Signal Error Analysis Approach Implicated Supervised Neural Networks

Farhan Ali [1,2,*], He Yigang [3,*], Cheng Wei Ding [2], Atta-ur-Rahman [4,*], Aghiad Bakry [4], Dania Alkhulaifi [4], and Mohammad Ahmad Saleem Khasawneh [5]

[1] College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China
[2] School of Electrical Engineering and Automation, Hefei University of Technology, Hefei, China
[3] School of Electrical Engineering and Automation, Wuhan University, Wuhan, China
[4] Department of Computer Science, College of Computer Science and Information Technology, Imam Abdulrahman bin Faisal University, Dammam, Saudi Arabia
[5] Special Education Department, King Khalid University, Abha, Saudi Arabia
Email: farhan.ali9566@gmail.com (F.A.); 18655136887@163.com (H.Y.); chengweiding@mail.hfut.edu.cn (C.W.D.); aaurrahman@iau.edu.sa (A.R.); ababakri@iau.edu.sa (A.B.); 2220500211@iau.edu.sa (D.A.); mkhasawneh@kku.edu.sa (M.A.S.K.)
*Corresponding author

***Abstract*—This paper investigates signal quality enhancement by addressing critical challenges in signal error detection. The complex data generated by communication systems often introduce significant transmission errors, necessitating robust methodologies to ensure reliable communication. Advanced neural networks and machine learning techniques were employed to address these challenges, focusing on Artificial Neural Networks (ANNs) using backpropagation and feedforward Multi-Layer Perceptron (MLP) models. Deep neural networks and Support Vector Machines (SVMs) were also explored for comparative analysis. Simulations were conducted using artificially generated datasets of radio waves to train the models and evaluate their error-detection capabilities. The results highlighted the superior performance of the ANN model, which demonstrated higher accuracy and efficiency in optimizing signal transmission compared to the Deep neural network and SVM approaches. These findings delineate the effectiveness of ANN in mitigating transmission errors and achieving reliable communication in dynamic network environments. These findings hold significant implications for improving the reliability and efficiency of signals, contributing to advancements in wireless communication systems and their ability to handle the increasing demands of modern connectivity.**

***Keywords*—5G, signal error, supervised learning, Feed Forward Neural Network (FFNN)**

I. Introduction

5G Continuous research, standardization, and evaluation are conducted due to its powerful fundamental characteristics and essential features [1, 2]. Neural networks and Machine learning models are extensively used in the 5G research domain for predictions [3, 4]. This describes that neural network models such as General Regression Neural Network (GRNN), Radial Basis Function Neural Network (RBFNN), approximate RBFNN, and exact RBFNN are effectively performing accuracies and classifying interference occurrences in Networks [5–7]. AI/Machine Learning (ML) is also developed to accurately classify signals in 5G advanced networks using classifiers like XGBOOST and random forest, which accurately differentiate into Non-Line of Sight (NLOS) and Line of Sight (LOS) that achieve significant accuracy [8]. Recent research and developments describe that neural networks are highly effective in forecasting 5G signal strength. Multilayer perception achieved the highest accuracy in predicting Reference Signal Receiving Power (RSRP) [9]. Ultra-Dense Networks (UDN) utilized neural networks for training to forecast signal strength in radio environment maps with limited observation, which showed strong and consistent performance [10]. Deep neural networks notably enhance accuracy in the conventional method of analyzing wireless propagation characteristics [11]. These studies collectively address the ability of neural networks to accurately predict 5G signal strength in both current 5G and future beyond 5G networks [12].

Neural Networks (NNs) exhibit distinct advantages in signal error reduction compared to conventional architectures [13–15]. Their straightforward structure, where information flows unidirectionally from input to output [16], allows for effective training and error correction mechanisms. This simplicity facilitates the approximation of complex functions and enhances classification accuracy, particularly in applications like human action recognition and signal processing [17]. Signal error reduction in communication systems has been a research focus due to challenges posed by multipath fading, interference, and high mobility [18, 19]. These techniques are effective, such as Feed Forward Neural Networks (FFNNs) [20], Support Vector Machines (SVMs) [21]. Pilot-based channel estimation methods have been developed to enhance signal reliability and accuracy,

Manuscript received November 18, 2024; revised December 10, 2024; accepted December 31, 2024; published May 22, 2025.

doi: 10.12720/jait.16.5.760-769

which provide insights into channel conditions. At the same time, algorithms such as Minimum Mean Square Error (MMSE) offer improved Channel State Information (CSI) estimation. Despite these advances, practical implementation in dynamic environments remains a complex challenge. Existing techniques for signal error reduction face limitations in high-mobility scenarios, where accurate and efficient CSI estimation becomes increasingly challenging. Simplified models used in many studies fail to capture real-world environments' complexities adequately. Additionally, the integration of novel approaches, such as dual antenna systems and enhanced machine learning algorithms, has not been thoroughly explored. Current methods also encounter difficulties maintaining low latency and ultra-reliable communication, particularly under dynamic channel conditions, limiting their scalability and real-time applicability.

This research paper employs Artificial Neural Network (ANN), FFNN, Back Propagation Neural Network (BPNN) and machine learning techniques to improve communication reliability by reducing impairments and attaining optimal signal performance through real-time optimization and validation. It underscores the significance of parameter selection, network architecture, and activation functions in optimizing neural networks while mitigating biases and enhancing applications for reliable results.

II. LITERATURE REVIEW

ANNs have gained significant attention as an appropriate choice for simulating systems owing to their effectiveness in several technical and scientific domains, including signal processing, image processing, control systems, and associative memory [22, 23]. The development of ANNs is serving as a versatile, appropriate, and accurate modeling alternative to computational simulations [24]. Detection and prediction through neural networks offer significant benefits for addressing intricate computational challenges in several previously considered unattainable activities for classically coded systems [25–28]. ANN can be trained using a small amount of data to replicate complicated physical simulations accurately. A Multi-Layer Perceptron (MLP) is an FFNN that is completely linked. It effectively manages substantial volumes of input data and generates rapid predictions following training. A comparable level of accuracy can be attained even when working with lower sample sizes. MLP, or Multilayer Perceptron, is a neural network capable of solving complex nonlinear problems. MLP using the back-propagation training method is extensively employed in addressing challenges necessitating supervised learning and in scientific computing and parallel distributed processing, such as engineering applications. ANN has been used to simulate various systems and analyze their behavior [29–31]. The evaluation of machine learning models has been conducted utilizing these current data sources, yielding excellent and promising results in terms of real-time detection. ANNs have achieved significant performance in reducing error by supervised learning [32–34]. Table I describes various models that achieved performance in accurately supervised and unsupervised learning in 5G neural networks.

TABLE I. SUPERVISED AND UNSUPERVISED LEARNING MODELS IN 5G

Literature	Learning Techniques	Algorithm Used	Contribution to 5G
[35]	Supervised	Radial Basis Function, SupportVector Machine	Resource Allocation and Cloud Radio AccessNetwork (CRAN) in 5G user band in position
[36]	Supervised	Random Forests, Multi-Class Classifier	Reduction of signaling overhead in Cloud Radio Access Networks (CRANs) with dense radio heads Improved performance compared to conventional channel state information acquisitionmethod.
[37]	Supervised	Linear Regression	Wireless channel path-loss prediction for mm-wavefrequencies by reducing the required number of measurements and the channel capacity
[38]	Supervised	Linear Regressionand LogisticRegression	Prediction of user load using time-series analysisusing Recursive Neural Network (RNN) and Long Short-Term Memory (LSTM) models in Anomaly detection using Logistic Regression in 5G cellular network
[39]	Unsupervised	Hierarchicalclustering	Optimum number of cluster discoveries for mm-Wave Nonorthogonal Multiple Access (NOMA) systems
[40]	Unsupervised	Deep Neural Based Secure Framework	Detection and prevention of attack propagation
[41]	Unsupervised	Recurrent Neural Network	Beam sweeping pattern prediction during celldiscovery in 5G
[42]	Unsupervised	Independent Component Analysis	Reduced dispersion of polynomial-based model coefficients in reducing memory requirement
Proposed Algorithm	Supervised Learning	ANN (FFNN, BPNN)	Improved Signal Accuracy for 5G communication

The proposed approach in this paper deals with transmitted signals' sensitivity to distortions and impairments induced by the communication path [43], which is a significant challenge when using an optimized ANN model. These distortions may decrease the signal quality, resulting in errors and reducing the communication process's overall efficiency. Solving this issue is essential to guaranteeing high-fidelity and reliable data transfer, especially in current telecommunication systems, where signal integrity is critical. So, in this paper, ANN (FFNN and BPNN) is applied to reduce signal error until it approaches optimal signal performance.

III. Proposed Methodology

A. ANN Architecture

Fig. 1 describes the architecture of the FFNN used in this study. The input layer is the data entry point, and signals are forwarded to the hidden layer. Every hidden layer consists of several neurons, weight regularization, activation function sigmoid, and dropout. The output layer produces the output. The output layer produces results by taking data from the hidden, sigmoid, and classification layers. The data set is classified as 70% training, 15% testing, and 15% validation. Several hyperparameters are used as a learning rate of 0.01 to calculate output. The output layer is the classification layer in the employed architecture, which provides the final decision with the highest probability.

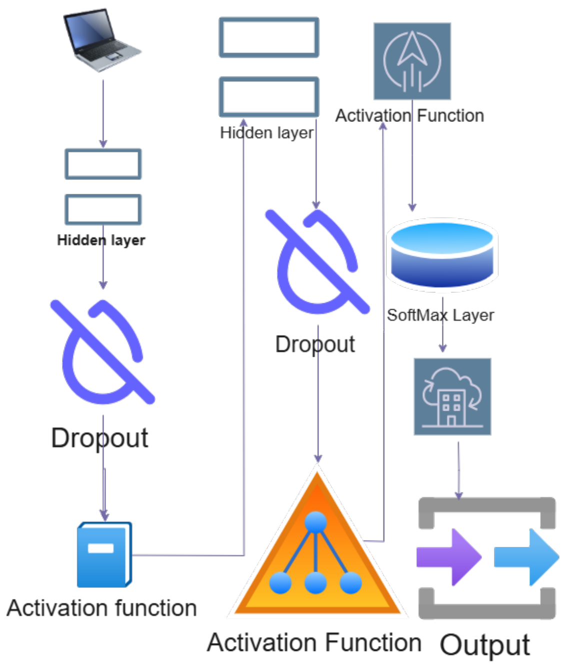

Fig. 1. FFNN architecture of the proposed approach.

B. Mathematical Model

The NN model offers a flexible and intelligent framework for adjusting to the many characteristics and scenarios inherent in 5G signal streaming simulations. Incorporating waveform types, channel models, and other simulation elements into the ANN model establishes it as a powerful tool for optimizing signal processing decisions in simulated complex and dynamic wireless network settings.

$$\mathfrak{b}_j = \Upsilon_1 + \sum_{i=1}^{m} (\omega_{ij} \times r_i) \tag{1}$$

$$\mathfrak{g}_j = \frac{1}{1+e^{-\mathfrak{b}_j}} \tag{2}$$

where $j = 1, 2, 3, \ldots, n$.

Eqs. (1) and (2) depict a FFNN model for optimizing signal processing in simulations of 5G signal streaming. Eq. (1) computes the amount of input given to a node, taking into account a reference value Υ_1 and a weighted combination of inputs from different nodes incorporates factors such as waveform type, guard band, bandwidth, and channel models. Eq. (2) utilizes a sigmoid activation function to compute the node output, introducing flexibility, adaptability, and non-linearity. This FFNN model is specifically developed to make dynamic decisions about signal processing tasks, such as selecting the appropriate modulation scheme and equalizing the channel, in response to the constantly changing and varied conditions simulated in 5G networks. The sigmoid function maps the net input to a range bounded by 0 and 1, indicating the activation likelihood. The activation function provides non-linearity to the ANN model, enabling it to capture intricate correlations in the input data effectively. The result is significant as it helps to make signal processing choices, including selecting modulation schemes, channel equalization, and overall network optimization. The equations encapsulate the fundamental nature of the ANNs capacity to analyze and acquire knowledge from the simulation parameters, facilitating optimum decision-making in signal processing for efficient and dependable communication inside the intricate 5G setting. Eq. (1) adjusts for the effect of fading by calculating the weighted sum of inputs, where each weight ω_{ij} and input r_i can be impacted by fading circumstances. The channel characteristics are influenced by fading phenomena, such as block fading, and FFNN is specifically built to adjust to these fluctuations throughout the learning process.

$$\mathfrak{b}_{\boldsymbol{k}} = \Upsilon_2 + \sum_{\boldsymbol{j=1}}^{\boldsymbol{n}} (\boldsymbol{v_{jk}} \times \mathfrak{g}_{\boldsymbol{j}}) \tag{3}$$

Eq. (3) is the sum of the inputs to the output layer in the FFNN model. It plays a vital role in the decision-making process of the FFNN, especially for tasks linked to the functions of the output layer. The output layer, as defined by this equation, is of utmost importance in determining the signal processing choices made in the 5G simulation, such as choosing the most suitable modulation scheme, equalizing the channels, or making other decisions that affect the effectiveness and dependability of signal transmission in the simulated wireless network. The value of $\mathfrak{b}_k$ in this equation is obtained by adding together the result of multiplying the weights v_{jk} with the outputs $\mathfrak{g}_j$ from all nodes in the previous layer. The value Υ_2 is included in this aggregated entirety after being multiplied by its respective weight. The weights v_{jk} undergo adaptation through advanced interactions with simulation parameters to ensure that the FFNN efficiently processes the signal for successful communication in the dynamic and complex simulated 5G environment. Eq. (3) incorporates a complex mechanism that utilizes acquired knowledge to make well-informed decisions at the output layer, improving the flexibility and intelligence of the signal processing in the 5G simulation [44].

$$\mathfrak{g}_k = \frac{1}{1+e^{-\mathfrak{b}_k}} \quad where\ k = 1,2,3, \ldots, r \tag{4}$$

Eq. (4) functions as the activation function for the output layer, playing a vital part in determining the final decisions for signal processing. It calculates the output g_k of the k_{th} node in the output layer by applying a sigmoid activation function to the net input h_k. The output g_k, which is determined by the sigmoid activation, plays a role in making decisions related to tasks like modulation scheme selection, channel equalization, and other important tasks for maximizing signal transmission in the simulated 5G network. The sigmoid function transmits the net input to a number within the range of 0 to 1, which indicates the likelihood of activation for the k_{th} node. This activation function is essential in determining the ultimate decisions related to signal processing. The output g_k which is determined by the sigmoid activation, is anticipated to play a role in choices related to modulation scheme selection, channel equalization, and other important tasks for maximizing signal transmission in the simulated 5G wireless network. The sigmoid activation function provides non-linearity to the model, enabling it to capture complex relationships and patterns in the simulated data. Therefore, FFNN becomes proficient in acquiring knowledge from the ever-changing wireless environment intelligence. FFNN significantly helps in improving signal processing, eventually boosting the efficiency and reliability of 5G communication in the simulated situation.

$$E = \frac{1}{2}\sum_k(\tau_k - g_k)^2 \tag{5}$$

The error function is crucial during FFNN training phase. The primary purpose is to iteratively modify the weights to reduce the signal error and improve the accuracy of predictions. The network adapts weights to minimize the discrepancy between predicted and target outputs. Eq. (5) describes the error function in the context of backpropagation. It is primarily used to measure the discrepancy between the anticipated signal g_k output and the target output τ_k for each node in the output layer. The variable k in this equation denotes the index of nodes in the output layer. The error function is a crucial element in the training process, enabling the optimization of the ANN to deal with the intricacies of the dynamic wireless environment effectively. The network's capacity to acquire knowledge and adjust to various simulation settings is demonstrated, enhancing the effectiveness of the 5G signal. This is achieved by modifying the network's weights using BPNN to minimize errors, guaranteeing that its predictions closely align with the desired target outputs.

$$\Delta W \propto -\frac{\partial E}{\partial W}$$

$$\Delta \upsilon_{j,k} = -\epsilon\frac{\partial E}{\partial \nu_{j,k}} \tag{6}$$

Eq. (6) delineates the rate of change in the output layer weights throughout the backpropagation process. Eq. (6) is essential for optimizing the network's parameters during the training phase to minimize the error E between the predicted g_k and target τ_k outputs for each node in the output layer. ΔW denotes the alteration in weight W for a particular link in the output layer. The negative gradient of the error concerning the weight $-\frac{\partial E}{\partial W}$, signifies the optimal direction for adjusting the weight to minimize the error. The factor of proportionality, denoted by ∝∝, indicates that the weight change is directly proportional to the negative gradient. On the other hand, the learning rate defines the magnitude of the weight modifications. The symbol $\Delta\upsilon_{j,k}$ denotes the difference in weight $\upsilon_{j,k}$ that connects the previous layer's j_{th} node to the output layer's k_{th} node. The symbol ϵ $(-\epsilon\frac{\partial E}{\partial \nu_{j,k}})$ represents the negative gradient of the error concerning this weight. The learning rate ϵ determines the size of the weight modifications. The weight adjustment procedure guarantees that the network continuously improves its internal representations by aligning the calculated outputs g_k with the desired outputs τ_k through iterative refinement. The network's capacity to provide precise predictions and enhance signal processing tasks in the simulated 5G environment is improved by the repeated modifications directed by the error gradients.

$$\Delta\upsilon_{j,k} = -\epsilon\frac{\partial E}{\partial g_k}\times\frac{\partial g_k}{\partial h_k}\times\frac{\partial h_k}{\partial \nu_{j,k}} \tag{7}$$

Eq. (7) improves the network's ability to predict outcomes accurately. $\frac{\partial E}{\partial g_k}$ calculates the sensitivity of the error variations in the predicted output. $\frac{\partial g_k}{\partial h_k}$ measures the variation in the predicted output concerning the desired target output τ_k . $\frac{\partial h_k}{\partial \nu_{j,k}}$ Illustrates the influence of modifications to the weight $\nu_{j,k}$ on the desired output. Eq. (7) calculates the rate of change in the weight $\nu_{j,k}$ during the training process, the negative sign indicates the direction of adjustment to minimize the error. The learning rate ϵ modulates the magnitude of the weight update, ensuring a controlled and effective learning process. By aligning the computed outputs g_k with the intended outputs τ_k, Eq. (7) enhances the predictive accuracy of the network, contributing to the optimization of signal-processing tasks critical for effective communication.

$$\Delta\upsilon_{j,k} = \epsilon(\tau_k - g_k)\times g_k(1-g_k)\times(g_j)$$
$$\Delta\upsilon_{j,k} = \epsilon\xi_k g_j \tag{8}$$

Updated weight $\Delta\upsilon_{j,k}$ can be obtained by substituting specific values into Eq. (8). It is expressed as $\Delta\upsilon_{j,k} = \epsilon(\tau_k - g_k)\times g_k(1-g_k)\times(g_j)$. Further simplifying, the expression can be written as $\Delta\upsilon_{j,k} = \epsilon\xi_k g_j$, where ξk = $(\tau_k - g_k)\cdot g_k\cdot(1-g_k)$ is a composite term that represents the sensitivity of the error to the output g_k. The learning rate (ϵ) determines the extent to which the weights are adjusted, ensuring controlled and efficient learning, while ξk represents the sensitivity to errors, which is influenced by the predicted output g_k. The iterative learning process, influenced by the subtle interaction of these components, improves the network's predictive accuracy and optimizes its capability to handle the complexities of the dynamic 5G environment.

$$\Delta\omega_{i,j} \propto -\left[\sum_k \frac{\partial E}{\partial g_k} \times \frac{\partial g_k}{\partial h_k} \times \frac{\partial h_k}{\partial g_j}\right] \times \frac{\partial g_j}{\partial h_j} \times \frac{\partial h_j}{\partial \omega_{i,j}}$$

$$\Delta\omega_{i,j} = -\epsilon \left[\sum_k \frac{\partial E}{\partial g_k} \times \frac{\partial g_k}{\partial h_k} \times \frac{\partial h_k}{\partial g_j}\right] \times \frac{\partial g_j}{\partial h_j} \times \frac{\partial h_j}{\partial \omega_{i,j}}$$

$$\Delta\omega_{i,j} = \epsilon \left[\sum_k (\tau_k - g_k) \times g_k(1-g_k) \times (\nu_{j,k})\right] \times g_k(1-g_k) \times r_i$$

$$\Delta\omega_{i,j} = \epsilon \left[\sum_k (\tau_k - g_k) \times g_k(1-g_k) \times (\nu_{j,k})\right] \times g_j(1-g_j) \times r_i$$

$$\Delta\omega_{i,j} = \epsilon \left[\sum_k \xi_k\left(\nu_{j,k}\right)\right] \times g_j\left(1-g_j\right) \times r_i$$

$$\Delta\omega_{i,j} = \epsilon \xi_j\, r_i \tag{9}$$

The Eq. (9) that describe the process of updating the weight $\Delta\omega_{i,j}$ in FFNN within the context of 5G signal streaming simulations can be explained in a sequence of steps. The weight update expression is initially introduced, $\Delta\omega_{i,j} \propto -\left[\sum_k \frac{\partial E}{\partial g_k} \times \frac{\partial g_k}{\partial h_k} \times \frac{\partial h_k}{\partial g_j}\right] \times \frac{\partial g_j}{\partial h_j} \times \frac{\partial h_j}{\partial \omega_{i,j}}$. By simplifying the given expression, it can be simplified to a more concise form $\Delta\omega_{i,j} = -\epsilon\left[\sum_k \frac{\partial E}{\partial g_k} \times \frac{\partial g_k}{\partial h_k} \times \frac{\partial h_k}{\partial g_j}\right] \times \frac{\partial g_j}{\partial h_j} \times \frac{\partial h_j}{\partial \omega_{i,j}}$. More simplifying results in $\Delta\omega_{i,j} = \epsilon\left[\sum_k (\tau_k - g_k) \times g_k(1-g_k) \times (\nu_{j,k})\right] \times g_k(1-g_k) \times r_i$. Highlighting the relationship between target output (τ_k) the predicted output g_k the weights $\nu_{j,k}$, and the input signal r_i. Expanding upon to $\Delta\omega_{i,j} = \epsilon\left[\sum_k (\tau_k - g_k) \times g_k(1-g_k) \times (\nu_{j,k})\right] \times g_j(1-g_j) \times r_i$. Eq. (9) concisely highlights the combined influence of the error sensitivity (ξ_k) and the network's activation states on the weight update. The final form $\Delta\omega_{i,j} = \epsilon\xi_j\, r_i$ illustrate the culmination of these considerations, where $\Delta\omega_{i,j} = \epsilon\left[\sum_k \xi_k\left(\nu_{j,k}\right)\right] \times g_j\left(1-g_j\right) \times r_i$ captures the collective influence of the error sensitivity (ξ_k) and the weights $\nu_{j,k}$ on the activation state of the network. By updating this rule.

$$\nu^{+}_{j,k} = \nu_{j,k} + \lambda_F \Delta\upsilon_{j,k} \tag{10}$$

Eq. (10) outlines the iterative process for updating the weights (ν) that connect the hidden layer to the output layer. $\nu^{+}_{j,k}$ denotes the revised weight, $\nu_{j,k}$ refers to the existing weight, λF represents the learning rate, and $\lambda_F \nu_{j,k}$ signifies the weight modification. Eq. (10) represents the modification of weights using the learning rate (λF) and the computed weight change $\Delta\nu_{j,k}$ obtained from the backpropagation process. The objective is to iteratively refine these weights to minimize the error and improve the FFNN's accuracy in predicting the signal output. The learning rate (F) regulates the magnitude of these adjustments, guaranteeing a trade-off between stability and convergence in the learning process. The symbol represents the network's ability to adapt and learn by continuously refining the weights to improve the accuracy of predicting the output signals. The learning rate (λ_F) ensures a controlled and effective adjustment, aligning the network's predictions with the complex dynamics of 5G signal scenarios.

$$\omega^{+}_{i,j} = \omega_{i,j} + \lambda_F \Delta\omega_{i,j} \tag{11}$$

Eq. (11) provides the procedure for updating the weights between the input and hidden layers in an FFNN for MSR2C (Multi-Stage Re-encoder to Classifier) learning in error-free signal reception. The expression represents the iterative update of the weight $\omega^{+}_{i,j}$ that connects the i_{th} node in the input layer to the j_{th} node in the hidden layer. $\omega^{+}_{i,j}$ describes the updated weight, $\omega_{i,j}$ is the current weight, λ_F is the learning rate, and $\Delta\omega_{i,j}$ does the backpropagation process determine the weight change. This process is vital for enhancing the network's capacity to capture and depict the input signal's characteristics precisely. The learning rate (λ_F) is essential for the convergence of MSR2C-ABPNN. The learning rate determines the magnitude of weight adjustments during the learning process. Choosing the optimal value for λ_F is necessary, as it requires finding a delicate balance between the rate at which the system converges, the reliability of the learning process at which convergence occurs, and the stability to efficiently learn and adapt to the complexities of the signal data. This, in turn, results in accurate signal reception without any errors.

IV. Result and Discussions

A. ANN (FFNN and BPNN) Performance

The input parameters are encoded as a vector corresponding to the neural network's input layer. The parameters being used have been multiplied by each of their weights (*w*) and a bias that is constant (*b*), leading to the computation of the output function (g_j) for each node in the hidden layer ∑ summation symbol. The primary purpose of an activation function is to limit the size of the summation of input values for all of the nodes inside a hidden layer. This restricted summation then operates as the input value for the nodes in the succeeding layer. An identical mathematical technique determines how information is sent from the hidden layer to the output layer. The back-propagation algorithm is a supervised learning approach requiring a dataset including input-output vector pairs. The input parameters are integrated as vectors into the network corresponding to a node in the input layer. The input parameters are further multiplied by their corresponding weights (*w*) and a constant bias (*b*), resulting in the output function (g_k) for every node in the hidden layer. $\Delta\upsilon$ weight change in neuron, τ_k: target neuron, the purpose of an activation function is to restrict the magnitude of the summation of every single node in a hidden layer, which serves as the input for the subsequent layer's nodes. The method of information transfer between the hidden layer and output layer is consistent and adheres to the same mathematical principles. The back-propagation algorithm provides a supervised learning technique requiring a dataset including pairs of input-output vectors. Multiple neural networks are developed to examine and compare the performance of different transfer functions. Each data set training, validation, and testing accuracy signal is presented in Figs. 2 and 3. Fig. 2 describes the best validation through a feedforward MLP, while Fig. 3 defines the result of backpropagation. Based on the information mentioned above, the entire performance was evaluated. Initially, the transfer functions applicable to feedforward MLP are being considered in Fig. 2. The findings are examined and analyzed, presenting the outcomes derived from implementing a neural network whereby different combinations of transfer functions were employed in both the hidden and output layers. The findings were computed using MATLAB, a numerical computing environment incorporating many paradigms. The evaluation of the optimal ANN model was examined regarding its performance in testing, validation, and training. When selecting the optimal detection model, it is crucial to prioritize its detection capability, with the performance seen during testing serving as the primary criterion. The generalization ability of a neural network model is influenced by the number of neurons in the hidden layer of AN models. Therefore, an alternative experimental method was employed to determine the optimal number of neurons in the hidden layer. Various network topologies were assessed, examining varying quantities of hidden neurons ranging from 50 to 150. The selection of the optimal network model was determined by evaluating the Mean Squared Error (MSE). To mitigate the potential impact of random correlation resulting from the random initialization of weights, the performance testing was conducted 500 times for each network architecture.

The findings of this research indicate that the ANN network frameworks were evaluated and compared according to how they performed. The network with 50–150 hidden neurons was observed to have a superior generalization capacity compared to other networks. Using Tan Sigmoid transfer functions in the two corresponding inputs-hidden and hidden-output layers-resulted in optimal translation performance. Various algorithms for training were evaluated, and it was found that the Bayesian technique had superior performance compared to other training techniques, as evidenced by lower MSE values obtained during testing. The efficacy of the optimal model is seen in Fig. 2 and 3, where the MSE attains its lowest values. The validation, training, testing, and overall have been determined as optimal correlation. The most effective network configuration is described in Fig. 4 (performance validated through Cross-validation (1000×3626) and Fig. 5 (performance validated through linear regression) through ML cross-validation and regression, achieving a remarkable accuracy level of 99.99%. Each network model in ANN serves as an approximation of an actual system. Therefore, getting a deeper understanding of the internal information in the data obtained from the modeled network is crucial.

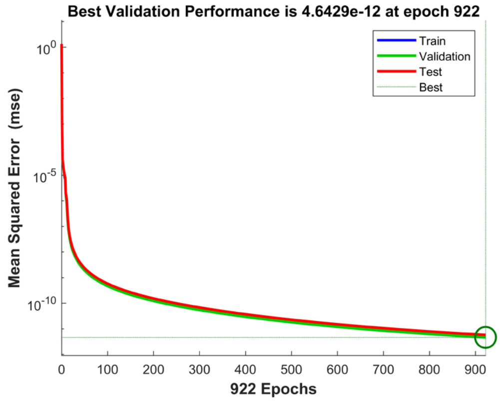

Fig. 2. Best validation through FFNN.

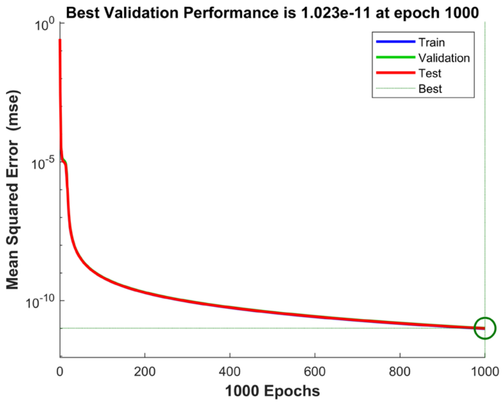

Fig. 3. Best validation through BPNN.

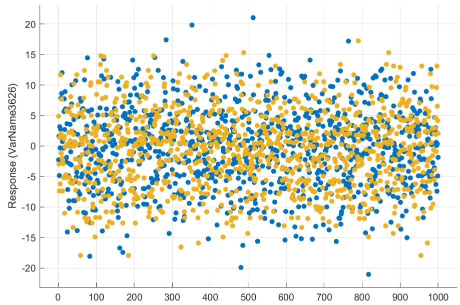

Fig. 4. Cross validation.

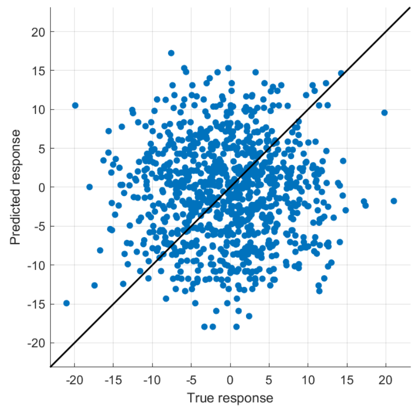

Fig. 5. Linear regression.

B. Feed Forward Neural Network Results

Fig. 6 describes the framework receiving the data in discrete segments, after which the output functions are generated. The evaluation and validation findings for the suggested approach's training and validation accuracy levels are provided in Figs. 7 and 8. Increasing the number of iterations improves both the learning and validation phases' precision, reliability, and accuracy. According to Figs. 7 and 8, the chosen FFNN model has effectively addressed the problem of signal accuracy. As a result, it is an excellent candidate for delivering the higher accuracy required in signal classification. FFNN's highest performance is 99.9%, far more perfect than other models. As a result, FFNN's efficiency is adequate.

TABLE II. DATA SPLIT RATIO

Samples	Testing	Validation
2538	544	544
750	150	150

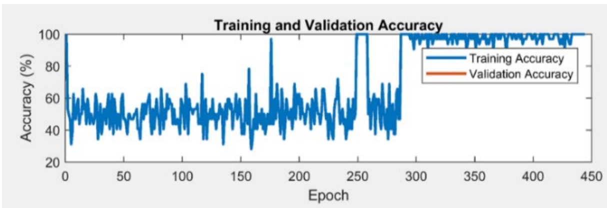

Fig. 6. Signals accuracy.

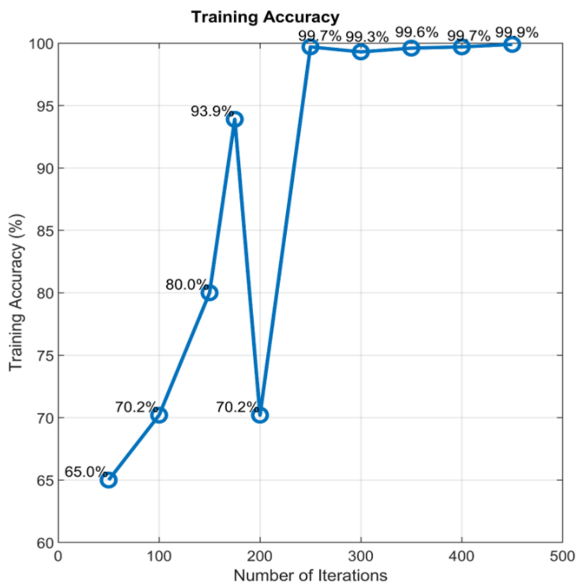

Fig.7. Training accuracy of FFNN.

Table II describes the data split ratio. Fig. 7 presents important knowledge of the model's learning process to attain error-free signal reception, as delineated in the mathematical model equations. The X-axis labeled "Number of Iterations" represents the training epochs associated with the iterations conducted on the training data. The "Training Accuracy" Y-axis concurrently indicates the model's performance on an independent validation dataset, expressed as a percentage. Commencing with 50 iterations, the model achieves a notable validation accuracy of 65%, demonstrating its proficiency in accurately classifying 99% of the validation dataset. Performance generally increases between 100 and 400 cycles, aligning with the iterative optimization process in learning models. The activation functions in Eqs. (1) and (2) augment the model's capacity to adjust to the training data. Upon completing 250 cycles, the validation accuracy attains an impressive 99.7%, demonstrating substantial advancement in reducing signal errors. The training accuracy significantly enhanced, varying from 99.6% to 99.9% during 200–450 iterations. This signifies that the model attains high accuracy, with additional training yielding only negligible enhancements. Eq. (4), governing the output layer, is essential for attaining high precision. The study's findings, articulated using equations, demonstrate that the ANN exhibits remarkable performance from beginning to end and continually enhances its accuracy throughout the training cycles. The combined effect of error sensitivity, activation functions, and iterative optimization improves the adaptability of FFNN. Nonetheless, while analyzing these findings, it is essential to account for variables and the practicality of their use in real-world contexts. The accurate reception of signals is inferred from the reduction in validation errors, illustrating the model's proficiency in learning and effectively managing signals during training.

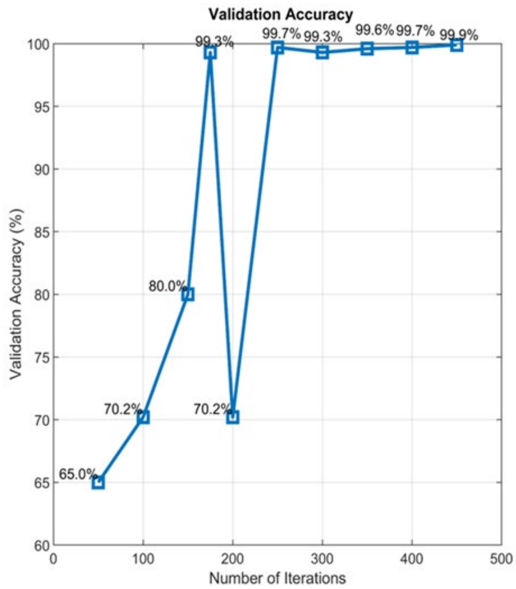

Fig. 8. Validation accuracy.

Fig. 8 presents a comprehensive study on improving the FFNN methodology's validation accuracy over different training iterations or epochs. The X-axis, labeled "Number of Iterations," denotes the extent of training the algorithm has received, whilst the Y-axis, labeled "Validation Accuracy," reflects the system's proficiency in data classification, expressed as a percentage. The algorithm's performance steadily increases during the first training phase, namely iterations 50 to 450. Commencing with an initial accuracy of 65% at the 50th iteration, the performance steadily improves, concluding in a validation accuracy of 99.9% at the 450th iteration. This exhibits the algorithm's ability to learn and modify actions based on the supplied training data, as Eq. (2) demonstrated. A rapid improvement rate occurs following the first phase (iterations 50 to 400). The validation accuracy increases from 65% to 99.7%, indicating substantial advancement in data classification efficacy. This aligns with the iterative optimization outlined in Eqs. (6) and (7), wherein the algorithm modifies its parameters to minimize mistakes and improve accuracy. The concluding phase (iterations 200 to 400) demonstrates consistent improvement, achieving a validation accuracy of 99.7% at iteration 400, which then escalates to an impressive 99.9% by iteration 450. These data suggest that the algorithm is progressively approaching an unusually high degree of accuracy, maybe reaching its optimal efficiency. The improvement in accuracy aligns with the adaptive learning illustrated by Eqs. (4) and (5), showcasing the model's ability to refine its predictions perpetually. This presents evidence of the algorithm's ability to improve accuracy with each successive training cycle. The algorithm begins with a satisfactory level of accuracy and enhances its performance, ultimately achieving an extraordinarily high validation accuracy described in Fig. 9.

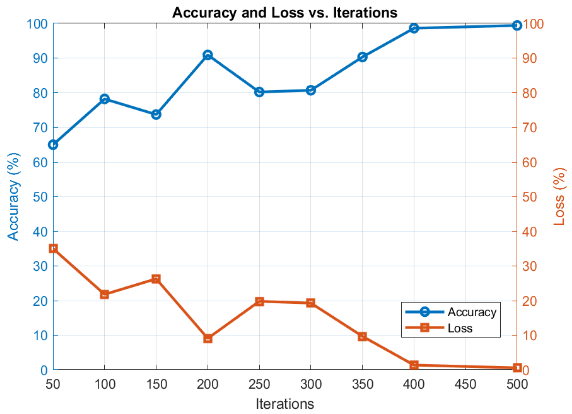

Fig. 9. Accuracy vs. loss.

The outcomes demonstrate that the algorithm is proficiently designed for its intended purpose, yielding precise predictions when used on previously unobserved data. It is crucial to carefully evaluate the model's practical applicability in real-world contexts.

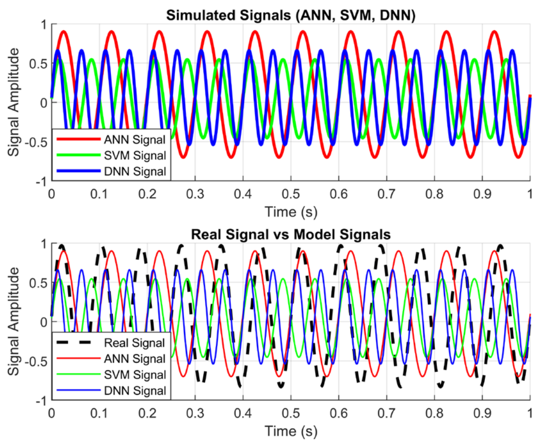

Fig. 10. Comparison of ANN, SVM, and DNN.

C. Comparison of ANN, SVM, and DNN

The optimal signal is described in Fig. 10 and attains exceptional accuracy. ANN signal simulates an actual structure for signal classification and efficient transmission. The algorithm is specifically developed to identify the most efficient transmission method for encrypting signals. The primary stage in facilitating the model's training and testing requires establishing signal parameters, such as the sampling frequency and time vector. The training dataset constructs and trains an ANN with certain layers and configurations. Accuracy is verified by comparing the expected signal with the actual signal. The precision of ANN is higher than that of SVM and DNN. The SVM attains an accuracy of 54% in the identical context, while DNN has 66%. Fig. 10 illustrates the accuracy of ANN. These results demonstrate the accuracy attained by the Ann model. Additionally, Fig. 10 compares ANN, SVM, and DNN, which depicts the expected results of signals compared to real signals.

V. Conclusion

The simulated 5G signal streaming framework featuring FFNN provides valuable where iterative learning of the model, adapting to fading channels and noise, underscores its potential for enhancing signal reliability. Promising improvements in validation accuracy were observed in the framework in refining 5G signal processing for optimized wireless communication systems. The present study involves the development of simulations for accurate signal categorization and optimal transmission. Specifically, the algorithm is designed to select the most efficient transmission method for encrypting a given signal. To facilitate the training and testing of the model, the initial step involves defining signal parameters. The training dataset generates and trains an FFNN with specific layers and choices. Accuracy is determined by comparing the anticipated labels to the actual labels. The accuracy of artificial neural networks FFNNs is 99.9% achieved in received signals. The gradients guide weight adjustments during training, enabling the ANN to minimize errors and effectively learn to approximate desired signal outputs. These results demonstrate the accuracy attained by the Ann model. Additionally, compares ANN, SVM, and DNN, which depicts the expected results of signals compared to real signals. Results illustrate the ANN's role in signal processing, emphasizing its training process and optimization objectives.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, FA, HY; methodology, FA, HY, CWD; software, FA, AR; validation, FA, AR, DA, AB, MASK; formal analysis, FA; investigation, FA HY; resources, HY; data curation, FA, AB, CWD AR; writing—original draft preparation, FA; writing—review and editing, FA, HY, AB, CWD, AR, MASK, DA;

visualization, FA; supervision, HY; project administration, HY; all authors had approved the final version.

FUNDING

This work was supported by the National Natural Science Foundation of China under Grant No. 51977153, 51977161, 51577046, the State Key Program of the National Natural Science Foundation of China under Grant No. 51637004, the national key research and Development plan "Important Scientific Instruments and Equipment development" Grant No.2016YFF0102200, Equipment research project in advance Grant No.41402040301.

REFERENCES

[1] F. Ali and H. Yigang, "Spectrum for next generation technologies," in *Proc. the 2019 8th International Conf. on Software and Information Engineering*, New York, 2019, pp. 188–191.

[2] F. Ali, Y. He, G. Shi *et al.*, "Future generation spectrum standardization for 5G and internet of things.," *J. Commun.*, vol. 15, no. 3, pp. 276–282, 2020.

[3] A. Al-Jumaily, A. Sali, V. P. G. Jiménez *et al.*, "Evaluation of 5G and Fixed-Satellite Service Earth Station (FSS-ES) downlink interference based on Artificial Neural Network Learning Models (ANN-LMS)," *Sensors*, vol. 23, no. 13, 6175, 2023. doi: 10.3390/s23136175

[4] J. N. Hwang, S. Y. Kung, M. Niranjan *et al.*, "The past, present, and future of neural networks for signal processing," *IEEE Signal Process Mag.*, vol. 14, no. 6, pp. 28–48, 1997. doi: 10.1109/79.637299

[5] A. Al-Jumaily, A. Sali, M. Riyadh *et al.*, "Machine learning modeling for radiofrequency electromagnetic fields (RF-EMF) signals from mmwave 5G signals," *IEEE Access*, vol. 11, pp. 79648–79658, 2023. doi: 10.1109/ACCESS.2023.3265723

[6] Q. Li, P. Sisk, A. Kannan *et al.*, "Machine learning based time domain millimeter-wave beam prediction for 5G-advanced and beyond: design, analysis, and over-the-air experiments," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 6, pp. 1787–1809, 2023. doi: 10.1109/JSAC.2023.3275613

[7] F. Ali and H. Yigang, "Spectrum sensing-focused cognitive radio network for 5G revolution," *Front Environ Sci*., vol. 11, 2023. doi: 10.3389/fenvs.2023.1113832

[8] P. A. D. N. Jayawardana, H. Obaid, T. Yesilyurt *et al.*, "Machine-learning-based los detection for 5G signals with applications in airport environments," *Sensors*, vol. 23, no. 3, 1470, 2023. doi: 10.3390/s23031470

[9] Q. Xiang, X. Wang, J. Lai *et al.*, "5G network reference signal receiving power prediction based on multilayer perceptron," in *Proc. 2022 China Automation Congress (CAC)*, Xiamen, 2022, pp. 19–24. doi: 10.1109/CAC57257.2022.10055904

[10] K. Tonchev, A. Ivanov, N. Neshov *et al.*, "Learning graph convolutional neural networks to predict radio environment maps," in *Proc. 2022 25th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, Herning, 2022, pp. 392–395. doi: 10.1109/WPMC55625.2022.10014842

[11] Z. Xiang, W. Niu, P. Zhao *et al.*, "5G wireless intelligent propagation model and application of simulation engine in production," in *Proc. 2022 IEEE 2nd International Conf. on Information Communication and Software Engineering (ICICSE)*, Chongqing, 2022, pp. 126–131. doi: 10.1109/ICICSE55337.2022.9828923

[12] Y. Liu, J. Dong, W. Huangfu *et al.*, "3.5 GHz outdoor radio signal strength prediction with machine learning based on low-cost geographic features," *IEEE Trans Antennas Propag*, vol. 70, no. 6, pp. 4155–4170, 2022. doi: 10.1109/TAP.2022.3175214

[13] D. Gao, Q. Guo, G. Liao *et al.*, "Signal detection in mimo systems with hardware imperfections: Message passing on neural networks," *IEEE Trans Wirel Commun.*, vol. 23, no. 1, pp. 820–834, 2024. doi: 10.1109/TWC.2023.3283275

[14] S. Ko, M. Shin, G. Kim *et al.*, "System-informed neural network for frequency detection," *IEEE Signal Process Lett*, vol. 31, pp. 2980–2984, 2024. doi: 10.1109/LSP.2024.3483036

[15] J. A. Castro-Correa, J. H. Giraldo, M. Badiey *et al.*, "Gegenbauer graph neural networks for time-varying signal reconstruction," *IEEE Trans Neural Netw Learn Syst.*, vol. 35, no. 9, pp. 11734–11745, 2024. doi: 10.1109/TNNLS.2024.3381069

[16] S. Jain, A. K. Yadav, R. Kumar *et al.*, "Artificial neural network based spectrum sensing in wireless regional area network," *IEEE Access*, pp. 48941–48950, 2024. doi: 10.1109/ACCESS.2024.3384532

[17] S. Zargari, A. Hakimi, C. Tellambura *et al.*, "Enhancing AmBC systems with deep learning for joint channel estimation and signal detection," *IEEE Transactions on Communications*, vol. 72, no. 3, pp. 1716–1731, 2024. doi: 10.1109/TCOMM.2023.3333868

[18] A. Volkov, C. Jie, A. Gorelik *et al.*, "Classification of radio signals in multipath fading channel using neural network," in *Proc. 2024 Conf. of Young Researchers in Electrical and Electronic Engineering (ElCon)*, Saint Petersburg, 2024, pp. 919–923. doi: 10.1109/ElCon61730.2024.10468419

[19] A. A. Jihado and A. S. Girsang, "Hybrid deep learning network intrusion detection system based on convolutional neural network and bidirectional long short-term memory," *Journal of Advances in Information Technology*, vol. 15, no. 2, pp. 219–232, 2024.

[20] M. Gao, Y. Guo, Z. Liu *et al.*, "Feed-forward neural network blind equalization algorithm based on super-exponential iterative," in *Proc. 2009 International Conf. on Intelligent Human-Machine Systems and Cybernetics*, Hangzhou, 2009, pp. 335–338. doi: 10.1109/IHMSC.2009.92

[21] S. Liu, P. Reviriego, X. Tang *et al.*, "Result-based re-computation for error-tolerant classification by a support vector machine," *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 1, pp. 62–73, 2020. doi: 10.1109/TAI.2020.3028321

[22] E. Viera-Martin, J. F. Gómez-Aguilar, J. E. Solís-Pérez *et al.*, "Artificial neural networks: A practical review of applications involving fractional calculus," *Eur Phys J Spec Top*., vol. 231, no. 10, pp. 2059–2095, 2022. doi: 10.1140/epjs/s11734-022-00455-3

[23] C. Zhang and M. van der Baan, "Microseismic signal reconstruction from strong complex noise using low-rank structure extraction and dual convolutional neural networks," *IEEE Trans Neural Netw Learn Syst.*, vol. 35, no. 11, pp. 15554–15564, 2024. doi: 10.1109/TNNLS.2023.3288142

[24] S. Ko, M. Shin, G. Kim *et al.*, "System-informed neural network for frequency detection," *IEEE Signal Process Lett.*, vol. 31, pp. 2980–2984, 2024. doi: 10.1109/LSP.2024.3483036

[25] S. Wang, S. Zhang, J. Ma *et al.*, "Graph-neural-network-based wifi indoor localization system with access point selection," *IEEE Internet Things J*., vol. 11, no. 20, pp. 33550–33564, 2024. doi: 10.1109/JIOT.2024.3430087

[26] M. Baleya, H. Shalaby, K. Kato *et al.*, "Neural network ensemble for precise laser spot position determination on a quadrant detector," *IEEE Photonics Technology Letters*, vol. 36, no. 2, pp. 115–118, 2024. doi: 10.1109/LPT.2023.3337722

[27] N. Shruthi and K. Ramesha, "A deep learning based hybrid precoding scheme with limited feedback approach for improved compression and minimized reconstruction error in massive MIMO," *Journal of Advances in Information Technology*, vol. 16, no. 1, pp. 91–100, 2025. doi: 10.12720/jait.16.1.91-100

[28] W. H. Lee and M. Kim, "NsigNet: A neural network design for detecting the number of signals under sparse observations," *IEEE Internet Things J*., vol. 11, no. 11, pp. 19355–19367, 2024. doi: 10.1109/JIOT.2024.3350636

[29] S. Hamedi and H. D. Jahromi, "Performance analysis of all-optical logical gate using artificial neural network," *Expert Syst Appl*., vol. 178, 115029, 2021. doi: 10.1016/j.eswa.2021.115029

[30] H. D. Jahromi and S. Hamedi, "Artificial intelligence approach for calculating electronic and optical properties of nanocomposites," *Mater Res Bull*., vol. 141, 111371, 2021. doi: 10.1016/j.materresbull.2021.111371

[31] S. Hamedi, H. D. Jahromi, and A. Lotfiani, "Artificial intelligence-aided nanoplasmonic biosensor modeling," *Eng Appl Artif Intell*., vol. 118, 105646, 2023. doi: 10.1016/j.engappai.2022.105646

[32] P. B. Duong, B. Ghimire, K. Dietmayer *et al.*, "Supervised machine learning assisted hybrid positioning based on GNSS and 5G," in *Proc. 2022 IEEE 12th International Conf. on Indoor Positioning and Indoor Navigation (IPIN)*, Beijing, 2022, pp. 1–8. doi: 10.1109/IPIN54987.2022.9918098

[33] S. Chaudhari and H. Kwon, "Machine learning based interference whitening in 5G NR mimo receiver," in *Proc. 2022 IEEE 95th Vehicular Technology Conf.: (VTC2022-Spring)*, Helsinki, 2022, pp. 1–6. doi: 10.1109/VTC2022-Spring54318.2022.9860578

[34] R. U. Mageswari, Gousebaigmohammad, D. Dulam *et al*., "Machine learning empowered accurate CSI prediction for large-scale 5G networks," *Wirel Commun Mob Comput*., vol. 2022, pp. 1–6, 2022. doi: 10.1155/2022/7085731

[35] N. Singh, S. Agrawal, T. Agarwal *et al*., "RBF-SVM based resource allocation scheme for 5G cran networks," in *Proc. 2018 3rd International Conf. and Workshops on Recent Advances and Innovations in Engineering (ICRAIE)*, Jaipur, 2018, pp. 1–6. doi: 10.1109/ICRAIE.2018.8710423

[36] S. Imtiaz, H. Ghauch, G. P. Koudouridis *et al*., "Random forests resource allocation for 5G systems: Performance and robustness study," in *Proc. 2018 IEEE Wireless Communications and Networking Conf. Workshops (WCNCW)*, Barcelona, 2018, pp. 326–331. doi: 10.1109/WCNCW.2018.8369028

[37] S. Aldossari and K. C. Chen, "Predicting the path loss of wireless channel models using machine learning techniques in mmwave urban communications," in *Proc. 2019 22nd International Symposium on Wireless Personal Multimedia Communications (WPMC)*, Lisbon, 2019, pp. 1–6. doi: 10.1109/WPMC48795.2019.9096057

[38] S. Sevgican, M. Turan, K. Gokarslan *et al*., "Intelligent network data analytics function in 5G cellular networks using machine learning," *Journal of Communications and Networks*, vol. 22, no. 3, pp. 269–280, 2020. doi: 10.1109/JCN.2020.000019

[39] D. Marasinghe, N. Jayaweera, N. Rajatheva *et al*., "Hierarchical user clustering for mmwave-noma systems," in *Proc. 2020 2nd 6G Wireless Summit (6G SUMMIT)*, Levi, 2020, pp. 1–5. doi: 10.1109/6GSUMMIT49458.2020.9083909

[40] N. Ravi, P. V. Rani, and S. M. Shalinie, "Secure Deep Neural (SeDeN) framework for 5G wireless networks," in *Proc. 2019 10th International Conf. on Computing, Communication and Networking Technologies (ICCCNT)*, Kanpur, 2019, pp. 1–6. doi: 10.1109/ICCCNT45670.2019.8944654

[41] A. Mazin, M. Elkourdi, and R. D. Gitlin, "Accelerating beam sweeping in mmwave standalone 5G new radios using recurrent neural networks," in *Proc. 2018 IEEE 88th Vehicular Technology Conf. (VTC-Fall)*, Chicago, 2018, pp. 1–4. doi: 10.1109/VTCFall.2018.8690810

[42] P. Jaraut, G. C. Tripathi, M. Rawat *et al*., "Independent component analysis for multi-carrier transmission for 4G/5G power amplifiers," in *Proc. 2017 89th ARFTG Microwave Measurement Conf. (ARFTG)*, Honololu, 2017, pp. 1–4. doi: 10.1109/ARFTG.2017.8000817

[43] F. Ali and H. Yigang, "A novel approach of optimal signal streaming analysis implicated supervised feedforward neural networks," *IET Signal Processing*, vol. 2024, no. 1, 2024. doi: 10.1049/2024/2819057

[44] C. Aggarwal, "The backpropagation algorithm," in *Neural Networks and Deep Learning: A Textbook*, C. C. Aggarwal, Ed., Cham: Springer International Publishing, 2023, pp. 29–71. doi: 10.1007/978-3-031-29642-0_2