidden unit is represented as W_{ij}.

3. Related Work

This section provides an overview of the pertinent literature on the WPLA approach. WPLA is an authentication method designed to identify a wireless transmitter by analyzing physical layer characteristics within the transmission [11]. The focus of scholarly inquiry within WPLA has been directed toward methodologies such as radio frequency fingerprint (RFF) and channel-based techniques. The inception of RFF technology dates to 1995, when Toonstra and Kinsner [12] first proposed its conceptual framework. Rooted in exploiting manufacturing defects in wireless device components, RFF leverages minor imperfections inherent in the launch signal. Drawing a parallel to human biometric fingerprint identifiers, RFF is a practical and effective technique for authenticating wireless devices based on hardware irregularities.

On the other hand, due to the uniqueness, space-variability, time-variability, and reciprocity of the wireless channel, the physical layer features demonstrate the unique nature of the wireless channels used by the transmission parties. By analyzing channel features, for example, channel state information (CSI), CIR, CFR, and received signal strength. Signal classification and identification have become required with the advancement of wireless communication technologies. Described signal intelligence is a subject of research based on extracting signal features from unidentified radio frequency signals, which include modulation, center frequency, bandwidth, protocols, and transmitter identity [13]. Significant studies have implemented their classification methods by utilizing RFF [14] and channel features [15–18].

In recent years, DL has been increasingly integrated with WPLA within the context of secure wireless networks. This section introduces the pertinent literature exploring the intersections between DL and WPLA. Due to DL's excellent classification capabilities, deep neural networks perform exceptionally well when used for authentication. Liao et al. [16] created a multiuser authentication approach that could recognize numerous devices with little resource consumption using deep neural networks with data augmentation techniques. However, CNNs were chosen in some research because of their dependability and robust learning capabilities, which have high accuracy and low loss function during training. Baldini et al. [19] used CNN and recurrence plot techniques to develop classification approaches for the physical layer authentication challenge. To identify different devices by utilizing distinctive RF fingerprints, Aminuddin et al. [20] presented a methodology based on CNN to secure wireless transmission in a wireless local area network. Liao et al. [21] adopted deep neural networks, CNN, and convolution preprocessing neural networks to perform physical layer authentication in IWSN.

Furthermore, some research examined the relationship between the number of hidden layers and authentication rates, and it was discovered that the authentication rate improved as the number of hidden layers increased. In contrast, Ma et al. [22] used LSTM as an effective classifier to determine authorized and unauthorized users and increase detection efficiency and accuracy through simulations with varied channel conditions. Chatterjee et al. [23] demonstrated how to use the radiofrequency properties that the wireless device produces to authenticate devices in IoT networks as a trustworthy physical, unclonable function. Accordingly, a simple ML model was designed to consider receiver imperfections, channel, and data unpredictability.

4. System Model

We propose a WPLA model aimed at enhancing the security of IWSN with minimal impact on communication resources. Figure 1 illustrates numerous sensor nodes deployed across various locations within the industrial environment. We consider different wireless sensor nodes within the broadcast range of other nodes. For the sake of simplicity, we assume the availability of channel information for authorized nodes while the channel information for unauthorized nodes remains unknown. The

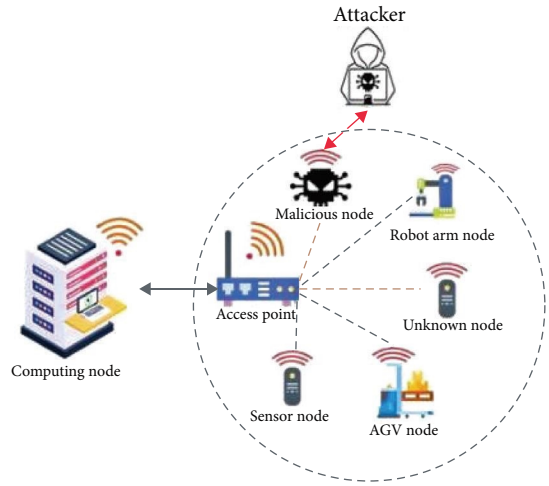

FIGURE 1: Illustration of our considered communication system model.

computational node is responsible for validating the legitimacy of received messages and ascertaining their origin from an authorized node based on channel information.

To initiate the authentication process, the computing node cyclically disseminates request messages to nodes within the industrial wireless network. Upon receipt of the message, all nodes transmit response messages to the computing node, incorporating both the pilot signal and identity information. Subsequently, upon receiving these response messages, the computing node conducts an initial assessment to ascertain the node type and detect potential identity conflicts by comparing the provided information with the stored identity details of authorized nodes. In the presence of conflicting declarations of identity, the node may be deemed unauthorized. Conversely, if the identity remains consistent, a node with an identical ID may be flagged as potentially malicious. Thus, the authentication process is briefly characterized as follows:

$$f(\mathrm{CFR}, \mathrm{ID}) = \begin{cases} \mathrm{ID}_i = \mathrm{ID}_j \text{ and } \mathrm{CFR}_i = \mathrm{CFR}_j, \text{Authorized node} \\ \mathrm{ID}_i = \mathrm{ID}_j \text{ and } \mathrm{CFR}_i \neq \mathrm{CFR}_j, \text{Malicious node} \\ \mathrm{ID}_i \neq \mathrm{ID}_j, \text{Unauthorized node} \end{cases}. \tag{13}$$

The detailed sequence diagram is depicted in Figure 2, where Node 1 is designated as authorized, Node 2 is defined as malicious, and Node 3 is unauthorized. The computing node is involved in identity authentication for the three nodes. When a valid node attempts to gain access, the computing node initially determines whether it is an authorized or malicious node. Subsequently, it undergoes physical layer channel authentication. Similarly, suppose a node requesting authentication lacks valid identity credentials. In that case, the computing node is initially categorized as an unauthorized node with incorrect identity information before proceeding with physical layer channel authentication.

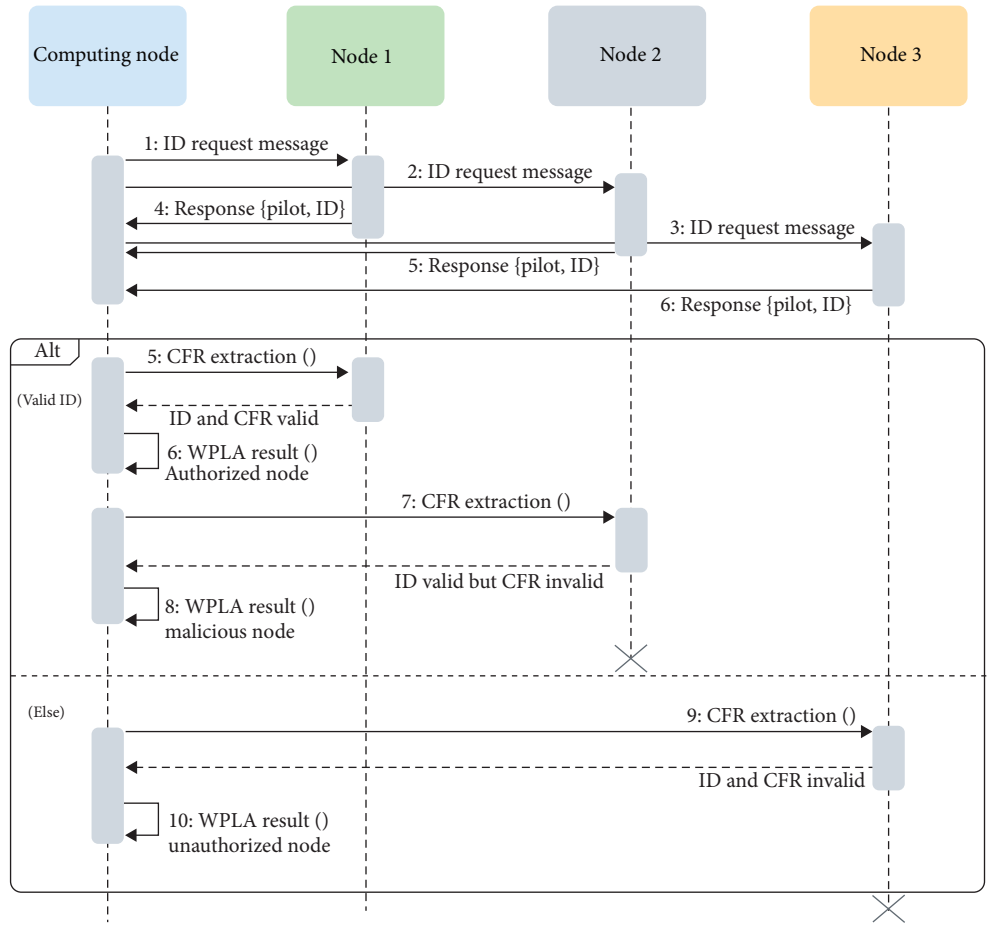

Figure 2: Sequence diagram of WPLA model.

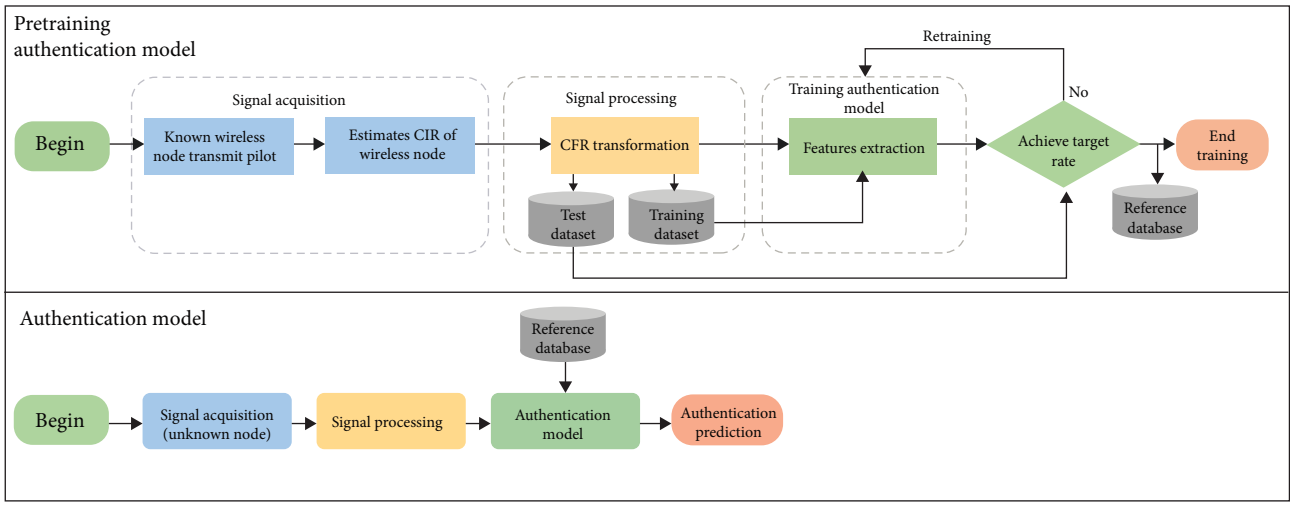

Figure 3: Proposed WPLA framework.

5. WPLA Framework

This section introduces a proposed WPLA framework for physical layer authentication based on different DL architectures, as illustrated in Figure 3. The procedural delineation of the proposed model comprises two distinct stages: the initial pretraining model stage and the subsequent authentication model stage. The initial phase of the pretraining model encompasses three consecutive stages. In the beginning, the signal acquisition part lets the authentication model get CIR data from different nodes that are spread out in an industrial setting. Subsequently, data processing transforms the acquired CIR

data into CFR values while eliminating irrelevant values. Lastly, the refined CFR data serves as input for the authentication module, facilitating the training and validation of a neural network model deployed on the authentication model. In the subsequent stage, the authenticator system, seeking to ascertain the identity of unknown nodes, transmits a set of CIR vectors to the authentication model. Upon receipt of these vectors, the authentication model extracts CFRs, undergoes a processing step, and inputs the processed CFRs into the trained authentication model. The authentication model then outputs an authentication decision.

5.1. Signal Acquisition. The CIRs constitute crucial information for the development of wireless communication networks. From a mathematical standpoint, the communication channel linking a transmitter and receiver is characterized by CIRs, serving as a comprehensive model that encapsulates the cumulative impact of various elements such as reflectors, absorbers, path loss, and environmental intricacies between them [24]. In particular, CIRs encompass all signal paths from the transmitter to the receiver, including those resulting from reflection, diffraction, or scattering [25]. Moreover, CIRs provide insights into both the propagation conditions and the positions of the receiver and transmitter within the given environment.

5.2. Signal Processing. In the following, the processing of CIRs is described in a simplified manner. Usually, CIRs are obtained by transmitting a pseudo-random sequence $s(t)$ known to both the transmitter and receiver. This property can be exploited to estimate the signal propagation channel. A convolution of $s(t)$ with the CIR that results in the received signal $y(t)$ is a straightforward model for the signal's propagation.

$$y(t) = s(t) * h(t) + n_y(t), \tag{14}$$

where $*$ denotes convolution. $n_y(t)$ represents the noise components, modeled as zero-mean white Gaussian noise.

We denote the continuous-time CIR of an L-path baseband wireless communication channel as follows:

$$h(t) = \sum_{i=1}^{L} h_i \delta(t - \tau_i), \tag{15}$$

where $\delta\ (t-\tau_i)$ is the Dirac delta function representing a delayed multipath replica of the transmitted signal arriving at time τ_i with power $|h_i|^2$. In particular, $h_i = a_i e^{j\theta i}$, where a_i and θ_i denote the amplitude and phase of the ith replica. We note that $h(t)$ fully describes the communication channel between the transmitter and receiver.

The complex received signal consists of the in-phase (I) and quadrature (Q), $I_i = a_i \cos(\theta_i)$, and $Q_i = a_i \sin(\theta_i)$, determine the in-phase and quadrature.

$$x(t) = xI(t) + xQj(t). \tag{16}$$

The signal samples $x(t) \in C$, $t = 0, \ldots, t-1$ are a time series of complex raw samples that are characterized as a data vector. This study considers simple data representation as a fast Fourier transform (FFT) converted the CIR into a discrete CFR. The FFT is a mathematical operation that converts signals from the time domain to the frequency domain. In order to reduce the number of calculations required for the FFT, discrete Fourier transform is used [26].

$$X(i) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) \cdot e^{\frac{-j2\pi ni}{N}}. \tag{17}$$

Here, $X(i)$ characterizes the frequency content of the time samples $x(n)$ associated with the original signal $x(t)$.

The result of the transformation, the FFT vector, consists of two sets of values: one that carries the real component and another that holds the imaginary component. Next, we convert the FFT vector to amplitude A_i by performing:

$$A_i = \sqrt{Ii^2 + Qi^2}. \tag{18}$$

The last step in this phase is splitting the dataset into a training set and a test set, where each set is composed of the following:

$$\text{Dataset}\{X\text{set}, Y\text{set}\}. \tag{19}$$

The set of CFR vectors for each node is termed Xset, and the classification label for each node's CFR vector is termed Yset.

5.3. Training Authentication Model. This section explores the utilization of DL architectures in the authentication model to address the inherent high dimensionality of CIRs. The rationale behind this selection is grounded in the various advantages that DL architectures offer, particularly in addressing the challenges associated with high-dimensional pattern identification, as substantiated by previous research findings [27, 28].

5.3.1. CNN Model Description. In the following section, we outline the visible and hidden layers of the proposed CNN model structure, as depicted in Figure 4. We begin by loading the CFR input sample in the visible layer. Subsequently, we reshape the sample in both the training and test sets to a fixed size of (1, 1, 8,188). The hidden layers consist of two convolution layers. The first convolutional layer comprises 256 neurons, followed by a Dropout layer. The second layer is a convolutional layer composed of 128 neurons, followed by a Dropout layer. Following a flattening layer, the combination of dropout regularization and the max norm has demonstrated excellent performance in preventing overfitting. The penultimate layers in CNN structure are the dense layers, which include neurons fully connected with all feature maps in the convolution layers. The first dense layer contains 64 neurons, and ReLU activation functions are applied to accelerate convergence during the training process. Finally, the last dense layer utilizes softmax activation to perform node classification.

5.3.2. LSTM Model Description. The proposed LSTM model with different layers is demonstrated in Figure 5. First, the CFR of the signal is fed to all neurons of the LSTM model for

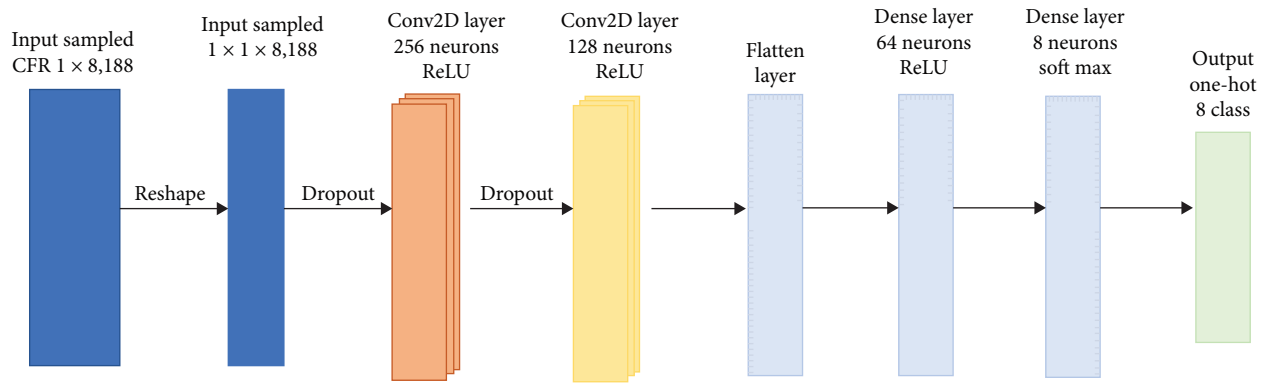

Figure 4: The proposed CNN structure.

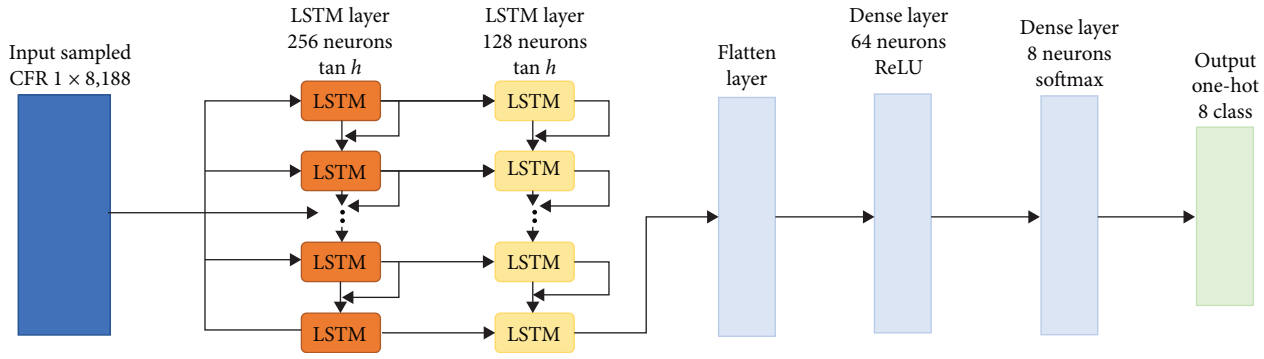

Figure 5: The proposed LSTM structure.

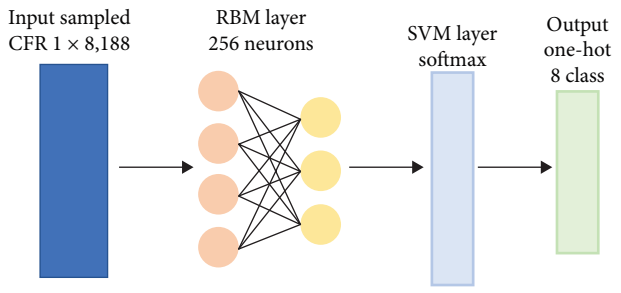

Figure 6: The proposed RBMs structure.

classification. The first layer contains 256 neurons, whereas the second has 128 neurons, and tanh activation functions are used in both layers. The flattened layer receives the 128-dimensional vector, the final output of the second LSTM layer. The penultimate layers in the LSTM structure are the dense layers, consisting of neurons fully connected with all feature maps in the LSTM layers. Then, the first dense layer contains 64 neurons, and ReLU activation functions are used. A dense softmax layer is the final layer that places the nodes' categorized features into one of eight output classes.

5.3.3. RBMs Model Description. The proposed RBMs model structure is a composite of one layer of an RBMs structure and a support vector machine (SVM) layer, as illustrated in Figure 6. RBMs represent a form of probabilistic modeling based on unsupervised nonlinear feature learning. Standard RBMs are binary; the input and output only have "0" and "1" states, where "0" indicates that the unit is inactive and "1" means that the unit is active [29]. In the proposed model, we utilized Bernoulli RBMs, where all units are binary stochastic. This requires that the input data be either binary or real-valued, falling between 0 and 1 [30]. Consequently, we normalized the dataset before training the model. After the feature extraction in the RBMs layer, the obtained features are sent to the softmax layer. Typically, good results come from feeding an RBM or a hierarchy of RBMs' features into a linear classifier like a linear SVM or a perceptron [30]. Hence, we incorporated an SVM classifier in the last layer to optimize authentication performance.

6. Evaluation Results and Discussion

This section delineates the implementation details of the models. Subsequently, it provides an exposition on the performance evaluation of the proposed models for authentication. Furthermore, an examination and comparative analysis of the authentication and convergence rates between the proposed DL models and traditional ML algorithms are conducted. Additionally, training and validation loss profiles are assessed to identify an optimal structure configuration. Finally, strategies to mitigate the issue of model overfitting are discussed.

6.1. Dataset Description. To obtain a CIR dataset within an industrial environment, authentic datasets sourced from the NIST [4] were utilized. The CIRs were acquired within the

Table 1: NIST dataset parameters configuration of the channel measurement system.

Parameter	Parameter setting
The center frequency GHz	5.4
Receiver/transmitter antenna polarization	Omni-directional, V Pol
The receiving antenna gain	−3.5
The transmitting antenna gain	3.6
The transmitting power	1.25 W
The sampling rate MHz	80
The channel impulse response in the time domain	8,188 × 1 complex vector

confines of a standard industrial site, specifically a machine shop. The machine shop, an indoor environment, possesses outer dimensions measuring approximately 12 by 15 m, with a ceiling height of roughly 7.6 m. The distance between the transmitter and receiver was at most 50 m. Channel-sounding tests were conducted using two portable devices, a transmitter and a receiver, strategically positioned within the factory. The capture of the CIR transpired as the receiving equipment moved from one acquisition point to another during the CSI measurements, resulting in each record denoting a distinct position. Consequently, the maximum distance between successive acquisitions is restricted to 1 m. The specific parameter configuration of the utilized dataset is detailed in Table 1.

6.2. Implementation Details. The proposed models have been implemented using Python, with Keras and TensorFlow employed for the CNN and LSTM models. Additionally, Sklearn was utilized for implementing the RBMs, SVM, and KMeans models. The training of these models was conducted on a workstation equipped with an NVIDIA graphics card and a high-performance CPU, such as the Intel Core i7, paired with 16 GB of RAM.

The dataset used for our evaluation analysis comprises 10,000 samples systematically partitioned into two distinct subsets: the training set, which consists of 8,000 samples (80%), and the test set, encompassing 2,000 samples (20%). Within this dataset, there are eight nodes, with three nodes designated as authorized. Additionally, the dataset includes five unauthorized nodes, which are classified as three malicious nodes and two identified as unknown. The training set was further divided into training and validation sets at a ratio of 7 : 1. The model training process was executed with parameters as outlined in Table 2, providing a comprehensive overview of the diverse hyperparameters employed in configuring the proposed models.

6.3. Evaluation Metrics. Evaluating the WPLA model's performance involved assessing several key performance indicators, including accuracy, precision, recall, and F1-score. The subsequent section outlines the definitions and formulas associated with these metrics.

The metrics are characterized by true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Accuracy measures the classifier's ratio of correctly predicting a node class. Precision represents the ratio of all positively predicted classes that are both positive and correct. Recall, or sensitivity, gauges the proportion of the model's predicted positive and correct classes relative to the total number of actual positive classes. The F1 score offers a balanced assessment by amalgamating precision and recall into a unified metric.

Table 2: Deep model configuration.

Hyperparameters	DL models
Loss function	Categorical crossentropy
Optimizer	Adam
Learning rate	0.001
Epochs	100
Batch size	512
Dropout rate	0.5

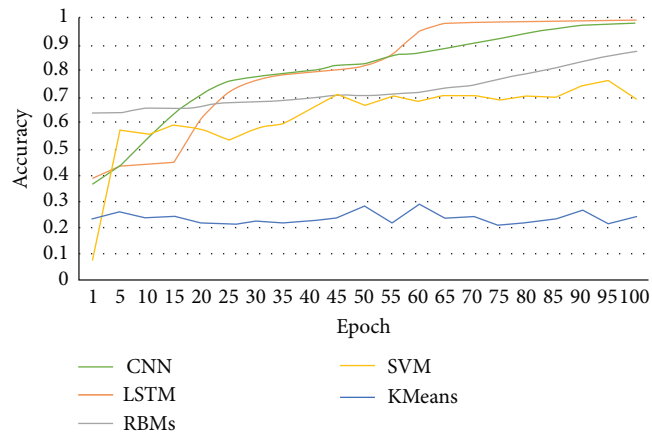

Figure 7: Accuracy rates comparison of different models.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}}, \tag{20}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{21}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{22}$$

$$\text{F1score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{23}$$

6.4. Comparative Analysis. A comparative analysis of the WPLA model used both DL architectures and traditional ML algorithms. This was undertaken to enhance the accuracy of evaluating the proposed authentication methodology. Each model's performance was assessed within a consistent framework, with uniform data dimensions set at 8,188. Figure 7 and Table 3 show that the DL models outperform their traditional ML counterparts. Specifically, SVMs and KMeans, representative of traditional ML models, exhibited comparatively inferior authentication performance.

In instances where DL architectures result in a 40% enhancement in performance, it is noteworthy that the model

Table 3: Result comparison for different DL and ML models.

Models	Accuracy	Precision	Recall	F1
CNN	0.985	0.9851	0.9843	0.9847
LSTM	0.9954	0.9954	0.9954	0.9954
RBMs	0.8752	0.9095	0.8896	0.8995
SVM	0.6947	0.8014	0.7936	0.7975
KMeans	0.2425	0.1215	0.1939	0.1494

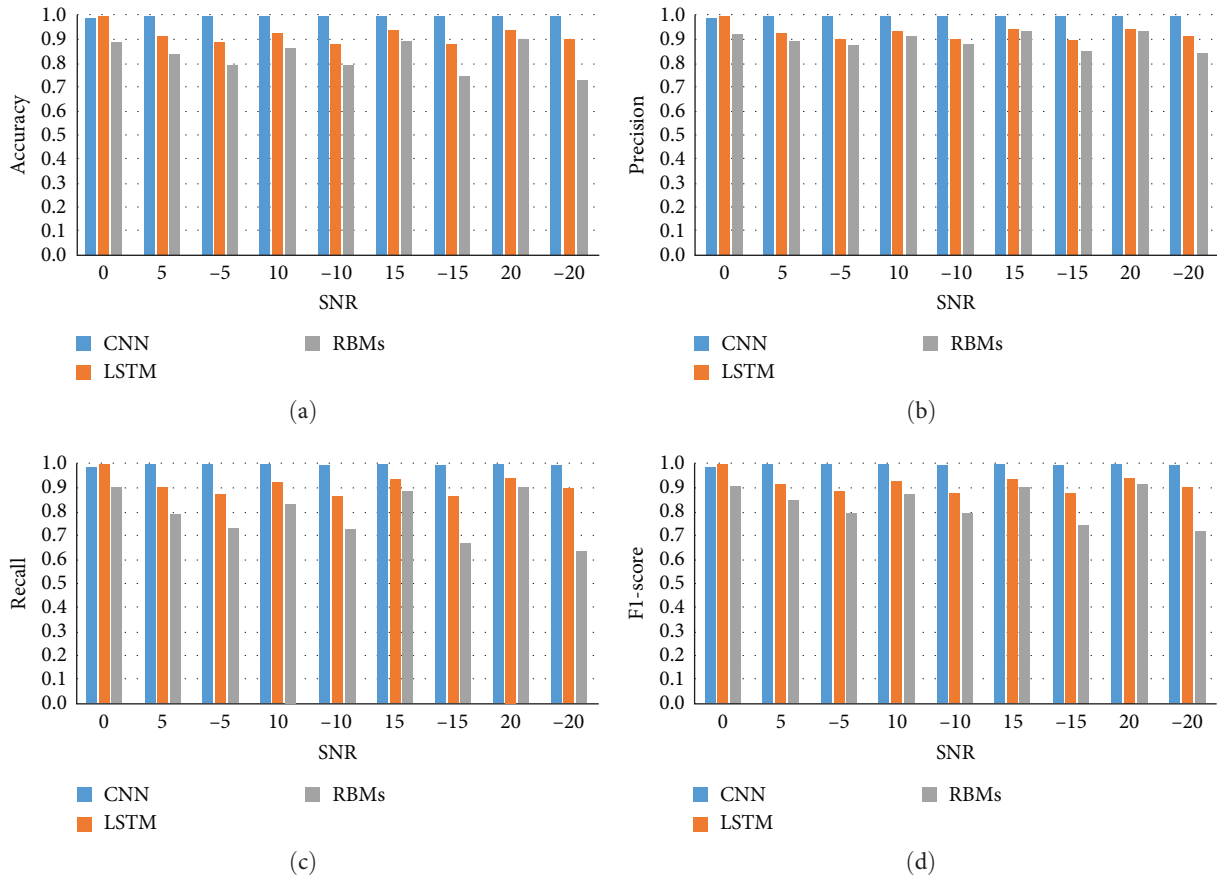

Figure 8: Result comparison under different SNRs: (a) accuracy, (b) precision, (c) recall, and (d) F1-score.

based on LSTM exhibits superiority over other models. The authentication accuracy rate attained a commendable 99% within the iteration range of 1–100. Another model proposed in this study, CNN, demonstrated comparable performance to the LSTM model, achieving an authentication accuracy rate of 98%. Conversely, the RBMs model achieved authentication rates in the range of 87%. In contrast, traditional ML algorithms SVM and KMeans demonstrated 69% and 24% authentication accuracy rates, respectively. The unsupervised KMeans model consistently produced 30% and 20% accuracy rates. In this comparison study, it is important to note that all models were given the same amplitude-form training and test sets, except for the RBMs and SVM models, whose inputs were normalized using L2.

Gaussian white noise is deliberately introduced into the extracted CFR data to assess the robustness of the proposed authentication models. This simulation aims to evaluate the model's performance under varying signal-to-noise ratio (SNR) conditions, and the associated authentication evaluation results are presented in Figure 8 and Table 4. High scores across a variety of metrics, consistently ranging between 98% and 99%, are evidence that the CNN model's performance exhibits resilience to SNR variations.

In contrast, the LSTM model's performance shows sensitivity to SNR changes. This sensitivity becomes apparent compared to the CNN, particularly under varying SNR conditions. Notably, the LSTM model demonstrates superior performance at 0 dB compared to CNN; however, this advantage decreases with changes in SNR. Specifically, at −15 dB, the LSTM model shows an accuracy of 87%, precision of 89%, and recall of 86%. Similarly, the RBM model shows sensitivity to changes in SNR. Positive and incremental effects on all

Table 4: Result comparison under different SNRs.

Model	SNR (dB)	Accuracy	Recall	Precision	F1-score
CNN	0	0.9850	0.9843	0.9851	0.9847
	5	0.9978	0.9978	0.9979	0.9978
	−5	0.9964	0.9962	0.9965	0.9964
	10	0.9971	0.9971	0.9971	0.9971
	−10	0.9926	0.9926	0.9928	0.9927
	15	0.9981	0.9979	0.9981	0.9980
	−15	0.9922	0.9919	0.9929	0.9924
	20	0.9969	0.9969	0.9971	0.9970
	−20	0.9931	0.9921	0.9932	0.9926
LSTM	0	0.9954	0.9954	0.9954	0.9954
	5	0.9101	0.9011	0.9191	0.9100
	−5	0.8867	0.8683	0.9009	0.8843
	10	0.9258	0.9208	0.9325	0.9266
	−10	0.8786	0.8599	0.8962	0.8777
	15	0.9358	0.9315	0.9399	0.9357
	−15	0.8774	0.8628	0.8946	0.8784
	20	0.9397	0.9364	0.9436	0.9400
	−20	0.9014	0.8924	0.9129	0.9025
RBMs	0	0.8841	0.8976	0.9151	0.9063
	5	0.8354	0.7891	0.8947	0.8386
	−5	0.7904	0.7277	0.8749	0.7945
	10	0.8574	0.8280	0.9100	0.8671
	−10	0.7904	0.7195	0.8788	0.7912
	15	0.8936	0.8803	0.9301	0.9045
	−15	0.7421	0.6618	0.8445	0.7421
	20	0.9016	0.9022	0.9334	0.9175
	−20	0.7250	0.6311	0.8407	0.7210

metrics are observed at SNR levels 5, 10, 15, and 20 dB, culminating in an F1-score of 92% at 20 dB. Conversely, deleterious effects on all metrics are noted at SNR levels of −5, −10, −15, and −20 dB, resulting in an F1-score of 72% at −20 dB.

6.5. Models Assessment. The evaluation of the performance of the proposed models is conducted by utilizing a loss function. This function serves to quantify the errors induced by the model precisely. A diminished loss value indicates a reduction in errors within the model's predictions, whereas an elevated loss value signifies increased errors. The assessment of how well a DL model aligns with the training and validation sets is facilitated by examining metrics termed training and validation losses. These losses are derived by aggregating the errors for each sample within the respective training and validation sets. Furthermore, the training and validation losses are measured after each epoch. This iterative measurement process aids in ascertaining whether adjustments to the model are warranted.

Within the models of CNN, LSTM networks, and RBMs, the loss function employed was categorical cross-entropy, expressed as follows:

$$L = -\sum_{i=1}^{N} t_i \cdot \log(p_i), \tag{24}$$

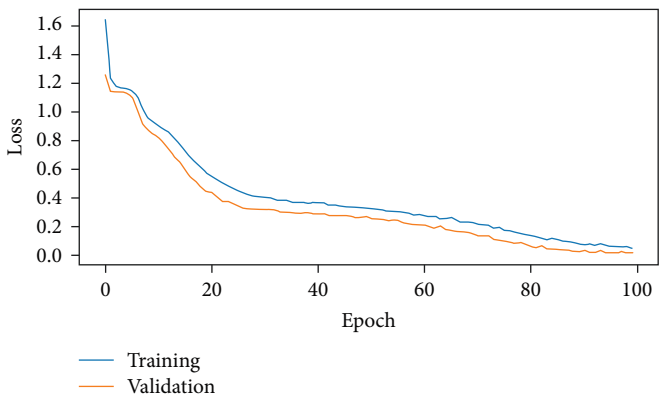

Figure 9: The cross-entropy loss over epochs for the CNN model.

where p_i refers to the softmax probability, t_i signifies the ground truth in the form of one-hot encoding, and the training batch size is indicated by N.

The evaluation of the discrepancy between the actual and predicted probability distributions for each class in the given problem is quantified through cross-entropy as a scoring metric. The attainment of a minimized score corresponds to an optimal cross-entropy value of 0. The evolution of

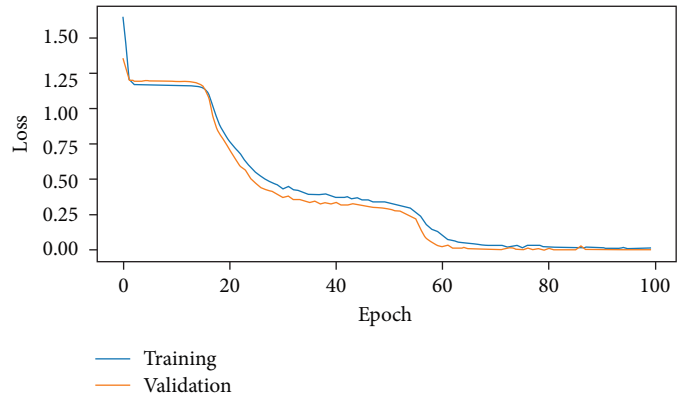

Figure 10: The cross-entropy loss over epochs for the LSTM model.

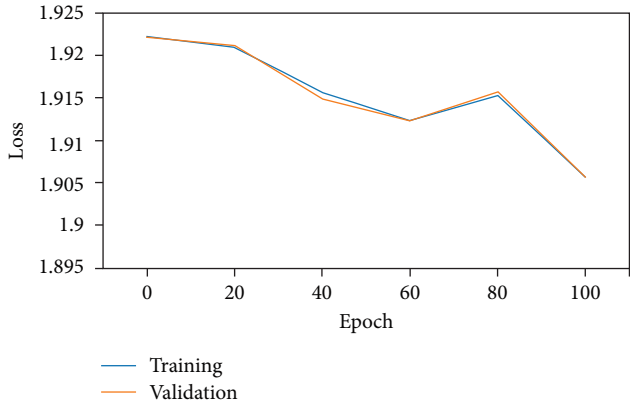

Figure 11: The cross-entropy loss over epochs for the RBMs model.

Table 5: Training information and accuracy comparison between DL models.

Models	CNN	LSTM	RBMs
Number of trainable parameters	2,138,056	8,853,576	2,096,384
Training time per epoch (s)	3	5	6
Training time (s)	300	400	625
Prediction time (μs/sample)	4	1	9
Prediction rate	0.9960	1	0.8665
Prediction loss	0.0188	0.00005	1.9056

cross-entropy loss over epochs for both the training and validation datasets is represented in Figures 9–11. Regarding the CNN and LSTM models, the depicted figures illustrate a consistent reduction in training and validation losses, suggesting an optimal fit. This observation implies a well-configured model, as there is an absence of discernible signs of overfitting or underfitting. Conversely, in the case of the RBMS model, the diminishment in both training and validation losses is comparatively less noticeable.

Table 5 displays that the prediction rate for both the CNN and LSTM models approaches 100% when the parameters are suitably selected. Notably, the CNN model exhibits the fastest training time compared to the other models. Conversely, the LSTM model demonstrates a notably reduced average training and prediction time, only 5 s per epoch and 1 μs per sample.

7. Conclusion

In the realm of IWSN, we present a WPLA model. This model autonomously learns from the frequency domain to improve identification performance and efficiency. Utilizing intelligent classifiers, both DL architectures and traditional ML algorithms are employed for physical layer authentication. The findings indicate that the proposed models show excellent performance, resulting in significantly improved authentication accuracy. High scores across various evaluation metrics indicate that the CNN model demonstrated exceptional performance, displaying resilience to SNR variations.

Finally, DL architectures provide practical solutions to training time and performance challenges, thereby offering a significant advantage in enhancing security systems. Despite these advancements, there remains considerable potential for further improvement in wireless communication security systems through DL architectures. Our future research initiatives will focus on establishing a generative adversarial network model to evaluate the capabilities of the proposed authentication model.

Abbreviations

CSI: Channel state information
CIR: Channel impulse response
CFR: Channel frequency response
DL: Deep learning
CNN: Convolutional neural network
LSTM: Long short-term memory
RBMs: Restricted Boltzmann machines
NIST: National institute of standards and technology
FFT: Fast Fourier transform.

Data Availability

A link to the database used in this work can be found at: https://www.nist.gov/ctl/smart-connected-systems-division/networked-control-systems-group/measurement-data-files (accessed on 3 September 2022).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Conceptualization has done by Lamia Alhoraibi, Daniyal Alghazzawi, and Reemah Alhebshi; methodology has done by Lamia Alhoraibi, Daniyal Alghazzawi, and Reemah Alhebshi; software-related task has done by Lamia Alhoraibi; validation has made by Lamia Alhoraibi, Daniyal Alghazzawi, Reemah Alhebshi, and Liqaa F. Nawaf; formal analysis has done by Lamia Alhoraibi, Daniyal Alghazzawi, and Liqaa F. Nawaf; writing—original draft preparation has made by Lamia Alhoraibi; review and editing has done by Fiona Carroll and Liqaa F. Nawaf; supervision has done by Daniyal Alghazzawi and Reemah Alhebshi.

Acknowledgments

The authors extend their sincere appreciation to the Deanship of Scientific Research at King Abdulaziz University (KAU), Jeddah, Saudi Arabia, for funding this project under grant no. RG-91-611-42.

References

[1] O. I. Uzougbo, S. S.-M. Ajibade, and F. Taiwo, "An overview of wireless sensor network security attacks: mode of operation, severity and mitigation techniques," 2020.

[2] L. Peng, A. Hu, J. Zhang, Y. Jiang, J. Yu, and Y. Yan, "Design of a hybrid RF fingerprint extraction and device classification scheme," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 349–360, 2019.

[3] N. Xie, Z. Li, and H. Tan, "A survey of physical-layer authentication in wireless communications," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 282–310, 2021.

[4] R. Candell, K. A. Remley, and N. Moayeri, *Radio Frequency Measurements for Selected Manufacturing and Industrial Environments*, NIST, Gaithersburg, MD, USA, 2016.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[7] R. Salakhutdinov and G. E. Hinton, "Deep Boltzmann machines," in *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, vol. 5, pp. 448–455, PMLR, 2009.

[8] V. Mnih, K. Kavukcuoglu, D. Silver et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, 2015.

[9] C. C. Aggarwal, *Neural Networks and Deep Learning*, Springer, 2018.

[10] V. Upadhya and P. S. Sastry, "Learning Gaussian–Bernoulli RBMs using difference of convex functions optimization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 10, pp. 5728–5738, 2022.

[11] Q. Wang, H. Li, D. Zhao, Z. Chen, S. Ye, and J. Cai, "Deep neural networks for CSI-based authentication," *IEEE Access*, vol. 7, pp. 123026–123034, 2019.

[12] J. Toonstra and W. Kinsner, "Transient analysis and genetic algorithms for classification," in *IEEE WESCANEX 95. Communications, Power, and Computing. Conference Proceedings*, vol. 2, pp. 432–437, IEEE, Winnipeg, MB, Canada, May 1995.

[13] A. Jagannath, J. Jagannath, and P. S. P. V. Kumar, "A comprehensive survey on radio frequency (RF) fingerprinting: traditional approaches, deep learning, and open challenges," *Computer Networks*, vol. 219, Article ID 109455, 2022.

[14] G. Shen, J. Zhang, A. Marshall, L. Peng, and X. Wang, "Radio frequency fingerprint identification for LoRa using deep learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2604–2616, 2021.

[15] F. Pan, Z. Pang, H. Wen et al., "Threshold-free physical layer authentication based on machine learning for industrial wireless CPS," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 6481–6491, 2019.

[16] R.-F. Liao, H. Wen, S. Chen et al., "Multiuser physical layer authentication in internet of things with data augmentation," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 2077–2088, 2020.

[17] L. Senigagliesi, M. Baldi, and E. Gambi, "Physical layer authentication with cooperative wireless communications and machine learning," in *2021 IEEE Latin-American Conference on Communications (LATINCOM)*, pp. 1–6, IEEE, Santo Domingo, Dominican Republic, November 2021.

[18] X. Qiu, J. Dai, and M. Hayes, "A learning approach for physical layer authentication using adaptive neural network," *IEEE Access*, vol. 8, pp. 26139–26149, 2020.

[19] G. Baldini, R. Giuliani, and F. Dimc, "Physical layer authentication of internet of things wireless devices using convolutional neural networks and recurrence plots," *Internet Technology Letters*, vol. 2, no. 2, Article ID e81, 2019.

[20] N. S. Aminuddin, M. H. Habaebi, S. H. Yusoff, and M. R. Islam, "Securing wireless communication using RF fingerprinting," in *2021 8th International Conference on Computer and Communication Engineering (ICCCE)*, pp. 63–67, IEEE, Kuala Lumpur, Malaysia, June 2021.

[21] R.-F. Liao, H. Wen, J. Wu et al., "Deep-learning-based physical layer authentication for industrial wireless sensor networks," *Sensors*, vol. 19, no. 11, Article ID 2440, 2019.

[22] T. Ma, F. Hu, and M. Ma, "A LSTM-based channel fingerprinting method for intrusion detection," in *2021 IEEE 5th International Conference on Cryptography, Security and Privacy (CSP)*, pp. 113–116, IEEE, Zhuhai, China, January 2021.

[23] B. Chatterjee, D. Das, S. Maity, and S. Sen, "RF-PUF: enhancing IoT security through authentication of wireless nodes using *in-situ* machine learning," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 388–398, 2019.

[24] M. Fomichev, L. F. Abanto-leon, M. Stiegler, A. Molina, J. Link, and M. Hollick, "Next2You: robust copresence detection based on channel state information," *ACM Transactions on Internet of Things*, vol. 3, no. 2, pp. 1–31, 2022.

[25] S. Kram, M. Stahlke, T. Feigl, J. Seitz, and J. Thielecke, "UWB channel impulse responses for positioning in complex environments: a detailed feature analysis," *Sensors*, vol. 19, no. 24, Article ID 5547, 2019.

[26] A. Goldsmith, *Wireless Communications*, pp. 1–692, Cambridge University Press, Cambridge, UK, 2005.

[27] P. I. Wójcik and M. Kurdziel, "Training neural networks on high-dimensional data using random projection," *Pattern Analysis and Applications*, vol. 22, pp. 1221–1231, 2019.

[28] B. Liu, Y. Wei, Y. Zhang, and Q. Yang, "Deep neural networks for high dimension, low sample size data," in *IJCAI'17: Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 2287–2293, AAAI Press, Melbourne, Australia, August 2017.

[29] N. Zhao and Z. Liu, "Communication signal recognition technique based on fusion deep belief network," in *2019 IEEE 19th International Conference on Communication Technology (ICCT)*, pp. 184–187, IEEE, Xi'an, China, October 2019.

[30] P. Smolensky, "Information processing in dynamical systems: foundations of harmony theory," in *Parallel Distributed Processing*, vol. 1, pp. 194–281, MIT Press, Cambridge, 1986.