

Hyperparameter Tuned Bidirectional Gated Recurrent Neural Network for Weather Forecasting

S. Manikandan^{1,*} and B. Nagaraj²

¹Department of Information Technology, Karpagam College of Engineering, Coimbatore, 641032, India

²Principal Rathiniam Technical Campus, Coimbatore, 641021, India

*Corresponding Author: S. Manikandan. Email: manikandanms@kce.ac.in

Received: 06 September 2021; Accepted: 10 November 2021

Abstract: Weather forecasting is primarily related to the prediction of weather conditions that becomes highly important in diverse applications like drought discovery, severe weather forecast, climate monitoring, agriculture, aviation, telecommunication, etc. Data-driven computer modelling with Artificial Neural Networks (ANN) can be used to solve non-linear problems. Presently, Deep Learning (DL) based weather forecasting models can be designed to accomplish reasonable predictive performance. In this aspect, this study presents a Hyper Parameter Tuned Bidirectional Gated Recurrent Neural Network (HPT-BiGRNN) technique for weather forecasting. The HPT-BiGRNN technique aims to utilize the past weather data for training the BiGRNN model and achieve the effective forecasts with minimum time duration. The BiGRNN is an enhanced version of Gated Recurrent Unit (GRU) that follows the process of passing input via forward and backward neural network and the outputs are linked to the identical output layer. The BiGRNN technique includes several hyper-parameters and hence, the hyperparameter optimization process takes place using Bird Mating Optimizer (BMO). The design of BMO algorithm for hyperparameter optimization of the BiGRNN, particularly for weather forecast shows the novelty of the work. The BMO algorithm is used to set hyperparameters such as momentum, learning rate, batch size and weight decay. The experimental result the HPT-BiGRNN approach has resulted in a lower RMSE of 0.173 whereas the Fuzzy-GP, Fuzzy-SC, MLP-ANN and RBF-ANN methods have gained an increased RMSE of 0.218, 0.216, 0.202 and 0.245 respectively.

Keywords: Weather forecast; time series prediction; deep learning; parameter tuning; bigrnn model; metaheuristics

1 Introduction

Weather conditions like wind, temperature and humidity profoundly affects human livelihood. Weather prediction offers analytical support to the smart transportations like air visibility analysis, traffic flow prediction, etc. [1]. As we know, traffic congestions and accidents are highly possible in weather conditions like heavy fog, heavy rain and so on. Timely and accurate weather prediction can save us from



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

weather disasters [2]. Weather prediction represents the scientific method of forecasting the state of atmosphere depending on time and certain locations [3]. Numerical Weather Prediction (NWP) utilizes computer algorithm for prediction based on the present weather condition by resolving a huge system of non-linear arithmetical equations on certain arithmetical model. In recent years, Various NWP methods have been proposed namely Weather Research and Forecasting (WRF) models. Based on [4], data driven computer modelling system could be utilized for reducing the computation power of NWP models. Especially, Artificial Neural Networks (ANN) could be employed for their adaptive nature and learning abilities based on previous knowledge [5]. Such feature makes the ANN technique most attractive in application domain to resolve highly non-linear phenomena.

The Artificial Neural Network (ANN) is one of the robust data modelling tools which can represent and capture complicated relations between outputs and inputs. It is proposed by the inspiration of executing artificial system which can implement intelligent tasks like human brain. Generally, ANN is capable of approximating any non-linear functions [6]. The Deep Neural Network (DNN) is a type of ANN made up of multilayer architectures that can recreate the raw datasets from the original feature space to a learned feature space. In other word, they could “learn” features through Neural Network (NN) rather than choosing features automatically and attain better generalization and higher accuracy through the learned feature. Deep Learning (DL) has accomplished a promising result in several fields like speech recognition, computer vision, scientific fields in physics, chemistry, bio-information and natural linguistic programming [7]. Recently, DL models are used for time sequence problems, where the correlation of the features is apparent, however, it is difficult to determine. Particularly, if the system behaviour is controlled by temporal/spatial contexts (like weather scheme), conventional Machine Learning (ML) methods are not suitable unlike DL methods, that are capable of manually extracting spatial and temporal features for better understanding of the system [8]. When the correlations are analysed and the feature is represented correctly, the prediction performance would be enhanced that is suitable to analyse the features of time sequence. According to this, several researchers have used DL for predicting weather conditions and few traditional complexities in forecasting would be tackled by a data driven method.

As the DL model performs well on weather forecasting problems, this study develops a Hyperparameter Tuned Bidirectional Gated Recurrent Neural Network (HPT-BiGRNN) technique that primarily executes the prediction process using BiGRNN technique. Secondly, the hyperparameters in the BiGRNN technique are optimally adjusted by the BMO algorithm. The design of BMO algorithm based on the mating behaviour of the birds helps to optimally select the hyperparameters of the BiGRNN technique and thereby boosting the overall predictive performance. A wide range of simulation analysis take place on the benchmark dataset and the experimental results are investigated under various aspects.

2 Prior Weather Forecasting Approaches

This section provides an extensive review of weather forecasting approaches available in the literature. Castro et al. [9] proposed a STConvS2S network, a DL framework constructed to learn spatial-temporal data dependency with convolution layer. The presented framework resolves the two main limitations of convolution network for predicting the sequence using the past data such as (1) it requires the length of output and input series to be equivalent and (2) they violate the temporal orders at the time of learning procedure. Sundaram [10] evaluated and developed the short-term weather prediction model with LSTM and calculate the performance of the traditional WRF-NWP models. The presented model consists of stacked LSTM layer which employs surface weather parameters with time period for weather forecasting. These models are tested with distinct numbers of LSTM layer, optimizer and learning rate that are optimized to an efficient short-term weather prediction.

Salman et al. [11] constructed an adaptive and robust statistical model to forecast univariate weather variables at Indonesian airport areas to examine the effects of intermediate weather variables associated with the prediction accuracy through single layer LSTM models and multilayer LSTM models. The presented method is an extension of LSTM models that includes intermediate variable signals to LSTM memory blocks. An approach based on the DNN model [12] for learning higher resolution demonstration from lower resolution prediction of weather as a real time application. The study result shows the outstanding performances and the approach is lightweight enough to run-on moderate computer system.

Rasp et al. [13] proposed a mixed model that uses a subset of original weather trajectory integrated with postprocessing steps of DNN model that represent nonlinear relationship not considered in the present postprocessing/mathematical model. A data driven methods with an efficient data fusion method for learning from the past data that integrates previous knowledge from NWP [14]. They consider the weather forecast problems as an end-to-end DL problem and resolve them by presenting a new NLE loss function. A significant advantage of the presented model is that it concurrently implements uncertainty quantification and single value prediction that represents the DUQ model.

Hewage et al. [15] proposed a lightweight data driven weather prediction models by examining and relating the performance of temporal modelling approach of LSTM and TCN with a dynamic ensemble method, traditional ML approaches, statistical forecasting approaches and the conventional WRF-NWP models. Singh et al. [16] used the UNET framework of DCNN using residual learning to learn global data driven model of precipitation. It is trained over reanalysis datasets proposed on the cubed sphere prediction to minimize error because of its spherical distortions. The result is related to the operational dynamical models by the India Meteorological Departments. The theoretical DL based models show doubling of grid points and area average skills evaluated in Pearson correlation coefficient in relation to operational systems. BiGRNN is a type of artificial neural network that is particularly well suited for evaluating and processing time sequence data. This is in contrast to standard neural networks, which are based on the weight connection between the layers. In order to maintain information from the previous instant, BiGRNN uses hidden layers, and the output is impacted by the present states and memories from the past.

An ensemble predictive scheme with DLWP models [17] that recurrently predict key atmospheric variables using 6 h' time resolution. These models use CNN on cubed sphere grids for producing global predictions. This method is computational effective, as it requires only 3 min on individual GPU for producing 320 member sets of 6 weeks forecast at 1.4° resolution. Ensemble spread is mainly built by randomizing the CNN training procedure for creating a set of 32 DLWP methods. In Peng et al. [18], a novel NN predictive model named EALSTM-QR has been proposed to predict the wind power considering the input of NWP and the DL methods. In this method, there are four major steps namely Attention, Encoder, bi-LSTM and QR. The combinational input contains the past wind power data and the extracted feature attained from the NWP at Attention and Encoder level. The bi-LSTM is used for generating the wind power time sequence predictive results.

3 The Proposed Weather Forecasting Model

The new BiGRNN technique is designed to effectively estimate the weather conditions. The BiGRNN technique encompasses a two-stage process namely BiGRNN based prediction process and BMO based hyperparameter tuning process. Fig. 1 demonstrates the process of proposed HPT-BiGRNN model.

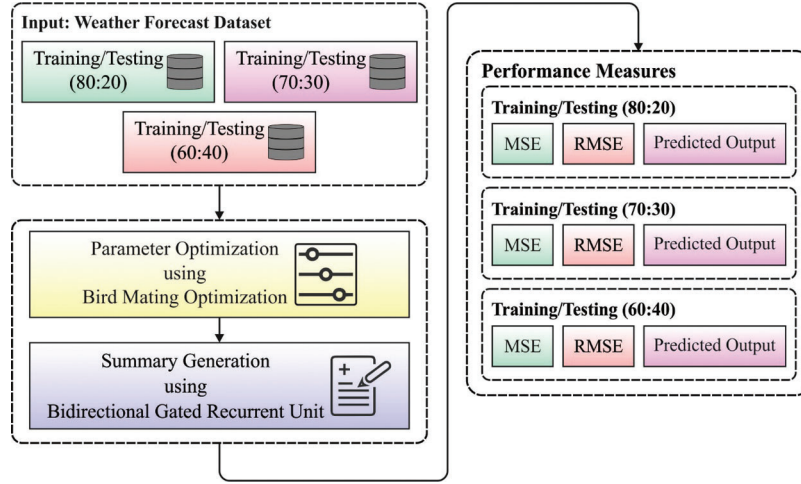


Figure 1: The process of HPT-BiGRNN model

3.1 Weather Forecasting Using BiGRNN Model

At the initial weather forecasting process, the BiGRNN technique receives the previous weather data as input and estimates the future weather condition. The RNNs capability of forecasting its output is well suitable for sequential data. The RNN has a hidden state h , that estimates the input value x and their preceding state. Consider the input series x_1, x_2, \dots, x_n , the RNN is signified by Eq. (1) and Eq. (2) for evaluating the output y_t . Here, h_t refers to the hidden state at time step t ; W_{hh} , W_{hx} and W_{yh} are the weights of the hidden-to-hidden layer connections, input connections and hidden-to-output connections respectively. b_h represents the bias to the hidden state and b_y implies the bias to the output state. f represents the activation function implemented on the hidden and output nodes throughout the network.

$$h_t = f(W_{hh}h_{t-1} + W_{hx}x_t + b_h) \quad (1)$$

$$y_t = f(W_{yh}h_t + b_y) \quad (2)$$

A major difference between RNNs and GRU is that GRU can capture long term dependencies and address the vanishing gradient issue frequently encountered if trained RNN [19]. During this unit the hidden state h_t of GRU is computed as follows.

$$h_t = u \otimes \tilde{h}_t + (1 - u) \otimes h_{t-1} \quad (3)$$

where u refers to the update gate that checks whether h_t is upgraded or not and \otimes is an element-wise multiplication. The upgrade gate u is upgraded by implementing the sigmoid logistic function σ and it is given by,

$$u_t = \sigma(W_{uh}h_{t-1} + W_{ux}x_t + b_u) \quad (4)$$

The candidate cell is updated using the hyperbolic tangent as follows.

$$\tilde{h}_t = \tanh(W_{hx}x_t + W_{hh}(r \otimes h_{t-1}) + b_h) \quad (5)$$

where r represents the reset gate for computing the relevance of h_{t-1} to h_t . The reset gate r is estimated in a subsequent manner by,

$$r_t = \sigma(W_{rh}h_{t-1} + W_{rx}x_t + b_r) \quad (6)$$

The BiGRNN has improved 2 layer framework that gives the resultant layer with contextual data of input at all moments. The idea of Bi-GRNN is that the input order is passed by forward NN and backward NN and later, the output of these two are related with similar resultant layer. Fig. 2 shows the framework of Bi-GRU model. During the Bi-GRNN of all layers, the forward layer computes the outcome of hidden layer at every time from forward to backward and the backward layer computes the outcome of hidden layer at every time from backward to forward [20].

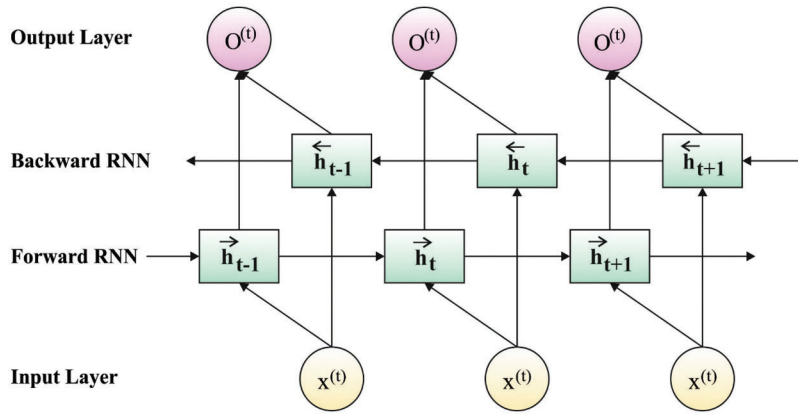


Figure 2: Structure of Bi-GRU model

The resultant layer superimposes and normalizes the output of forward and backward layers at all moments as follows.

$$\vec{h}_t^1 = f(w_{xh^1}^{\rightarrow}x_t + w_{h^1h^1}^{\rightarrow}\vec{h}_{t-1}^1 + b_{h^1}^{\rightarrow}) \quad (7)$$

$$\overleftarrow{h}_t^1 = f(w_{xh^1}^{\leftarrow}x_t + w_{h^1h^1}^{\leftarrow}\overleftarrow{h}_{t-1}^1 + b_{h^1}^{\leftarrow}) \quad (8)$$

$$\vec{h}_t^2 = f(w_{h^1h^2}^{\rightarrow}\vec{h}_t^1 + w_{h^2h^2}^{\rightarrow}\vec{h}_{t-1}^2 + b_{h^2}^{\rightarrow}) \quad (9)$$

$$\overleftarrow{h}_t^2 = f(w_{h^1h^2}^{\leftarrow}\overleftarrow{h}_t^1 + w_{h^2h^2}^{\leftarrow}\overleftarrow{h}_{t-1}^2 + b_{h^2}^{\leftarrow}) \quad (10)$$

$$y_t = g(w_{h^2y}^{\rightarrow}\vec{h}_t^2 + w_{h^2y}^{\leftarrow}\overleftarrow{h}_t^2 + b_y) \quad (11)$$

where $\vec{h}_t^1 \in R^H$ and $\overleftarrow{h}_t^1 \in R^H$ are the resultant vectors of hidden layers of the forward layer from 1st and 2nd layers of Bi-GRNN at time t , H refers to the number of units from the GRU cells, $\vec{h}_t^1 \in R^H$ and $\overleftarrow{h}_t^1 \in R^H$ are the resultant vectors of hidden layers of the backward layer from 1st and 2nd layers of the Bi-GRNN at time t , $y_t \in R^T$ is the score of matching word on all labels at time t , T refers to the count of tags, x_t signifies the NN input at time t , f demonstrates the GRNN modelling, g refers to the activation functions, where $g(x)_i = \frac{e^{x_i}}{\sum_{k=1}^n e^{x_k}}$. w and b represents the weight matrices.

3.2 Hyperparameter Optimization Using BMO Algorithm

The BiGRNN technique has several hyperparameters and the BMO algorithm is used to optimally select the momentum, learning rate, batch size and weight decay. The BMO is a population based stochastic search method developed for addressing continual optimization problem [21]. The behaviour of this algorithm depends on the mating strategy of bird species during mating season. The mating procedure of birds include the usage of three primary operators for producing a novel generation namely mutation, two parents mating and multi parents mating. In Two parents mating, they mate with one another for breeding only one novel breed, while in multi parents mating, a parent mate with at least two other parents for breeding one novel breed. Alternatively, Mutation is a method where the female parents produce novel broods without male parent by changing their own genes. On the traditional BMO algorithm, the assumptions that every bird in the society could modify its own strategy, once a novel generation is upgraded. Always, the birds with maximum fitness value are selected as polyandrous and parthenogenetic, where the birds with poor fitness value are selected as promiscuous. The residual bird in the society is considered as polygynous/ monogamous bird. Parthenogenesis is a kind of mating process where the females might generate a brood without mating with males. In this way, all females try to generate their broods by modifying her genes with predetermined rates. All females in the parthenogenetic groups produce a brood as follows.

```

for i = 1:n
    if  $r_1 > mcf_p$ 
         $x_{brood}(i) = x(i) + \mu \times (r_2 - r_3) \times x(i);$ 
    else
         $x_{brood}(i) = x(i)$ 
    end
end
end

```

whereas $x(i)$ represents the i^{th} birds, x_{brood} indicates the resulting broods, n denotes the dimensional problems (amount of genes), mcf_p signifies the mutation control factors of parthenogenesis, r_1 denotes the arbitrary values in the range of zero and one, μ denotes the step size. Monogamy is the two parents mating type where a male bird tends to mate with only one female bird. All male birds evaluate the qualities of female birds through a probabilistic method for selecting its mate. Females with good genes have a high possibility of being designated. Eq. (13) illustrate the procedure of generating novel broods from 2 designated parents [22].

```

 $\vec{x}_i = \vec{x} + w \times \vec{r} \cdot (\vec{x}^i - \vec{x})$ 
 $c = a$  arbitrary value in the range of 1 to  $n$ 
if  $r_1 > mcf$ 
     $x_{brood}(c) = l(c) - r_2 \times (l(c) - u(c));$ 

```

End

whereas w represents a time differing weights for adjusting the designated female birds, r indicates a $1 \times d$ vector where all the elements are randomly distributed numbers in the range of zero and one *also* this arbitrary vectors influence the equivalent elements of $(\vec{x}^i - \vec{x})$, n represents the dimensional problems, mfc is the mutation control factor lies between zero and one, u & l represents the upper bound and lower bound of the element respectively. From Eq. (13), it is clear that, all males search for a worthy female for producing good quality broods through his genes to generate novel genes. Next, the males try to enhance their generated broods by mutating another broods' gene with $1 - mcf$.

Polygamy is a multiparent mating procedure where males try to generate broods by mating with more than one female bird. The advantage of multi mating procedure is to generate broods with good genes.

Naturally, a polygynous bird mate with many female birds producing several broods. After choosing the female birds through the selection process, all males mate with their designated female birds. The resulting broods are generated as follows.

$$\vec{x}_i = \vec{x} + w \times \sum_{j=1}^{n_i} \vec{r}_j \cdot (\vec{x}_j - \vec{x})$$

$C = a$ arbitrary value in the range of 1 to n

if $r_1 > mcf$

$$x_{brood}(c) = l(c) - r \times (l(c) - u(c));$$

End

Let n_i be the number of selected females and x_j^i denote the j th selected birds. In BMO, promiscuous birds generate novel broods based on Eq. (13). Polyandry is a multi-parent mating system where females search for mating with several male birds. The female bird performs a selection process for selecting males. Later, all females mate with her designated males. In the BMO, polyandrous birds generate novel broods based on Eq. (14). The process of BMO is given as follows.

Step 1 (Parameter Initiation): Initiate the succeeding BMO parameter

Step 2 (Society Initiation): Arbitrarily initiate a group of possible solutions and include them in society. All the solutions are considered as birds and it is stated by a vector, using n length.

Step 3 (Society Calculation): Evaluate the quality of birds through a calculation process.

Step 4 (Ranking): Sort the birds in the society according to their quality.

Step 5 (Classification): split the society into five clusters of birds namely promiscuous, monogamous, polyandrous, parthenogenetic and polygynous. The polyandrous and parthenogenetic groups are considered as a female bird and the rest of the groups are considered as a male bird. The percentage of all the groups is determined at the parameters' initiation step.

Step 6 (Breeding): all the birds produce novel broods in their own pattern.

Step 7 (Substitution): When the quality of broods is higher than the quality of birds in the society, the brood replaces the bird or else, the birds remain in the society and the broods are abandoned.

Step 8 (End Criteria): Reiterate the steps 4 to 7 until a predefined NG is carried out.

Step 9 (Report the optimal): choose an optimal quality of birds in the society as an optimal solution.

Fig. 3 illustrates the flowchart of BMO technique.

4 Performance Validation

This section examines the performance of the HPT-BiGRNN technique on the weather forecasting process. The results are examined under varying sizes of training/testing data. Fig. 4 investigates the forecasting results of the HPT-BiGRNN technique with training/testing data of 80:20 and the results proved that the HPT-BiGRNN technique has attained a predicted value which is highly closer to the actual value. For instance, under 1 h, the HPT-BiGRNN technique has offered a predicted value of 7.6700 whereas the actual value is 7.6400. In addition, under 5 h, the HPT-BiGRNN method has obtained a predicted value of 7.6800 whereas the actual value is 7.6400. Also, under 10 h, the HPT-BiGRNN method has attained a predicted value of 8.8200 whereas the actual value is 8.8600. Additionally, under 15 h, the HPT-BiGRNN approach has offered a predicted value of 9.1600 whereas the actual value is 9.1000. Eventually, under 24 h, the HPT-BiGRNN method has obtained a predicted value of 7.5600 whereas the actual value is 7.5900.

Fig. 5 examines the forecasting outcomes of the HPT-BiGRNN approach under varying training/testing data of 70:30. The results depicted that the HPT-BiGRNN method has obtained a predicted value which is extremely closer to the actual value. For instance, under 1 h, the HPT-BiGRNN method has obtained a

predicted value of 7.5600 whereas the actual value is 7.6400. Besides, under 5 h, the HPT-BiGRNN technique has a predicted value of 7.5500 whereas the actual value is 7.6400. In the meantime, under 10 h, the HPT-BiGRNN manner has offered a predicted value of 8.9600 whereas the actual value is 8.8600. Eventually, under 24 h, the HPT-BiGRNN method has offered a predicted value of 7.6600 whereas the actual value is 7.5900.

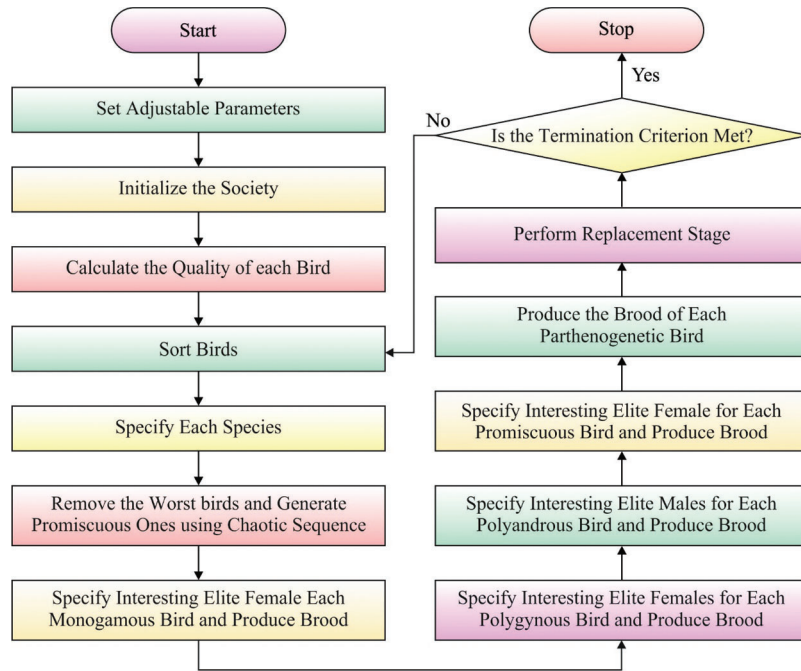


Figure 3: Flowchart of BMO algorithm

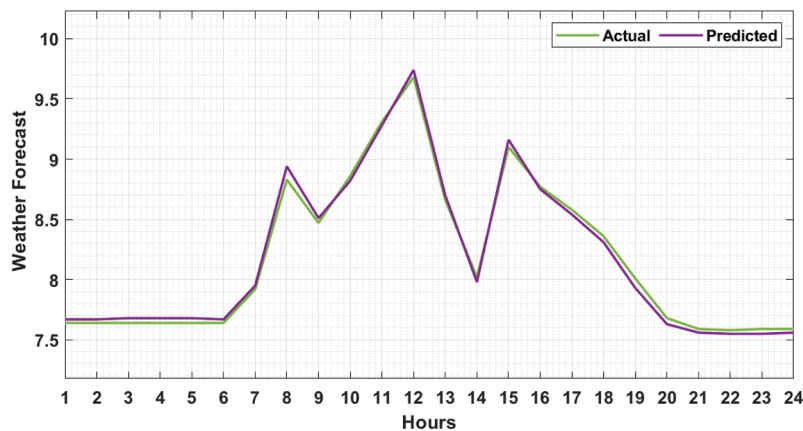


Figure 4: Weather forecast analysis of HPT-BiGRNN model on training/testing data (80:20)

Fig. 6 explores the forecasting results of the HPT-BiGRNN method under varying training/testing data of 60:40. The outcomes exhibited that the HPT-BiGRNN technique has reached a predicted value that is highly closer to the actual value. For sample, under 1 h, the HPT-BiGRNN technique has obtained a predicted value of 7.7500 whereas the actual value is 7.6400. In line with, under 5 h, the HPT-BiGRNN

technique has offered a predicted value of 7.7500 whereas the actual value is 7.6400. Followed by, under 10 h, the HPT-BiGRNN model has offered a predicted value of 9.0200 whereas the actual value is 8.8600. Eventually, under 24 h, the HPT-BiGRNN algorithm has a predicted value of 7.7700 whereas the actual value is 7.5900.

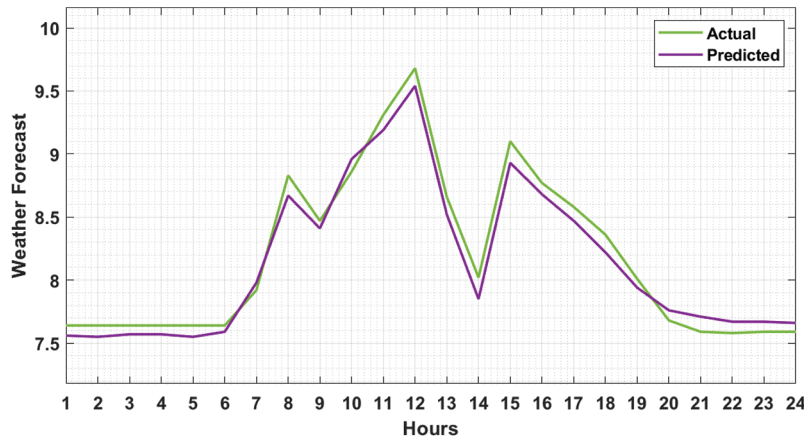


Figure 5: Weather forecast analysis of HPT-BiGRNN model on training/testing data (70:30)

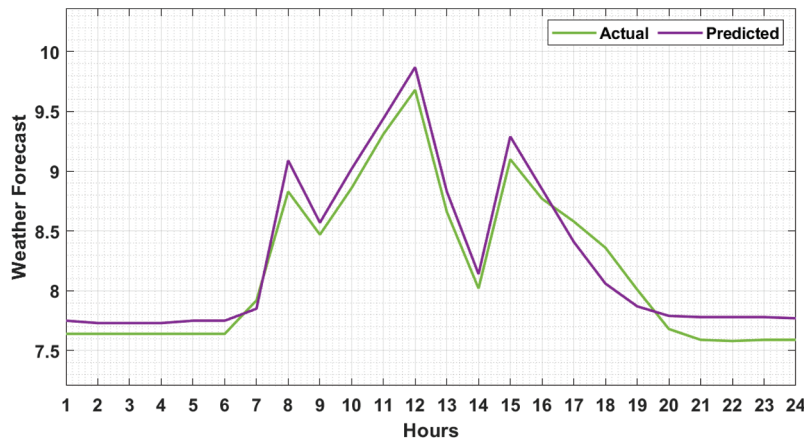
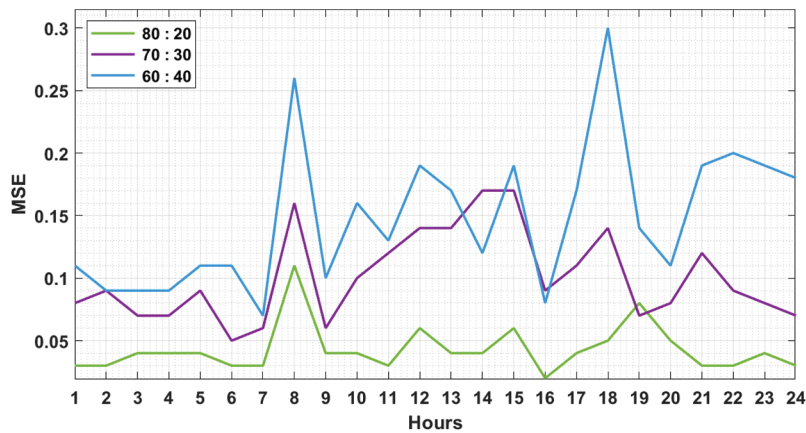


Figure 6: Weather forecast analysis of HPT-BiGRNN model on training/testing data (60:40)

Tab. 1 and Fig. 7 examines the performance of the HPT-BiGRNN technique in terms of MSE under varying training/testing data. The figure shows that the HPT-BiGRNN technique has resulted in effective outcomes with minimal MSE values under varying duration and training/testing data size. For instance, with 1 h, the HPT-BiGRNN technique has attained an MSE of 0.030, 0.080 and 0.110 under the training/testing data of 80:20, 70:30 and 60:40 respectively. Furthermore, with 5 h, the HPT-BiGRNN approach has reached an MSE of 0.040, 0.090 and 0.110 under the training/testing data of 80:20, 70:30 and 60:40 respectively. Then, with 10 h, the HPT-BiGRNN technique has gained an MSE of 0.040, 0.100 and 0.160 under the training/testing data of 80:20, 70:30 and 60:40 respectively. Simultaneously, with 24 h, the HPT-BiGRNN algorithm has obtained an MSE of 0.030, 0.070 and 0.180 under the training/testing data of 80:20, 70:30 and 60:40 respectively.

Table 1: MSE analysis of HPT-BiGRNN technique

Hours	MSE values		
	Train/Test (80:20)	Train/Test (70:30)	Train/Test (60:40)
1	0.030	0.080	0.110
2	0.030	0.090	0.090
3	0.040	0.070	0.090
4	0.040	0.070	0.090
5	0.040	0.090	0.110
6	0.030	0.050	0.110
7	0.030	0.060	0.070
8	0.110	0.160	0.260
9	0.040	0.060	0.100
10	0.040	0.100	0.160
11	0.030	0.120	0.130
12	0.060	0.140	0.190
13	0.040	0.140	0.170
14	0.040	0.170	0.120
15	0.060	0.170	0.190
16	0.020	0.090	0.080
17	0.040	0.110	0.170
18	0.050	0.140	0.300
19	0.080	0.070	0.140
20	0.050	0.080	0.110
21	0.030	0.120	0.190
22	0.030	0.090	0.200
23	0.040	0.080	0.190
24	0.030	0.070	0.180

**Figure 7:** MSE analysis of HPT-BiGRNN model

Tab. 2 and Fig. 8 validates the performance of the HPT-BiGRNN method with respect to RMSE under varying training/testing data. The figure demonstrated that the HPT-BiGRNN technique has resulted in effective outcomes with the lower RMSE values under varying duration and training/testing data size. For sample, with 1 h, the HPT-BiGRNN algorithm has reached an RMSE of 0.173, 0.283 and 0.332 under the training/testing data of 80:20, 70:30 and 60:40 respectively. Furthermore, with 5 h, the HPT-BiGRNN technique has attained an RMSE of 0.200, 0.300 and 0.331 under the training/testing data of 80:20, 70:30 and 60:40 respectively.

Table 2: RMSE analysis of HPT-BiGRNN technique

Hours	RMSE		
	Train/Test (80:20)	Train/Test (70:30)	Train/Test (60:40)
1	0.173	0.283	0.332
2	0.173	0.300	0.300
3	0.200	0.265	0.300
4	0.200	0.265	0.300
5	0.200	0.300	0.332
6	0.173	0.224	0.332
7	0.173	0.245	0.265
8	0.332	0.400	0.510
9	0.200	0.245	0.316
10	0.200	0.316	0.400
11	0.173	0.346	0.361
12	0.245	0.374	0.436
13	0.200	0.374	0.412
14	0.200	0.412	0.346
15	0.245	0.412	0.436
16	0.141	0.300	0.283
17	0.200	0.332	0.412
18	0.224	0.374	0.548
19	0.283	0.265	0.374
20	0.224	0.283	0.332
21	0.173	0.346	0.436
22	0.173	0.300	0.447
23	0.200	0.283	0.436
24	0.173	0.265	0.424

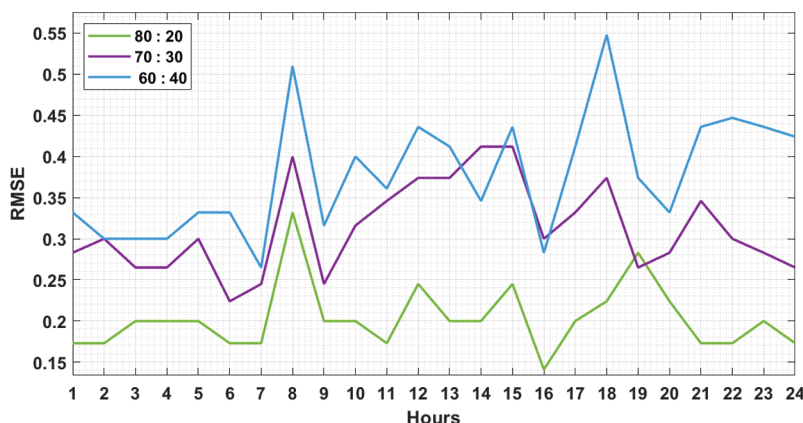


Figure 8: RMSE analysis of HPT-BiGRNN model

Then, with 10 h, the HPT-BiGRNN manner has attained an RMSE of 0.200, 0.316, and 0.400 under the training/testing data of 80:20, 70:30, and 60:40 correspondingly. Afterward, with 20 h, the HPT-BiGRNN technique has achieved an RMSE of 0.224, 0.283, and 0.332 under the training/testing data of 80:20, 70:30, and 60:40 correspondingly. Simultaneously, with 24 h, the HPT-BiGRNN methodology has obtained an RMSE of 0.173, 0.265, and 0.424 under the training/testing data of 80:20, 70:30, and 60:40 correspondingly.

A brief comparative results analysis of the HPT-BiGRNN technique with existing ones in terms of RMSE take place in [Tab. 3](#) and [Fig. 9](#) [23]. The comparison study portrayed that the HPT-BiGRNN technique has accomplished proficient results with the least RMSE values. For instance, with 1 h, the HPT-BiGRNN technique has resulted in a lower RMSE of 0.173 whereas the Fuzzy-GP, Fuzzy-SC, MLP-ANN, and RBF-ANN techniques have obtained a higher RMSE of 0.205, 0.225, 0.239, and 0.254. Similarly, with 10 h, the HPT-BiGRNN manner has resulted in a lower RMS of 0.200 whereas the Fuzzy-GP, Fuzzy-SC, MLP-ANN, and RBF-ANN algorithms have obtained an increased RMSE of 0.232, 0.243, 0.255, and 0.297. At the same time, with 15 h, the HPT-BiGRNN technique has resulted in a lesser RMSE of 0.245 whereas the Fuzzy-GP, Fuzzy-SC, MLP-ANN, and RBF-ANN methodologies have achieved a maximal RMSE of 0.251, 0.268, 0.273, and 0.289.

Table 3: Comparative RMSE analysis of HPT-BiGRNN technique with existing techniques

Hours	HPT-BiGRNN	Fuzzy-GP	Fuzzy-SC	MLP-ANN	RBF-ANN
1	0.173	0.205	0.225	0.239	0.254
2	0.173	0.218	0.226	0.239	0.260
3	0.200	0.224	0.230	0.238	0.273
4	0.200	0.226	0.233	0.247	0.263
5	0.200	0.224	0.234	0.250	0.267
6	0.173	0.218	0.227	0.255	0.274
7	0.173	0.260	0.378	0.354	0.314
8	0.332	0.361	0.406	0.424	0.365
9	0.200	0.237	0.253	0.276	0.417

(Continued)

Table 3 (continued)					
Hours	HPT-BiGRNN	Fuzzy-GP	Fuzzy-SC	MLP-ANN	RBF-ANN
10	0.200	0.232	0.243	0.255	0.297
11	0.173	0.237	0.255	0.267	0.271
12	0.245	0.268	0.273	0.300	0.283
13	0.200	0.225	0.251	0.234	0.312
14	0.200	0.272	0.279	0.267	0.282
15	0.245	0.251	0.268	0.273	0.289
16	0.141	0.227	0.247	0.261	0.297
17	0.200	0.250	0.274	0.291	0.242
18	0.224	0.265	0.314	0.328	0.285
19	0.283	0.297	0.310	0.292	0.313
20	0.224	0.263	0.256	0.234	0.343
21	0.173	0.241	0.244	0.228	0.301
22	0.173	0.214	0.232	0.219	0.272
23	0.200	0.220	0.232	0.226	0.254
24	0.173	0.218	0.216	0.202	0.245

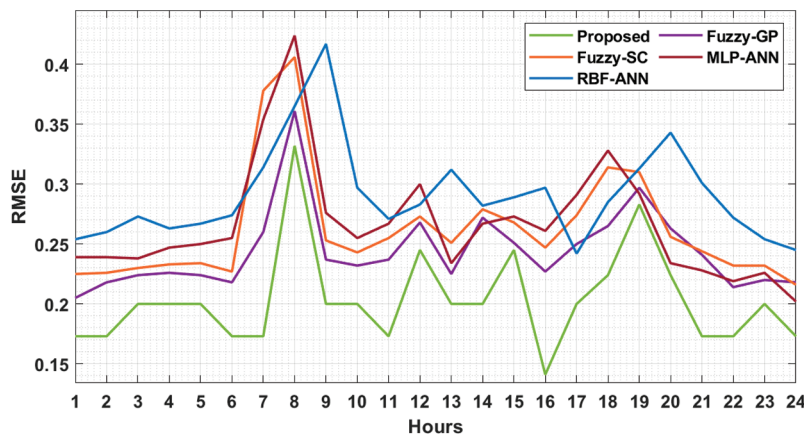


Fig. 9: Comparative analysis of HPT-BiGRNN model with existing techniques

Eventually, with 24 h, the HPT-BiGRNN approach has resulted in a lower RMSE of 0.173 whereas the Fuzzy-GP, Fuzzy-SC, MLP-ANN and RBF-ANN methods have gained an increased RMSE of 0.218, 0.216, 0.202 and 0.245 respectively. It is apparent that the HPT-BiGRNN technique has accomplished the maximum prediction results compared to the existing techniques.

5 Conclusion

In this study, a new BiGRNN technique is designed to effectively predict the weather conditions. The BiGRNN technique encompasses a two-stage process namely BiGRNN based prediction process and BMO

based hyperparameter tuning process. Primarily, the past weather data is used by the BiGRNN model to predict the weather condition. Later, the hyperparameters involved in the BiGRNN technique can be optimally tuned by the BMO algorithm. A wide range of simulation analysis take place on the benchmark dataset and its experimental results are investigated under various aspects. The detailed result analysis proves that the BiGRNN technique has outperformed the recent state of art approaches in terms of different measures. The experimental result the HPT-BiGRNN approach has resulted in a lower RMSE of 0.173 whereas the Fuzzy-GP, Fuzzy-SC, MLP-ANN and RBF-ANN methods have gained an increased RMSE of 0.218, 0.216, 0.202 and 0.245 respectively. In future, the BiGRNN technique can be tested using real time data to predict the rainfall and traffic flow.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] P. Hewage, M. Trovati, E. Pereira and A. Behera, "Deep learning-based effective fine-grained weather forecasting model," *Pattern Analysis and Applications*, vol. 24, no. 1, pp. 343–366, 2021.
- [2] L. Oana and A. Spataru, "Use of genetic algorithms in numerical weather prediction," in *18th Int. Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, Timisoara, Romania, pp. 456–461, 2016.
- [3] B. P. Sankaralingam, U. Sarangapani and R. Thangavelu, "An efficient agro-meteorological model for evaluating and forecasting weather conditions using support vector machine," *Smart Innovation, Systems and Technologies*, vol. 51, no. 1, pp. 65–75, 2016.
- [4] M. Reichstein, G. Camps-Valls, B. Stevens, M. Jung, J. Denzler *et al.*, "Deep learning and process understanding for data-driven earth system science," *Nature*, vol. 566, no. 7, pp. 195–204, 2019.
- [5] Y. Zheng, L. Capra, O. Wolfson and H. Yang, "Urban computing: Concepts, methodologies, and applications," *ACM Transaction of Intelligent System Technology*, vol. 5, no. 3, pp. 1–55, 2014.
- [6] J. Deepak Kumar, B. Prasanthi and J. Venkatesh, "An intelligent cognitive-inspired computing with big data analytics framework for sentiment analysis and classification," *Information Processing & Management*, vol. 59, no. 1, pp. 1–15, 2022.
- [7] S. Satpathy, S. Debbarma, S. C. Sengupta Aditya and K. D. Bhattacharyya Bidyut, "Design a FPGA, fuzzy based, insolent method for prediction of multi-diseases in rural area," *Journal of Intelligent & Fuzzy Systems*, vol. 37, no. 5, pp. 7039–7046, 2019.
- [8] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong *et al.*, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," *Advances in Neural Information Processing Systems*, vol. 28, no. 1, pp. 802–810, 2015.
- [9] R. Castro, Y. M. Souto, E. Ogasawara, F. Porto and E. Bezerra, "STconvs2s: Spatiotemporal convolutional sequence to sequence network for weather forecasting," *Neurocomputing*, vol. 426, no. 4, pp. 285–298, 2021.
- [10] M. Sundaram, "An analysis of air compressor fault diagnosis using machine learning technique," *Journal of Mechanics of Continua and Mathematical Sciences*, vol. 14, no. 6, pp. 13–27, 2019.
- [11] A. G. Salman, Y. Heryadi, E. Abdurahman and W. Suparta, "Single layer & multi-layer long short-term memory (LSTM) model with intermediate variables for weather forecasting," *Procedia Computer Science*, vol. 135, pp. 89–98, 2018.
- [12] S. Manikandan, S. Sambit and D. Sanchali, "An efficient technique for cloud storage using secured de-duplication algorithm," *Journal of Intelligent & Fuzzy Systems*, vol. 41, no. 2, pp. 2969–2980, 2021.
- [13] S. Rasp and S. Lerch, "Neural networks for postprocessing ensemble weather forecasts," *Monthly Weather Review*, vol. 146, no. 11, pp. 3885–3900, 2018.
- [14] N. Subramaniam and D. Paulraj, "An automated exploring and learning model for data prediction using balanced CA-SVM," *Journal of Ambient Intelligent and Humanized Computing*, vol. 12, no. 6, pp. 4979–4990, 2021.

- [15] P. Hewage, M. Trovati, E. Pereira and A. Behera, "Deep learning-based effective fine-grained weather forecasting model," *Pattern Analysis and Applications*, vol. 24, no. 1, pp. 343–366, 2021.
- [16] M. Singh, B. Kumar, D. Niyogi, S. Rao, S. S. Gill *et al.*, "Deep learning for improved global precipitation in numerical weather prediction systems," arXiv preprint arXiv:2106. vol. 120, no. 45, pp. 128–151.
- [17] R. Kamalraj, M. Ranjith Kumar, V. Chandra Shekhar Rao, A. Rohit and S. Harinder, "Interpretable filter based convolutional neural network for glucose prediction and classification using PD-SS algorithm," *Measurement*, vol. 183, no. 2, pp. 1–12, 2021.
- [18] X. Peng, H. Wang, J. Lang, W. Li, Q. Xu *et al.*, "EALSTM-QR: Interval wind-power prediction model based on numerical weather prediction and deep learning," *Energy*, vol. 220, pp. 119692, 2021.
- [19] M. J. Tham, "Bidirectional gated recurrent unit for shallow parsing," *Indian Journal of Computer Science and Engineering*, vol. 11, no. 5, pp. 517–521, 2020.
- [20] P. Li, A. Luo, J. Liu, Y. Wang, J. Zhur *et al.*, "Bidirectional gated recurrent unit neural network for Chinese address element segmentation," *ISPRS International Journal of Geo-Information*, vol. 9, no. 11, pp. 635–645, 2020.
- [21] A. Askarzadeh, "Bird mating optimizer: An optimization algorithm inspired by bird mating strategies," *Communication in Nonlinear Science and Numerical Simulation*, vol. 19, no. 4, pp. 1213–1228, Apr. 2014.
- [22] A. Arram, M. Ayob, G. Kendall and A. Sulaiman, "Bird mating optimizer for combinatorial optimization problems," *IEEE Access*, vol. 8, no. 4, pp. 96845–96858, 2020.
- [23] J. Faraji, A. Ketabi, H. Hashemi-Dezaki, M. Shafie-Khah and J. P. Catalão, "Optimal day-ahead self-scheduling and operation of prosumer microgrids using hybrid machine learning-based weather and load forecasting," *IEEE Access*, vol. 8, no. 5, pp. 157284–157305, 2020.