(Received 26 October 2024, Accepted 2 December 2024, Scheduled 11 December 2024)

Application of Attention Mechanism-Enhanced BiLSTM-CNN in Power Amplifier Behavioral Modeling and Predistortion

Jingchang Nan*, Shize Liu, and Jiadong Yu

School of Electronics and Information Engineer, Liaoning University of Engineering and Technology, Liaoning, China

ABSTRACT: Power amplifiers in wireless communication systems can introduce nonlinear distortion, degrade signal transmission quality, and increase power consumption. The paper presents a BiLSTM-CNN-based model for modelling power amplifier behaviour to address this issue. The model uses BiLSTM layers to capture temporal information from the signal data and incorporates a multi-head attention mechanism to focus on different temporal features. Additionally, convolutional layers process global features and reduce model parameters through weight sharing. Using this model, a digital pre-distortion (DPD) model is proposed to linearise the power amplifier through an indirect learning approach. The results show that the BiLSTM-CNN model achieves a normalised mean square error (NMSE) of -40.3 dB, and the DPD model enhances the adjacent channel power ratio (ACPR) of the communication system by 18 dB, demonstrating the model's feasibility. Comparative analysis with other network models indicates that BiLSTM-CNN outperforms traditional methods of fitting performance and convergence speed, showcasing its superiority.

1. INTRODUCTION

Power amplifier (PA) is essential in wireless communication systems. Its purpose is to increase the power of wireless signals, ensuring the stability and reliability of the signal throughout transmission [1–3]. Nevertheless, in real-world scenarios, the inherent nonlinear properties of the device and the interactions between signals in the power amplifier result in nonlinearities that affect its performance [4–6]. Behavioural modelling offers a powerful approach to analyze and model the nonlinear properties of a power amplifier. It does this by recording the input-output response of the amplifier and creating a nonlinear mapping relationship to replicate its internal operations. This enables the implementation of digital pre-distortion technology to mitigate the nonlinear effects of power amplifiers [7].

Traditional behavioural modelling methods include memoryless models and memory models. Early memoryless models were designed for the behavioural modelling of power amplifiers with narrowband signals and included Saleh model [8], Hammerstein model [9], and Wiener model [10]. As the bandwidth of modern signals is expanded, the impact of nonlinear memory effects in power amplifiers has become more pronounced, rendering the early memoryless models less effective. Subsequently, Volterra series and piecewise linear function models were introduced for modelling wideband power amplifiers [11]. These traditional behavioural modelling approaches represent the nonlinear relationships of power amplifiers using polynomial expressions.

In recent years, neural networks have garnered increasing attention in power amplifier behaviour modelling due to their high accuracy in fitting nonlinear systems [12–14]. Researchers have applied feedforward neural networks to power amplifier behaviour modelling and, considering the memory effects of wideband power amplifiers, introduced real-valued time-delay artificial neural networks (RVTDANN) [15]. Subsequently, memory-capable recurrent neural networks (RNNs) were utilised for power amplifier modelling. With the continuous advancement of recurrent neural networks, variants such as Long Short-Term Memory (LSTM) networks and Bidirectional Long Short-Term Memory (BiLSTM) networks have been incorporated into power amplifier behaviour modelling [16–18]. These approaches achieve high-precision modelling of power amplifiers; however, the complexity of the LSTM and BiLSTM models, due to their numerous internal parameters, results in longer training times.

To address the issue of complex parameters in LSTM network models, this paper proposes a BiLSTM-CNN model. The model's front end utilises a BiLSTM network to capture the temporal features of the input signals of the power amplifier. In contrast, the back end employs a Convolutional Neural Network (CNN) to process these features further. By leveraging the parameter-sharing properties of CNNs, the model reduces the overall number of parameters. Additionally, integrating a multi-head attention mechanism within the model allows for controlling data weights across different time steps. To evaluate the model's performance, the BiLSTM-CNN model is compared with other existing neural network models. Experimental results indicate that the proposed BiLSTM-CNN model achieves an NMSE of -39.3 dB, demonstrating a higher degree of fit than other models.

This paper presents the following contributions.

1. The proposed design based on BiLSTM mitigates the influence of memory effects in broadband power amplifiers.

* Corresponding author: Jingchang Nan (nanjingchang@lntu.edu.cn).

2. To tackle the problem of an excessive number of intricate model parameters, a fusion of CNN and BiLSTM is employed. This reduces the number of parameters in the model, leading to a drop in complexity and improved efficiency and interpretability.

3. An attention mechanism is introduced, allowing the model to dynamically assign weights to signals at different time points in the input sequence. This enhances the model's focus on important information, improving accuracy and efficiency.

4. The paper demonstrates excellent PA modelling and linearisation performance and achieves PA linearisation through the inverse model of BiLSTM-CNN.

The remainder of this work is organised as follows. Section 2 presents an overview of previous research on power amplifiers' properties and LSTM models' architecture. Section 3 details the proposed model structure. Section 4 delves into the theoretical concepts underlying the approach, and Section 5 extends this approach to the design of DPD. Section 6 discusses the empirical results, and Section 7 concludes with final remarks.

2. RELATED WORK

2.1. The Characteristics of Power Amplifiers

PA comprises amplification elements such as transistors or field-effect transistors exhibiting nonlinear characteristics [19]. As the input signal power to the PA increases, the amplifier enters its saturation region, leading to significant nonlinear distortion. The nonlinear distortion in the PA results in amplitude (AM/AM) distortion and phase (AM/PM) distortion [20]. With the increasing bandwidth of input signals in modern communications, the AM-AM and AM-PM characteristics of the PA evolve into scatter plots. In this scenario, the PA's output signal depends on the current and past input signals, giving rise to memory effects. This makes the behavioural modelling of the PA particularly important, as it allows for an accurate description of the PA's nonlinearities and memory effects. Effective digital pre-distortion (DPD) algorithms can be developed by constructing behavioural models to compensate for the PA's nonlinear distortion [21].

When evaluating the performance of power amplifiers, two commonly used key metrics for quantifying nonlinear distortion are Error Vector Magnitude (EVM) and Adjacent Channel Power Ratio (ACPR) [22]. EVM measures the deviation between the output signal and ideal signal, reflecting the quality of the signal. It is commonly used in communication systems to evaluate modulated signals. In addition to EVM, the correlation between input and output signals is also an effective method for assessing signal similarity [23]. By applying time shifts and amplitude scaling to the output signal to align its main peak with that of the input signal and then calculating the correlation coefficient, this method provides a more comprehensive evaluation of the signal similarity. Compared to EVM, which directly quantifies the error magnitude, correlation may be more sensitive to certain types of nonlinear distortion, especially in cases of severe distortion, providing a more stringent performance evaluation. ACPR, on the other hand, is used to assess the effectiveness of Digital Predistortion (DPD) by measuring the average power ratio between the main communication channel and adjacent channels, quantifying the impact of nonlinear distortion. Nonlinear distortion in power amplifiers leads to spectral regeneration, causing power leakage into adjacent channels, which can reduce the communication quality of nearby channels and even result in intermodulation and distortion. A lower ACPR indicates less adjacent channel leakage, thus smaller nonlinear distortion. DPD technology typically reduces ACPR to mitigate adjacent channel interference, thereby improving the overall performance of communication systems.

$$\mathrm{ACPR} = \frac{P_{\mathrm{adj}}}{P_{\mathrm{main}}} \tag{1}$$

where P_{adj} denotes the average power of the adjacent channels, and P_{main} represents the average power of the main channel.

2.2. LSTM Neural Network Models

Long Short-Term Memory (LSTM) is a recurrent neural network (RNN) that primarily processes text and time-series data [24]. LSTM uses gate controls and memory units to keep and process information over extended periods. The internal construction of an LSTM module is shown in Figure 1.

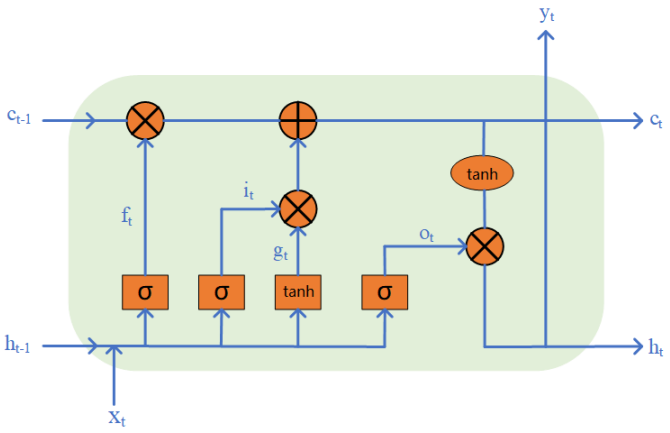

FIGURE 1. Internal structure of LSTM module.

The LSTM network essentially regulates the transmission of information using gate mechanisms, guaranteeing consistent gradient flow and efficient storing and updating of details while handling extended sequences. The LSTM model consists of three primary gate units: the input gate, forget gate, and output gate [25]. The forget gate determines which information from the previous cell state should be discarded. The model receives inputs from the current input and the last time step's hidden state, applies a sigmoid activation function to analyse them, and produces a vector ranging from 0 to 1. This vector indicates the proportion of each information component that should be retained.

$$f_t = \sigma\left[W_f \cdot (x_t + h_{t-1}) + b_f\right] \tag{2}$$

where f_t denotes the output of the forget gate; W_f and b_f represent the weight matrix and bias vector of the forget gate, respectively; and σ is the sigmoid activation function.

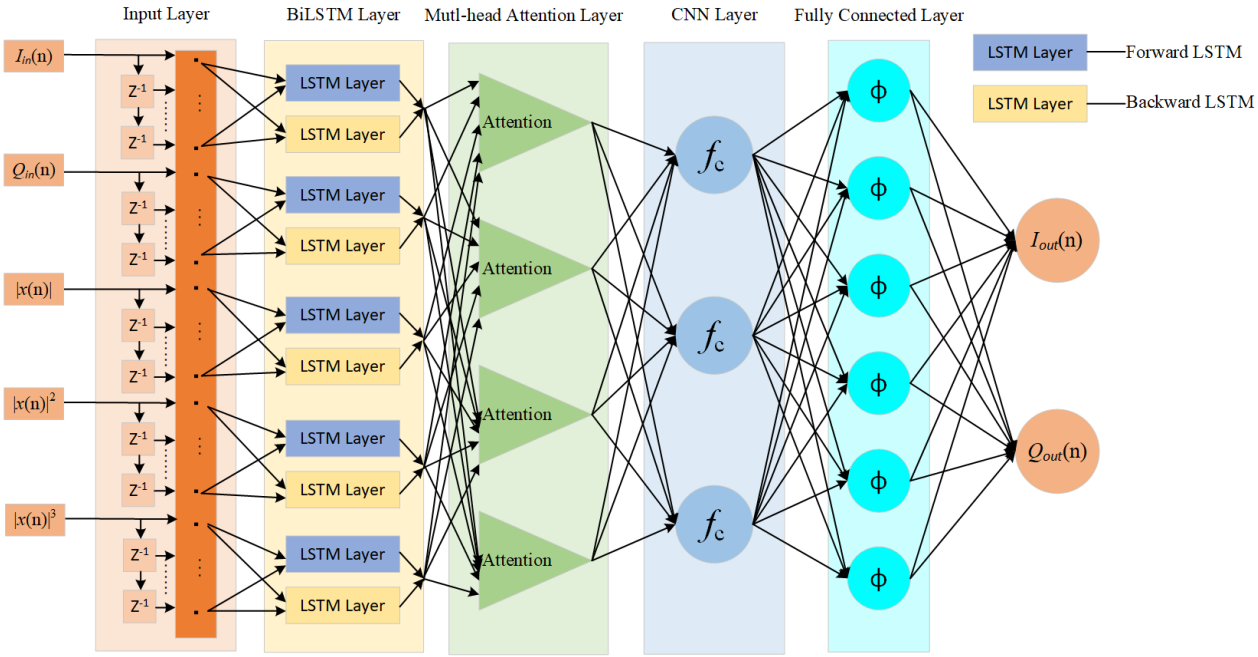

FIGURE 2. Model architecture diagram.

The output gate determines which input information from the current time step should be added to the cell state. The input gate receives the current input and the previous time step's hidden state, producing a vector i_t that indicates the proportion of new information to be written and generates a new candidate memory g_t using the Tanh activation function.

$$g_t = \tanh\left[W_g \cdot (x_t + h_{t-1}) + b_g\right] \quad (3)$$

$$i_t = \sigma\left[W_i \cdot (x_t + h_{t-1}) + b_i\right] \quad (4)$$

where i_t denotes the output of the input gate; W_i and W_g correspond to the weight matrices for the input gate and candidate memory, respectively; b_i and b_g are the bias vectors for the input gate and candidate memory, respectively.

In LSTM, the memory cell updates its state at each time step by combining the outputs of the forget gate and input gate. The state of the memory cell at the time step c_t is generated by weighting the previous time step's memory cell state c_{t-1} and the new candidate memory g_t.

$$c_t = f_t \otimes c_{t-1} + i_t \otimes g_t \quad (5)$$

The output gate controls which information will be output as the current time step's hidden state. It takes the current output and the previous time step's hidden state as input, generating a vector o_t that determines the proportion of information to be output. This vector is then passed through an applied tanh activation function c_t, producing the hidden state for the current time step h_t.

$$o_t = \sigma\left[W_o \cdot (x_t + h_{t-1}) + b_o\right] \quad (6)$$

$$y_t = h_t = o_t \otimes \tanh\left(c_t\right) \quad (7)$$

where o_t denotes the output of the output gate in the current state; W_o and b_o are the weight matrix and bias vector for the output gate, respectively; and y_t represents the output in the long-term state.

LSTM networks, through their gating mechanisms, can effectively capture and retain critical information over long time spans in sequences, making them particularly effective in the behavioural modelling of power amplifiers. Power amplifier models often exhibit complex time-series characteristics, including nonlinearity and long-term dependencies. LSTM's ability to maintain crucial historical information through memory cell states enhances its accuracy in modelling the dynamic characteristics of power amplifiers.

3. MODEL STRUCTURE

Figure 2 illustrates the BiLSTM-CNN model architecture proposed for PA behavioural modelling in this paper. It comprises five layers: input layer, BiLSTM layer, multi-head attention layer, convolutional layer, and fully connected layer. This model integrates BiLSTM and CNN models, enhancing the handling capability of time-series data while reducing model complexity. Additionally, including a multi-head attention mechanism improves the model's accuracy in fitting the nonlinear behaviours of power amplifiers.

3.1. Input Layer

In practical communication systems, signals are typically complex signals; however, neural networks can only process real-valued signals. Therefore, this paper decomposes the input complex signal into two orthogonal real components: I (in-phase component) and Q (quadrature component). To more comprehensively describe the characteristics of the signal, the envelope information of the signal is introduced into the neu-

ral network model, including the signal's amplitude $|x(t)|$, the square of the amplitude $|x(t)|^2$, and the cube of the amplitude $|x(t)|^3$ [26]. These envelope terms, along with I and Q components, are used as input features to the neural network.

To ensure that the separated I and Q components accurately reflect the characteristics of the original complex signal, the input layer first normalizes the I and Q components to ensure that they are input into the network on the same scale, eliminating any potential amplitude differences. To capture the nonlinear variations within the signal, the envelope information is also processed with multiple delays, creating features at different time steps. This allows the network to learn the dynamic changes of the signal in the time domain, improving its ability to respond to the temporal characteristics of the signal. To ensure that the phase information of the signal is effectively preserved, the input layer specifically introduces a phase calculation mechanism. By modeling the cross terms of the I and Q components, the network can learn the relative phase relationship between them, aiding in the recovery of the original phase characteristics of the signal. All input terms, including the I and Q components and envelope information, are processed with delays and then fed as the final input to the neural network, as shown below.

$$\begin{aligned}\mathbf{X}_t = & [I_{\text{in}}(t), I_{\text{in}}(t-1), \ldots, I_{\text{in}}(t-M); \\ & Q_{\text{in}}(t), Q_{\text{in}}(t-1), \ldots, Q_{\text{in}}(t-M); \\ & |x(t)|, |x(t-1)|, \ldots, |x(t-M)|; \\ & |x(t)|^2, |x(t-1)|^2, \ldots, |x(t-M)|^2; \\ & |x(t)|^3, |x(t-1)|^3, \ldots, |x(t-M)|^3] \end{aligned} \tag{8}$$

where $I_{\text{in}}(t)$ and $Q_{\text{in}}(t)$ represent the I/Q components of the envelope signal $x(t)$; $|x(t)|$ denotes the envelope amplitude; $I_{\text{in}}(t-i)$, $Q_{\text{in}}(t-i)$, and $|x(t-i)|$, $(i = 1, 2, ..., M)$ represent the corresponding terms of past samples; and M denotes the depth of the memory.

3.2. BiLSTM Layer

BiLSTM is an extension of LSTM, with its architecture shown in Figure 3. It improves the ability to capture forward and backward information in sequence data by combining two LSTMs: one processing the forward sequence and the other processing the backward sequence. The output of the BiLSTM is obtained by concatenating the two hidden state sequences [27]. The mathematical logic of BiLSTM is as follows.

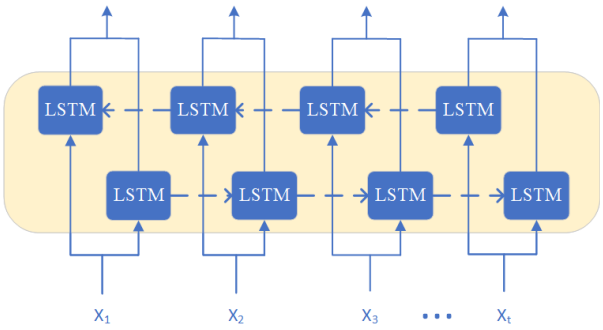

FIGURE 3. BiLSTM architecture diagram.

$$\overrightarrow{y}_{\text{forward}}(n) = \overrightarrow{\text{LSTM}}(x(1), x(2), \ldots, x(n)) \tag{9}$$

$$\overleftarrow{y}_{\text{backward}}(n) = \overleftarrow{\text{LSTM}}(x(n), x(n-1), \ldots, x(1)) \tag{10}$$

$$y_{\text{lstm}}(n) = \overrightarrow{y}_{\text{forward}}(n) + \overleftarrow{y}_{\text{backward}}(n) \tag{11}$$

where $\overrightarrow{y}_{\text{forward}}(n)$ and $\overleftarrow{y}_{\text{backward}}(n)$ represent the results of the LSTM processing the forward and backward sequences, respectively, and $y_{lstm}(n)$ is the final result of the BiLSTM.

3.3. Multi-Head Attention Layer

The self-attention mechanism is a technique used for sequence modelling and feature extraction. The multi-head attention layer integrates multiple parallel computations of single heads to achieve comprehensive feature correlations [28]. The model's expressive power is significantly enhanced by computing attention in parallel. The computational principles of the multi-head attention mechanism are illustrated in Figure 4. In [28], the modeling accuracy of the CNN was improved by incorporating a multi-head attention mechanism. In this paper, a multi-head attention mechanism combined with a BiLSTM network is used to enhance the model's accuracy.

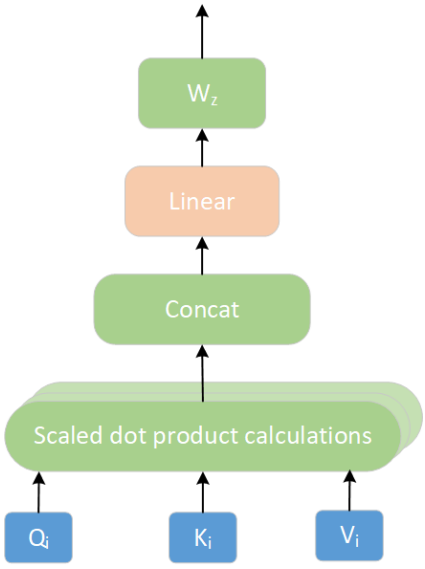

FIGURE 4. Multi-head attention diagram.

First, the query, key, and value matrices are divided into multiple heads. The input data X is transformed through three different linear transformations in each head to generate corresponding Q_i, K_i, and V_i.

$$Q_i = W_i^Q X \quad i = 1, \ldots, 8 \tag{12}$$

$$K_i = W_i^K X \quad i = 1, \ldots, 8 \tag{13}$$

$$V_i = W_i^V X \quad i = 1, \ldots, 8 \tag{14}$$

where W_i^Q, W_i^K, and W_i^V are the weight matrices for the query, key, and value, respectively.

Each attention head independently computes attention to obtain attention outputs, thereby capturing different features in the data. After computing attention across all heads, the outputs

from these heads are concatenated together to generate a new matrix.

$$z_i = \text{softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) V_i, \quad i = 1, \ldots, 8 \tag{15}$$

$$Z_i = \text{Concat}\left(z_1, z_2, \ldots, z_i\right), \quad i = 1, \ldots, 8 \tag{16}$$

where z_i denotes the attention output from a single head; Z_i represents the concatenated matrix of multiple heads' attention outputs; and d_k is the dimension of the key.

Finally, the concatenated matrix is linearly transformed by the weight matrix W_o, mapping it back to the original dimension to obtain the final attention matrix W_z.

$$W_z = Z_i W_o \;\; i = 1, ..., 8 \tag{17}$$

The introduction of attention mechanisms into the network model can significantly enhance the accuracy of power amplifier behavioral modeling, enabling the model to focus on critical temporal dependencies and patterns within the input signal. By dynamically allocating model resources to relevant signal features, the attention mechanism allows for a more precise capture of the inherent nonlinearity and memory effects in broadband power amplifiers. This selective focus not only improves the accuracy of behavioral modeling but also reduces the impact of redundant information, thereby enhancing the model's generalization capability.

3.4. Convolutional Layer

The convolutional layer processes data features using multiple convolutional kernels. Each input unit produces multiple output features by sliding a single convolutional kernel, significantly reducing the number of parameters required for the model and lowering its complexity [29], as shown in Figure 5.

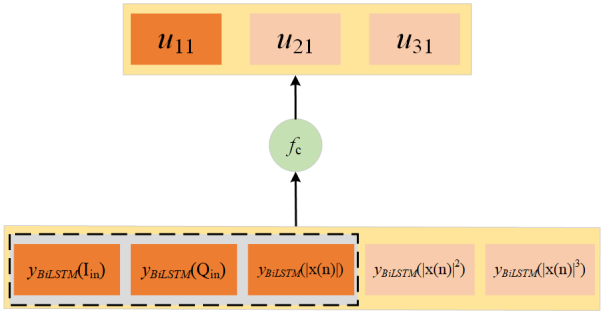

FIGURE 5. Convolutional kernel extraction range example.

The output of the BiLSTM layer, after being processed by the attention mechanism, is fed into the convolutional layer for global feature processing. The convolutional kernels continuously perform convolution operations to learn complex high-level features from simple low-level features, allowing them to capture essential details in the data features. The specific convolution operation is as follows.

$$h_l = y_{\text{BiLSTM}} \otimes w_l = \begin{bmatrix} y_{\text{BiLSTM}}\left(I_{\text{in}}\right) \\ y_{\text{BiLSTM}}\left(Q_{\text{in}}\right) \\ y_{\text{BiLSTM}}\left(|x\left(t\right)|\right) \\ y_{\text{BiLSTM}}\left(|x\left(t\right)|^2\right) \\ y_{\text{BiLSTM}}\left(|x\left(t\right)|^3\right) \end{bmatrix} \otimes w_l \tag{18}$$

where h_l represents the final output of the convolutional kernel, l the number of convolutional kernels, y_{BiLSTM} the output of the BiLSTM layer, and w_l the coefficients of the convolutional kernels.

The convolution process is followed by a nonlinear activation function, which transforms the linear output produced by the convolutional kernels into a nonlinear output, allowing the model to capture and represent more complex features. The activation function of the convolutional kernel is denoted as

$$u_l = f_c\left(h_l + b_l\right) \tag{19}$$

where u_l represents the feature map output of the convolutional kernel, f_c the activation function of the convolutional kernel, and b_l the bias term in the activation function process.

3.5. Fully Connected Layer

The final layer of the model is a fully connected layer, which transforms the data dimensions to produce an output vector with the exact dimensions as the I/Q components. In the fully connected layer, each neuron is connected to all neurons in the previous layer, allowing for the fusion of all features from the preceding layer and enabling higher-level feature representation. A simple linear function is also applied in each hidden unit to facilitate dimensional transformation. The computation formula for the fully connected layer is as follows.

$$y_{out} = \phi\left(w_{\text{out}} \cdot y_{\text{CNN}}\left(n\right) + b_{\text{out}}\right) \tag{20}$$

where w_{out} is the weight matrix, b_{out} the output bias, and y_{CNN} the output of the CNN layer.

4. MODEL ANALYSIS

4.1. Training of BiLSTM-CNN Model

Input the PA sample signals into the BiLSTM-CNN model and train the model. The Adam optimisation algorithm updates the model's parameters and learning rate during training. Network training aims to minimise the error between the actual output data and BiLSTM-CNN model's output. The training continues iteratively until the network converges. This paper defines the Mean Squared Error (MSE) as the cost function. Since the output sample features of the power amplifier consist of I and Q components of the signal, MSE can be expressed as follows:

$$\text{MSE} = \frac{1}{2N}\sum_{n=1}^{N}\left[\left(I_{\text{out}}(n) - \hat{I}_{\text{out}}(n)\right)^2 + \left(Q_{\text{out}}(n) - \hat{Q}_{\text{out}}(n)\right)^2\right] \tag{21}$$

where $I_{\text{out}}(n)$ and $Q_{\text{out}}(n)$ represent the outputs of the BiLSTM-CNN model, and $\hat{I}_{\text{out}}(n)$ and $\hat{Q}_{\text{out}}(n)$ denote the I/Q components of the actual output signal from the PA, with N being the number of samples.

In behavioural modelling of power amplifiers, Normalized Mean Squared Error (NMSE) is commonly used to assess the model's performance [30]. NMSE evaluates the difference between the model's predicted output signal and the actual output signal, normalising this difference. This paper also converts NMSE values into dB format for easier comparison and analysis. The expression for NMSE is as follows.

$$\text{NMSE} = 10\log_{10}\sum_{n=1}^{N}|y(n)-s(n)|^2 \Big/ \sum_{n=1}^{N}|y(n)|^2 \quad (22)$$

where $y(n)$ is the measured output signal, and $s(n)$ is the output signal predicted by the model.

Adam (Adaptive Moment Estimation) is a widely used adaptive learning rate optimisation algorithm in deep learning model training. It dynamically adjusts the learning rate by combining estimates of the mean and variance of the gradients, thus enhancing the stability and efficiency of the optimisation process [31]. In the Adam algorithm, parameters $\beta 1$ and $\beta 2$ control the decay rates of the moving averages of the gradients and their squares, and they are typically set close to 1. This paper's experiments validate the use of $\beta 1 = 0.95$ and $\beta 2 = 0.99$ for these parameters.

The optimisation process of the Adam algorithm begins by initialising parameters, including setting the time step to $t = 0$, model parameters to θ_0, the mean vector s_0, and the variance vector v_0 to zero, and defining the learning rate α. A constant ϵ is used to prevent the variance vector from being zero, and in this paper, it is set to 10^{-8}. At each time step t, the current gradient g_t is first calculated, followed by updating the mean vector s_t and variance vector v_t. To improve the accuracy of these estimates, Adam applies bias correction to the mean and variance vectors before updating the model parameters, using the corrected estimates $\hat{s}_t$ and $\hat{v}_t$. Adam's computation logic is as follows:

$$\theta_{t+1} = \theta_t - \frac{\alpha\hat{v}_t}{\sqrt{\hat{s}_t}+\epsilon} \quad (23)$$

$$s_t = \beta_2 s_{t-1} + (1-\beta_2)g_t^2, \quad \hat{s}_t = \frac{s_t}{1-\beta_2^t} \quad (24)$$

$$v_t = \beta_1 v_{t-1} + (1-\beta_1)g_t, \quad \hat{v}_t = \frac{v_t}{1-\beta_1^t} \quad (25)$$

The trained BiLSTM-CNN model serves as a black-box model for the input-output relationship of the power amplifier (PA), allowing for the direct prediction of the output signal based on the input signal. This model offers a more accurate analysis of the PA's nonlinear characteristics and provides a practical approach to improving PA nonlinearity. The BiLSTM-CNN model becomes a powerful tool for PA behaviour modelling and optimisation by deeply modelling the input-output relationship. The detailed training steps of the BiLSTM-CNN model are shown in Algorithm 1.

Input: I/Q,$|x(n)|$, $|x(n)|^2$, $|x(n)|^3$
Output: I/Q
Definition:
1. Load and preprocess the dataset;
2. Define the loss function of the model;
3. Set the convergence threshold to 1×10^{-8};
4. Set kernel size to 1×3;
5. Set hidden layer size of BiLSTM to 24;
6. Set the number of heads in multi-head attention to 4;
7. Set the number of neurons in the fully connected (FC) layer to 64;

Model Training:
Initialise the optimiser (Adam) and loss function;
for *each epoch* **do**
  for *each batch* **do**
    Perform forward propagation to compute the output;
    Calculate the loss;
    Clear the gradients from the previous step;
    Perform backpropagation to update parameters;
    Optimise model parameters using Adam;
    if *loss* $\leq 1 \times 10^{-8}$ **then**
      Break from training;
    end
  end
end

Algorithm 1: Training of the BiLSTM Model

4.2. Comparison of Different Model Parameters

In this section, we will optimize the model performance by comparing the performances under different parameters. Regarding parameter optimization, the neural network model has a clear advantage over traditional power amplifier behavior models. Traditional amplifier linear models, such as the Volterra model, rely on a set of predefined basis functions to approximate the nonlinear behavior of power amplifiers. As the system's nonlinearity becomes more complex, the computational load increases exponentially. The Volterra model requires manual adjustment of numerous parameters in the formula to accommodate different operating conditions, which is not only time-consuming but also difficult to scale to more dynamic systems. In contrast, neural networks can automatically learn the nonlinearities from the data. When being faced with particularly complex nonlinear systems, only a few hyperparameters need to be adjusted in the network structure, allowing for the optimization of the overall network parameters and achieving good modeling performance. Neural networks offer greater adaptability and can model complex systems more efficiently.

The input features of the BiLSTM-CNN model proposed in this paper include I/Q components of the power amplifier (PA) output signal, as well as combinations of $|x(n)|$, $|x(n)|^2$, and $|x(n)|^3$. These features are used to optimize model performance [32]. Experiments were conducted with the model under different learning rates, and the cost function and NMSE were compared. As shown in Table 1, the model achieves its optimal performance when the learning rate is set to 1×10^{-3}, yielding an MSE of 9.46×10^{-8} and an NMSE of -36.8 dB. Consequently, this paper selects 1×10^{-3} as the learning rate and sets the loss function threshold to 1×10^{-8}.

TABLE 1. Model performance with different learning rates.

Learning Rate	1×10^{-2}	5×10^{-3}	1×10^{-3}	5×10^{-4}
MSE (10^{-7})	2.4	1.65	0.946	1.50
NMSE (dB)	-32.4	-34.4	-36.8	-35.7

In the network model, the size of the hidden layer in the BiLSTM layer determines the model's performance and convergence speed. Five different hidden layer sizes for the BiLSTM layer are set, as shown in Table 2. The data in the table shows that when the hidden layer size is 24, the NMSE value is -38.3 dB, and when the hidden layer size continues to decrease, the NMSE value remains almost unchanged. A hidden layer size of 24 is selected for the BiLSTM layer to achieve better model performance and convergence speed.

TABLE 2. NMSE under different hidden layer sizes.

The size of the hidden layer	size = 48	size = 36	size = 24	size = 12	size = 6
NMSE (dB)	-34.4	-36.4	-38.3	-38.5	-38.9

The attention head count determines the flexibility and expressive power of the model when dealing with the multi-head attention mechanism. This paper analyzes the loss function values and corresponding NMSE performance with different numbers of attention heads using a BiLSTM hidden layer size of 24. The attention head count should be a divisor of the hidden layer size, as shown in Table 3. When the attention head count is 4, the model performance is optimal, with an NMSE value of -38.8 dB. However, further increasing the attention head count adversely affects the prediction accuracy of the test set. Therefore, the attention head count is set to 4 in the model.

TABLE 3. NMSE under different numbers of attention heads.

Attention Head count	2	4	6	8	12
NMSE (dB)	-36.8	-38.8	-38.6	-38.3	-37.6

The quantity of channels in the convolutional layer and the neuron count in the fully connected layer significantly impact the network model's performance. This paper evaluates the NMSE across varying channel counts in the convolutional layer, as illustrated in Figure 6. The figure shows that when the channel count is below 64, the NMSE decreases sharply, suggesting that a lower channel count fails to deliver sufficient modelling performance. Beyond 64 channels, the NMSE stabilises with minimal reduction. Balancing performance and convergence speed, the channel count for the convolutional layer is set to 64.

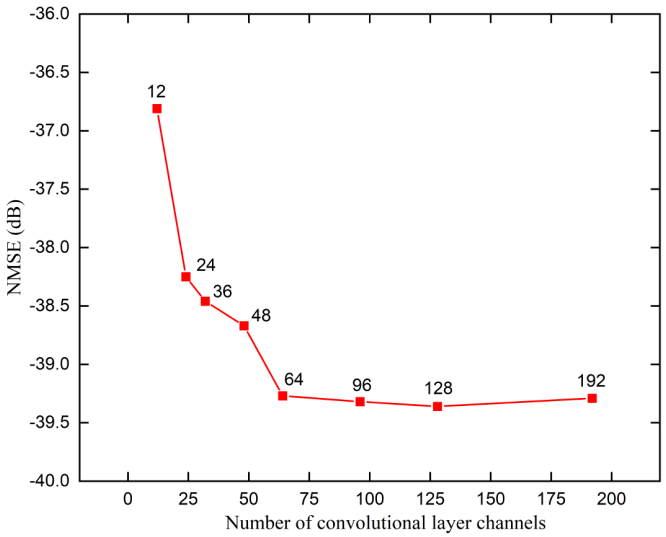

FIGURE 6. NMSE values under different convolutional layer channel counts.

5. DPD EXPANSION

In power amplifier (PA) behavioral modeling, digital predistortion (DPD) is a commonly used linearization technique. The basic principle of DPD is to apply predistortion to the input signal to compensate for the nonlinear effects of the PA, thereby achieving linearization of the PA's output signal. Based on the nonlinear characteristics of the PA, DPD designs an inverse model that distorts the input signal so that, after passing through the PA, the output signal closely approximates a linear signal. The DPD approach effectively enhances PA performance, particularly in communication systems, by reducing signal distortion and spectral regrowth caused by nonlinearities.

In this paper, the DPD method is combined with the proposed BiLSTM-CNN network to better capture the complex nonlinear behavior of the PA. The indirect learning method is employed to design the DPD, where the DPD is treated as the inverse model of the BiLSTM-CNN. The DPD structure is shown in Figure 7. In the training process, the behavioral modeling output of PA is the input of BiLSTM-CNN, and the behavior modeling input of PA is the output of BiLSTM-CNN. Then the trained DPD model is used to calibrate the DPD model on the main signal path, thus achieving PA linearization [33]. The input and output structures of the inverse BiLSTM-CNN model are shown below

$$\begin{aligned}\mathbf{Y}_{\mathrm{DPD}}(n) = [&I_{\mathrm{out}}(n), I_{\mathrm{out}}(n-1), \ldots, I_{\mathrm{out}}(n-M);\\ &Q_{\mathrm{out}}(n), Q_{\mathrm{out}}(n-1), \ldots, Q_{\mathrm{out}}(n-M);\\ &|y(n)|, |y(n-1)|, \ldots, |y(n-M)|;\\ &|y(n)|^2, |y(n-1)|^2, \ldots, |y(n-M)|^2;\\ &|y(n)|^3, |y(n-1)|^3, \ldots, |y(n-M)|^3\,] \end{aligned} \tag{26}$$

$$\mathbf{X}_{\mathrm{DPD}}(n) = \left[I_{\mathrm{in}}(n), Q_{\mathrm{in}}(n)\right]^T \tag{27}$$

where $Y_{\mathrm{DPD}}(n)$ and $X_{\mathrm{DPD}}(n)$ represent the output and input of the DPD; I_{out} and Q_{out} represent the I/Q components of the PA output signal; $|y(n)|$ represents the amplitude information of the PA output signal; and I_{in} and Q_{in} represent the I/Q components of the PA input signal.

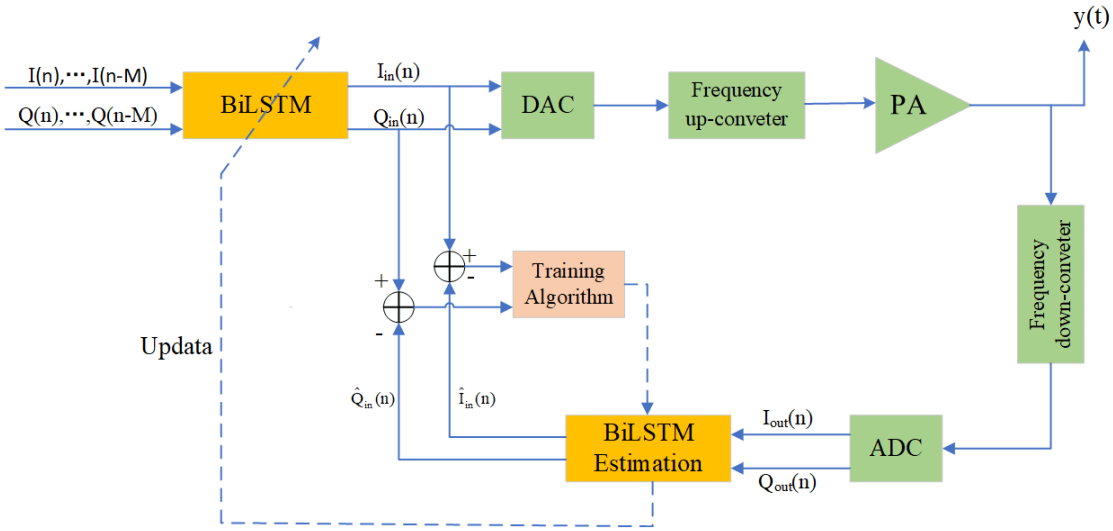

FIGURE 7. Indirect learning structure.

TABLE 4. Model architecture parameters.

Model	Input data	Num of input neurons	Num of neurons in the hidden layer	Activation
RVTDANN	I/Q, $\|x(n)\|$, $\|x(n)\|^2$, $\|x(n)\|^3$	24	64	Tanh
LSTM	I/Q, $\|x(n)\|$, $\|x(n)\|^2$, $\|x(n)\|^3$	12	24	Relu
MLP	I/Q, $\|x(n)\|$, $\|x(n)\|^2$, $\|x(n)\|^3$	16	64	Sigmoid
BiLSTM	I/Q, $\|x(n)\|$, $\|x(n)\|^2$, $\|x(n)\|^3$	10	20	Relu
BiLSTM-CNN	I/Q, $\|x(n)\|$, $\|x(n)\|^2$, $\|x(n)\|^3$	6	12	Relu

6. EXPERIMENTAL RESULTS AND ANALYSIS

6.1. Experimental Preparation

This study is based on a GaN Class-F power amplifier with a gain of approximately 30 dB. An Orthogonal Frequency Division Multiplexing (OFDM) signal with a transmission channel bandwidth of 20 MHz and adjacent channel bandwidth of 180 MHz is generated using MATLAB on a computer. The sampling frequency is set to 30.72×12 MHz, with a total of 70,000 sampling points. The network model is constructed using a PyTorch framework, and the modeling data is divided into training, testing, and validation sets in a $7:1:1$ ratio for model training and testing. The signal generated on the computer is transmitted to a signal generator (SG) and subsequently to the power amplifier (PA). After a 40 dB attenuation, the PA signal is sent to a spectrum analyzer (SA) and finally to the computer, as illustrated in Figure 8. Python is utilized for processing the final transmitted signals for power amplifier behavioral modeling. Python version 3.6 is installed in the Windows environment, and PyTorch version 1.6 is also installed on Windows.

6.2. Model Performance Analysis

This paper compares the BiLSTM-CNN with four typical network models. Table 4 presents the detailed configurations of the network models, including the number of input neurons and hidden layer neurons.

In the proposed BiLSTM-CNN model, the convolutional kernel size of the CNN is 1×3; the sequential length of the data is 5; and both the input and output feature channels of the convolutional layer are 5. The specific formula for calculating the Floating Point Operations (FLOPs) of the network model is as follows. Using the network model parameters in Table 4, the FLOPs of each model are calculated and shown in Table 5. It can be observed that the FLOPs of the BiLSTM-CNN model are only slightly higher than those of the Multilayer Perceptron (MLP), but the modeling accuracy is significantly improved compared to the MLP. Furthermore, the FLOPs of the BiLSTM-CNN are reduced by approximately 50% compared to the BiLSTM, demonstrating the superiority of the proposed

TABLE 5. Model FLOP comparison.

Model	FLOP
RVTDANN	3072
MLP	2048
LSTM	3552
BiLSTM	4960
BiLSTM-CNN	2574

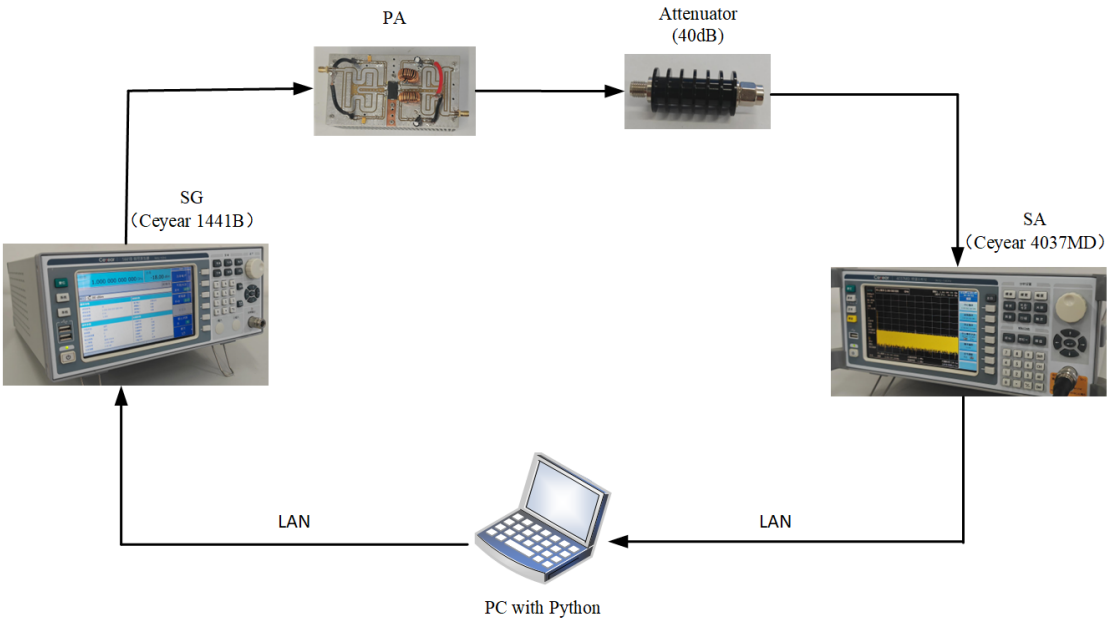

FIGURE 8. Experimental equipment.

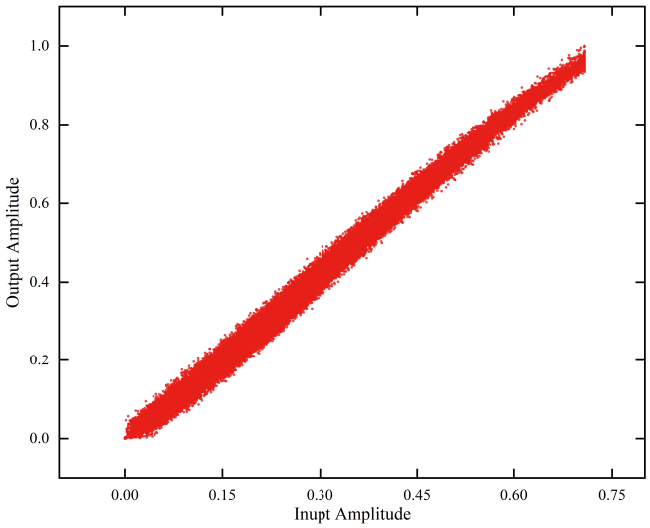

FIGURE 9. AM-AM curve.

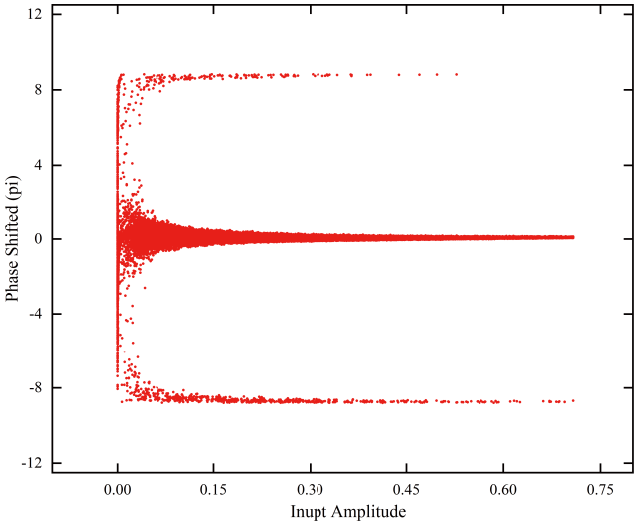

FIGURE 10. AM-PM curve.

model.

$$\text{FLOP}_{\text{FC}} = N_i \times N_h \times 2 \tag{28}$$

$$\text{FLOP}_{\text{LSTM}} = 4 \times (N_i \times N_h + N_h \times N_h + N_h) \tag{29}$$

$$\text{FLOP}_{\text{BiLSTM}} = 2 \times \text{FLOP}_{\text{LSTM}} \tag{30}$$

$$\text{FLOP}_{\text{CNN}} = K \times C_{\text{in}} \times C_{\text{out}} \times H_{\text{out}} \times 2 \tag{31}$$

where N_i represents the number of input neurons; N_h represents the number of output neurons; K is the size of the convolution kernel; C_{in} and C_{out} are the numbers of input and output channels in the convolution layer, respectively; and H_{out} denotes the length of the data sequence.

Figure 9 and Figure 10 show the power amplifier's AM-AM and AM-PM curves without DPD, respectively. The AM-AM curve reflects the power amplifier's memory effect and nonlinearity. The 'E' shape of the AM-PM curve illustrates the power amplifier's phase variation and distortion issues [34].

Figure 11(a) shows the input and output power spectral densities of the PA. Figures 11(b), (c), (d), (e), and (f) compare the PA output power spectral density with the output spectrum of the behavioral modeling models under a 20 MHz OFDM signal. The figure demonstrates that, both in-band and out-of-band, the BiLSTM-CNN model provides the best fit to the actual power amplifier output spectrum. In contrast, other typical network models show higher fitting accuracy in-band but exhibit significant deviations out-of-band. This verifies the superiority of the BiLSTM-CNN in PA behavior modeling.

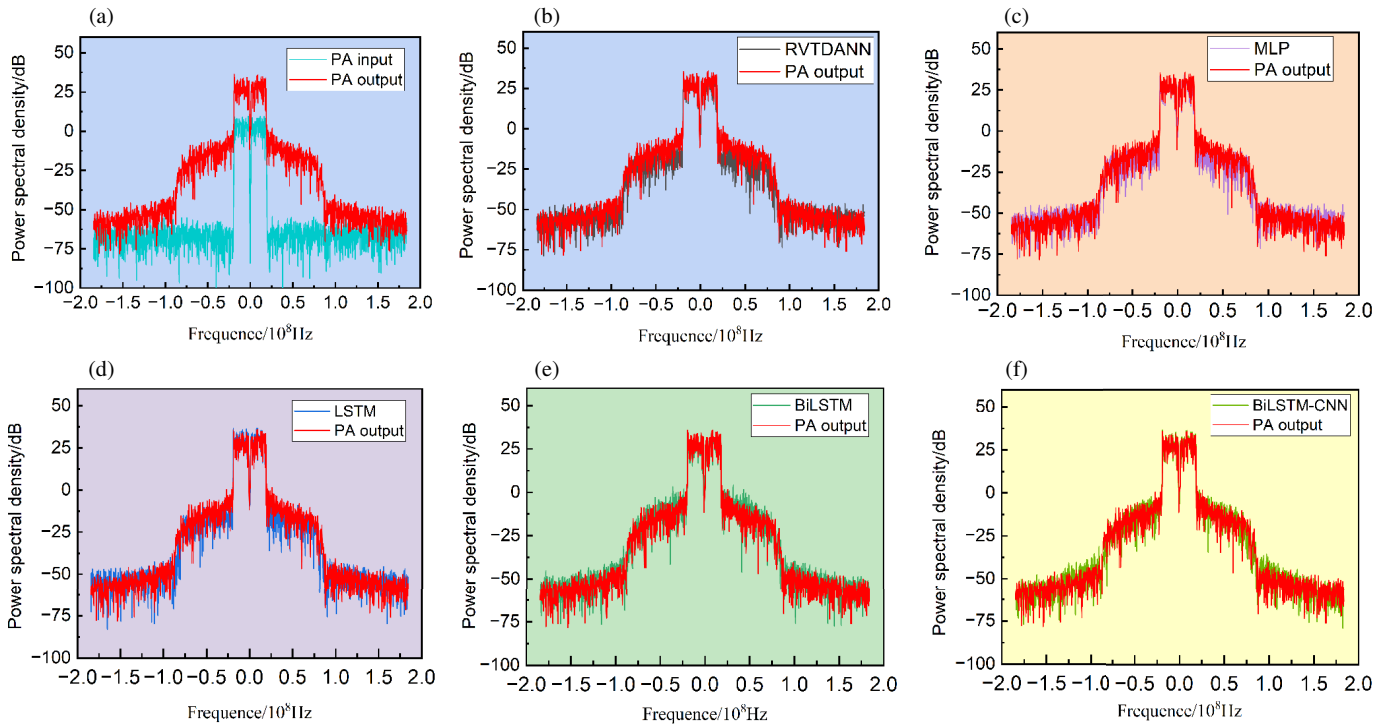

FIGURE 11. Comparison of PA behavioral modeling by different network models.

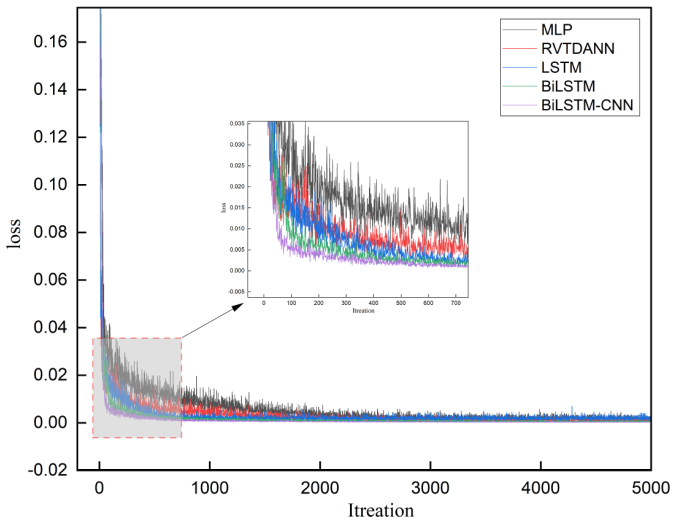

FIGURE 12. Loss function during the training process of different models.

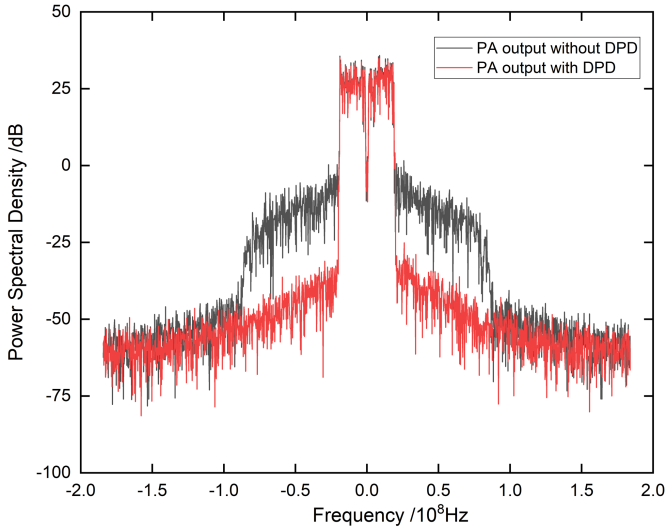

FIGURE 13. PA output spectrum after predistortion processing.

Figure 12 compares the loss function values of different models during the training process. It can be observed that when using the Adam optimiser with the same initial learning rate, the BiLSTM-CNN achieves a faster convergence speed than other network models. It converges from the 80th iteration and reaches a minimum convergence accuracy of 9.8e-8. This indicates that the BiLSTM-CNN is more efficient in the modelling process. The loss value of the BiLSTM-CNN on the test set reaches 1.27e-7, demonstrating that the BiLSTM-CNN model has good generalisation capability.

To validate the superiority of the proposed BiLSTM-CNN model, the final NMSE and training time of each model were compared and analyzed, as shown in Table 6. The experimental results are derived from the model structures in Table 4. It can be seen that the MLP model has the shortest training time of 114 seconds, but its NMSE value only reaches -34.8 dB. The BiLSTM model achieves an NMSE value of -38.4 dB, but the training time is 264 seconds due to its complex structure. The proposed BiLSTM-CNN model can improve NMSE performance by up to 5.5 dB compared to other network models while maintaining a relatively short training time.

TABLE 6. Modeling performance of different methods.

Model	NMSE (dB)	Training Time (s)
RVTDANN	−36.4	187
LSTM	−37.6	202
MLP	−34.8	114
BiLSTM	−38.4	264
BiLSTM-CNN	−40.3	137

Figure 13 shows the output spectra of the power amplifier model both with and without predistortion linearization. The comparison reveals that the power amplifier output spectrum without DPD exhibits significant spectral leakage, a primary manifestation of nonlinear distortion in the frequency domain. After applying DPD, the spectral leakage issue is significantly suppressed, effectively reducing adjacent channel interference caused by nonlinear distortion. Table 7 shows the specific ACPR values of the PA output after DPD improvement. It can be seen that the proposed DPD model can help improve the PA's ACPR by approximately 18 dB, verifying the feasibility of the proposed mod.

TABLE 7. PA ACPR performance with DPD.

Signal Style	ACPR (dBc) (−/ + 20 MHz)
PA Output Without DPD	−26.2/26.5
PA Output With DPD	−44.4/44.6

7. CONCLUSION

This work proposes using a BiLSTM-CNN to model the behavior of wideband PAs to capture their nonlinear and memory effects. The BiLSTM-CNN employs a BiLSTM network to effectively capture the temporal characteristics of PA signals, enhancing model accuracy through a multi-head attention mechanism, while reducing model complexity by sharing parameters within the convolutional neural network. Additionally, a DPD design based on the BiLSTM-CNN model is proposed, adopting the PA inverse model. A 20 MHz OFDM signal is used to evaluate the linearization performance of the BiLSTM-CNN model. Results show that the DPD created by the indirect learning method of BiLSTM-CNN improves the ACPR of the wideband PA by 18 dB, confirming the effectiveness of the BiLSTM-CNN model. Compared to other neural network models, the BiLSTM-CNN model achieves a 5.5 dB improvement in NMSE and demonstrates faster convergence, proving its efficiency.

ACKNOWLEDGEMENT

The second author (Shize Liu) would like to express his sincere gratitude to Prof. Nan Jingchang for providing funding for his thesis, as well as for Prof. Nan's invaluable technical and writing guidance throughout the process of our research. Shize Liu would also like to extend his thanks to the National Natural Science Foundation of China (61971210) for supporting this research on future-oriented wireless reconfigurable intelligent RF modules and neural network modeling.

REFERENCES

[1] Li, M., M. Chen, Y. Yang, Z. Sun, J. Li, Y. Wang, and X. Chen, "Effect of amplifier spontaneous emission noise on performance of space chaotic laser communication systems," *IEEE Journal of Quantum Electronics*, Vol. 57, No. 4, 1–8, Aug. 2021.

[2] Yao, Y., W. Shi, J. Pang, Z. Dai, and M. Li, "Design of a dual-input Doherty power amplifier with selectable output port," *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol. 70, No. 4, 1405–1409, Apr. 2023.

[3] Yu, X., M. Wei, Y. Song, Z. Wang, and B. Chi, "A PAPR-aware dual-mode subgigahertz CMOS power amplifier for short-range wireless communication," *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol. 63, No. 1, 44–48, Jan. 2016.

[4] Latha, Y. M. A. and K. Rawat, "Continuous class F-1 Ku band GaN MMIC power amplifier with an effect of nonlinear output capacitance," *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol. 70, No. 10, 3887–3891, Oct. 2023.

[5] Zhang, T., C. Yu, Y. Liu, S. Li, and B. Tang, "A low-complexity dual-band model for dual-band power amplifiers based on Volterra series," *Progress In Electromagnetics Research Letters*, Vol. 53, 101–106, 2015.

[6] Mokhti, Z. A., J. Lees, C. Cassan, A. Alt, and P. J. Tasker, "The nonlinear drain-source capacitance effect on continuous-mode class-B/J power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, Vol. 67, No. 7, 2741–2747, Jul. 2019.

[7] Wu, J., S. He, J. Peng, P. Hao, and F. You, "Magnitude scaling-based behavioral model for power amplifiers with dynamic power transmission," *IEEE Microwave and Wireless Components Letters*, Vol. 32, No. 5, 463–466, May 2022.

[8] Abedi, F., H. S. Fahama, A. Hamzah, and F. Mohammed, "Adaptive filter predistorter for memory Salehm model," in *2022 5th International Conference on ngineering Technology and Its Applications (IICETA)*, 338–342, Al-Najaf, Iraq, 2022.

[9] Maik, G., "Nonlinear RF impairments modeling with OFDM symbol-based parallel Wiener-Hammerstein structure," *Signal Processing*, Vol. 224, 109587, 2024.

[10] Liu, W., W. Na, F. Feng, L. Zhu, and Q. Lin, "A Wiener-type dynamic neural network approach to the modeling of nonlinear microwave devices and its applications," in *2020 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)*, 1–3, Hangzhou, China, 2020.

[11] Janjanam, L., S. K. Saha, R. Kar, and D. Mandal, "Volterra filter modelling of non-linear system using artificial electric field algorithm assisted Kalman filter and its experimental evaluation," *ISA Transactions*, Vol. 125, 614–630, 2022.

[12] Du, H., F. Wan, V. Mordachev, E. Sinkevich, X. Chen, G. Fontgalland, D.-T. Do, S. Ngoho, and B. Ravelo, "Nonlinear testing-based EMI characterization of wireless communication transmitter with microwave power amplifier," *Progress In Electromagnetics Research C*, Vol. 147, 27–37, 2024.

[13] Xu, G., H. Yu, C. Hua, and T. Liu, "Chebyshev polynomial-LSTM model for 5G millimeter-wave power amplifier linearization," *IEEE Microwave and Wireless Components Letters*, Vol. 32, No. 6, 611–614, 2022.

[14] Zhao, J., C. Yu, J. Yu, Y. Liu, and S. Li, "A robust augmented combination of digital predistortion and crest factor reduction for RF power amplifiers," *Progress In Electromagnetics Research C*, Vol. 57, 181–191, 2015.

[15] Truong, H. T., B. P. Ta, Q. A. Le, D. M. Nguyen, C. T. Le, H. X. Nguyen, H. T. Do, H. T. Nguyen, and K. P. Tran, "Light-weight federated learning-based anomaly detection for time-series data

in industrial control systems," *Computers in Industry*, Vol. 140, 103692, 2022.

[16] Sun, J., W. Shi, Z. Yang, J. Yang, and G. Gui, "Behavioral modeling and linearization of wideband RF power amplifiers using BiLSTM networks for 5G wireless systems," *IEEE Transactions on Vehicular Technology*, Vol. 68, No. 11, 10 348–10 356, 2019.

[17] Wang, H., Y. Zhang, J. Liang, and L. Liu, "DAFA-BiLSTM: Deep autoregression feature augmented bidirectional LSTM network for time series prediction," *Neural Networks*, Vol. 157, 240–256, 2023.

[18] Xia, T., Y. Song, Y. Zheng, E. Pan, and L. Xi, "An ensemble framework based on convolutional bi-directional LSTM with multiple time windows for remaining useful life estimation," *Computers in Industry*, Vol. 115, 103182, 2020.

[19] Wei, L.-Y., F.-C. Chen, and Z.-B. Zhang, "Design of dual-band power amplifier based on microstrip coupled-line bandstop filter," *AEU — International Journal of Electronics and Communications*, Vol. 185, 155450, 2024.

[20] Zhong, Y., Z. Dai, and S. Hu, "Asymmetric Doherty power amplifier design considering the effects of peaking power amplifier early conduction," *AEU — International Journal of Electronics and Communications*, Vol. 179, 155307, 2024.

[21] Wu, X. and L. Chen, "2-D orthogonal polynomial model for concurrent dual-band digital predistortion based on complex Gaussian assumption," *AEU — International Journal of Electronics and Communications*, Vol. 135, 153704, 2021.

[22] Böhler, J., J. Huber, J. Wurz, M. Stransky, N. Uvaidov, S. Srdic, and J. W. Kolar, "Ultra-high-bandwidth power amplifiers: A technology overview and future prospects," *IEEE Access*, Vol. 10, 54 613–54 633, 2022.

[23] Qu, K., J. Xu, Z. Han, and S. Xu, "Maximum relevance minimum redundancy-based feature selection using rough mutual information in adaptive neighborhood rough sets," *Applied Intelligence*, Vol. 53, No. 14, 17 727–17 746, 2023.

[24] Li, T., M. Hua, and X. Wu, "A hybrid CNN-LSTM model for forecasting particulate matter (PM2.5)," *IEEE Access*, Vol. 8, 26 933–26 940, 2020.

[25] Slawiñski, E., F. Rossomando, F. A. Chicaiza, J. Moreno-Valenzuela, and V. Mut, "LSTM network in bilateral teleoperation of a skid-steering robot," *Neurocomputing*, Vol. 602, No. 2, 128248, 2024.

[26] Zhang, S., X. Hu, Z. Liu, L. Sun, K. Han, W. Wang, and F. M. Ghannouchi, "Deep neural network behavioral modeling based on transfer learning for broadband wireless power amplifier," *IEEE Microwave and Wireless Components Letters*, Vol. 31, No. 7, 917–920, 2021.

[27] Mustaqeem, M., M. Sajjad, and S. Kwon, "Clustering-based speech emotion recognition by incorporating learned features and deep BiLSTM," *IEEE Access*, Vol. 8, 79 861–79 875, 2020.

[28] Xiao, M., B. Yang, S. Wang, Z. Zhang, X. Tang, and L. Kang, "A feature fusion enhanced multiscale CNN with attention mechanism for spot-welding surface appearance recognition," *Computers in Industry*, Vol. 135, 103583, 2022.

[29] He, K., G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 42, No. 2, 386–397, Feb. 2020.

[30] Zhang, D., H. Lv, S. Yan, Y. Hu, Q. Zhang, C. Han, R. Zhao, and Y. Zhang, "Multi-objective neural network modeling and applications to microwave power amplifiers," *Microelectronics Journal*, Vol. 149, 106244, 2024.

[31] Song, H., H. Wang, J. Wu, and J. Yang, "An efficient ADAM-type algorithm with finite elements discretization technique for random elliptic optimal control problems," *Journal of Computational and Applied Mathematics*, Vol. 454, 116199, 2024.

[32] Alkhoder, A., A. Assimi, and M. Alhariri, "Frequency domain model of power amplifier for OFDM signals," *AEU — International Journal of Electronics and Communications*, Vol. 110, 152867, 2019.

[33] Schuartz, L., A. T. Hara, A. A. Mariano, B. Leite, and E. G. Lima, "Comparison between direct and indirect learnings for the digital pre-distortion of concurrent dual-band power amplifiers," in *Proceedings of the 32nd Symposium on Integrated Circuits and Systems Design (SBCCI'19)*, 1–5, New York, NY, USA, 2019.

[34] Devi, R. V. S., K. R. Bindu, and D. G. Kurup, "Behavioral modeling and digital predistortion of RF power amplifiers based on time-delay kernel ridge regression," *AEU — International Journal of Electronics and Communications*, Vol. 152, 154239, 2022.