IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 4 November 2024, accepted 1 December 2024, date of publication 4 December 2024, date of current version 16 December 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3511113

RESEARCH ARTICLE

Nature-Inspired Driven Deep-AI Algorithms for Wind Speed Prediction

MUHAMMAD DILSHAD SABIR[1], LAIQ KHAN[1], KAMRAN HAFEEZ[1], ZAHID ULLAH[2], (Graduate Student Member, IEEE), AND STANISLAW CZAPP[3], (Member, IEEE)

[1]Department of Electrical and Computer Engineering, COMSATS University Islamabad, Islamabad 44000, Pakistan
[2]Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy
[3]Faculty of Electrical and Control Engineering, Gdansk University of Technology, 80-233 Gdansk, Poland

Corresponding author: Zahid Ullah (zahid.ullah@polimi.it)

This work was supported by the Politecnico di Milano for providing Open Access within the CRUI CARE Agreement.

ABSTRACT Predicting wind energy production accurately is crucial for enhancing grid management and dispatching capacity. However, the inherent unpredictability of wind speed poses significant challenges to achieving high prediction accuracy.To address this challenge, this study introduces a novel pre-processing framework that leverages thirteen nature-inspired optimization algorithms to extract and combine Intrinsic Mode Functions (IMFs) of atmospheric and wind speed variables. The objective function ensures that the selected IMF combinations exhibit high correlation, enhancing their predictive relevance.The outputs of these algorithms are further refined using the proposed Optimal Search IMF (OAIMF) algorithm, which reduces redundancy and selects a minimal yet highly relevant set of IMF combinations for wind speed prediction.The methodology was validated through a case study conducted at the Climate, Energy, and Water Research Institute (CEWRI), NARC, Islamabad, Pakistan, leveraging real-world atmospheric data.Experimental results demonstrate that the proposed framework significantly outperforms direct prediction methods and state-of-the-art pre-processing techniques. For instance, the framework achieved an RMSE of 2.73 on an LSTM network and 3.86 on a GRU network, compared to RMSE values of 19.78 and 18.89, respectively, for direct prediction. Superior performance was also observed across MAE, MAPE, and R^2 metrics.This study highlights the critical role of robust pre-processing in enhancing deep learning-based wind speed prediction. By integrating nature-inspired optimization with a novel IMF selection strategy, the proposed approach advances the state-of-the-art in renewable energy forecasting.

INDEX TERMS Wind speed, wind power prediction, renewable energy resource, deep learning, artificial intelligence, empirical mode decomposition, gated recurrent units, long short-term memory, nature-inspired algorithms.

NOMENCLATURE

RECNN	Residual CNNs.
SVM	Support Vector machine.
WPD	Wavelet packet decomposition.
OAIMF	Optimal Search IMF algorithm.
AGRU	Attentive mechanism GRU.
CLSTM	Convolution long short term memory.
GRNN	Generalized regression neural network.
MBE	Mean Bias Error.
MAE	Mean Absolute Error.
MAPE	Mean Absolute Percentage Error.
MSE	Mean Squared Error.
MPE	Main Percentage Error.
MRE	Mean Relative Error.
NRMSE	Normalized Root Mean Square Error.
NMAE	Normalized Mean Absolute Error.
MLP	Multi-layer perception.
IMF	Intrinsic Mode Function.
$N_l(t)$	Lower envelope of signal.
$N_u(t)$	Upper envelope of signal.
$\nabla L(\theta_i)$	Gradient of loss function.

The associate editor coordinating the review of this manuscript and approving it for publication was Thomas Canhao Xu.

M	Momentum.
θ_i	Training parameter i.
DF	Dragon Fly Algorithm.
GA	Genetic Algorithm.
BSO	Bird Swarm Optimization.
WOA	Whale Optimization Algorithm.
$COOT$	Coot Algorithm.
JSO	Jump Spider Optimization.
$COKOO-III$	Cokoo-III Algorithm.
MAO	Mexican Axolotl Optimization Algorithm.
PSO	Particle of Swarm Optimization.
$Firefly$	Firefly Algorithm.
$Harmony$	Harmony Algorithm.
ABC	Artificial Bee Colony Algorithm.
$COKOO-II$	Cokoo-II Algorithm.

I. INTRODUCTION

Renewable energy sources (RESs) have gained importance compared with conventional non–renewable energy sources due to numerous factors [1]. It is important for developing countries to increase the utilization of RESs in different energy sectors to achieve sustainable growth [2]. Amongst these RESs, wind energy is an important source due to its free availability and abundance in quantity [3]. It further reduces the pollutant emissions involved while producing electricity as well as a reduction in the operational costs of conventional plants [4]. The wind speed pattern is highly irregular and unpredictable [5]. Therefore, wind power integration into power grids will affect its stability and reliability [6]. Due to the large-scale integration of wind power and its variations, power system operation is disturbed as extra reserves are needed to balance net power [7]. This further increases the operating costs as changes occur in wind power plants output [8]. The irregular nature of wind turbine generators can produce voltage dips, frequency variations, and absorption of reactive power in a power system [9]. Therefore, precise prediction of wind speed is important in the regulation of a power system and addressing all challenges [10]. Wind power is influenced by two important parameters wind speed and its direction [11], [12]. The initiation of wind speed prediction is urgent and crucial for the efficient control of power in wind farms [13]. There are three steps involved in wind speed forecasting; wind speed prediction, wind data processing and performance criteria.

In [14], CNN and GRU are combined for estimating features and dependencies in wind series data. A deep belief network (DBN) is used for the prediction of wind speed [15]. In [16], a residual-based CNN is proposed that shows good performance. The short-term model is proposed that reduces the training time [17]. A stacked denoising auto-encoder (SDAE) based LSTM network is proposed with suitable feature selection to obtain the desired results [18]. Adaptive LSTM is applied to co-relate factors involved in the prediction of wind data [19]. An empirical mode decomposition (EMD) is used for the decomposition of wind data, and then the LSTM model is used for its forecasting [20]. LSTM-E (encoder-decoder) is used to predict wind speed time series data [21]. In [22] attention mechanism (A-GRU) is used for the feature extraction. Selecting the most valuable factors in wind power production, the residual convolution network (RE-CNN) is proposed by comparing lower level features with high level features to get the spatial information [23].

In literature, wind speed forecasting models can be categorized as physical models, statistical models, artificial intelligence (AI), and hybrid models [24]. The physical model depends on the climate and geographical data in order to forecast wind speed with long-term or short-term accuracy. The pre-processing of wind speed data is helpful in improving its accuracy [25]. In statistical models, time series data and real-time data correspond to each other to predict future behavior, making it challenging in terms of accuracy [26]. AI-based methods use algorithms to capture nonlinear attributes in wind speed data and achieve forecasting by training this data [27]. The deep learning method is applied for the prediction of wind speed using long short-term memory (LSTM) and convolution neural network (CNN) models, but it includes drawbacks of overfitting and degrading performance [28]. Recently, a lot of attention has been focused on Hybrid models due to their numerous advantages. In these models, pre-processing and optimized algorithms are used for filtering the raw wind speed data and improving prediction accuracy by further implementing deep neural networks [29].

A decomposition method is proposed to remove inflated noise from data using Generalized Regression Neural Network (GRNN) and Extreme Learning Machine (ELM) models [30]. A hybrid model with different deep learning networks, including CNN, LSTM, and CNN-LSTM, are compared, but results were strictly site-specific [31]. Another hybrid model with decomposition, forecasting and optimization modules is proposed. The data was pre-processed with wavelets transform accompanied by GRU (Gated recurrent unit) for perdition, and finally, SNN (Skipping Neural Network) is applied in reducing error [32]. A non-iterative decay method based on VMD and Convolution LSTM is used for predicting and LSTM method for error reducing [33]. These deep learning methods merge with different algorithms for data decomposition, and the feature option has shown better accuracy in the prediction of wind speed. However, more tunning parameters and processing time are needed, and optimization of parameters is required to achieve optimal performance. Nature-inspired algorithms can provide optimized solutions and are extensively used in different areas of renewable energy systems. These algorithms adopt nature's intelligence and behavior and resolve problems in several technical issues [34]. These algorithms include Genetic Algorithm (GA) [35], Particle Swarm Optimization (PSO) [36], Artificial Bee Colony (ABC) [37], Firefly (FA) [38] and grey wolf optimizer (GWO) [39].

The role of these algorithms is to discover the optimized performance parameters of predicted models [40]. A wind

forecasting structure is developed based on wind data pattern association with climate data improving the accuracy in predicting electric output power [41]. A PSO method is used to predict wind power utilizing ANN [42]. These nature-inspired individual algorithms indicate good performance in simple problems, whereas hybrid algorithms accomplish better results in addressing complex issues. However, difficulty in tuning of multi-parameters and computational complexity are disadvantages in the implementation stage. Therefore for wind power forecasting, the most suitable selection of meta-heuristic algorithm is essential.

Standard Empirical mode decomposition (EMD) [43] splits the signal into Intrinsic Mode Functions (IMFs) and a residual trend using an iterative sifting process, but it is subject to mode mixing and is highly sensitive to noise. Ensemble EMD (EEMD) [44] improves upon this by adding white noise, therefore, smartly reducing mode mixing through averaging over several decompositions and extracting IMFs. Complete Ensemble EMD with Adaptive Noise (CEEMDAN) [45] refines EEMD further by using adaptive noise for reducing mode mixing as well as preserving the characteristics of signals. Multivariate EMD (MEMD) [46] processes several associated signals at the same time while Bivariate EMD (BEMD) [47] deals with two interrelated signals together. EMD with data preprocessing applies techniques such as detrending before EMD so as to increase precision. Hilbert-Huang transform (HHT) combines EMD with Hilbert Transform in order to conduct an elaborate analysis of time-frequency domain. Nonlinear EMD [48] variants alter standard methods for effectively handling certain types of signals or improving performance in particular situations.

A. MOTIVATION AND RESEARCH CHALLENGES

The following motivation and key scientific challenges linked with the prediction of wind speed requires further investigation;

1) ***Energy Crisis:*** Fossil fuel are unfriendly to the environment and responsible for climate changes in the world. There is a mismatch between demand and supply and problem of power shortages can be addressed by utilizing renewable energy resources to generate electric power [49].
2) ***Renewable Energy Resources:*** Wind energy is considered as a clean and environmental friendly energy resource. However, wind speed prediction models need accurate climate data including parameters i.e., wind speed, humidity, and wind direction [50]. Due to wind speed irregular nature, interfacing with grid may create certain challenges and issues. Therefore, accuracy in wind speed prediction is an important factor in solving these issues [51].
3) ***Exploration of Deep Learning Methods:*** The wind power output from generator can be shown in Eqn. 1.

$$P = \frac{1}{2}\rho A v^3 \tag{1}$$

where ρ represents air density (kg/m^3), A is the extent area of wind turbine (m^2) and v shows wind speed in m/s. The relationship between wind power and wind velocity is almost 3 times. An error in wind speed forecasting can influence the wind power production significantly [52]. Deep learning algorithms are accurate in hybrid models but need additional parameters for training at the cost of processing time. To overcome this issue, different types of optimization algorithms interfaced with deep learning methods are to be tested and needs further exploration [53].

Nowadays, the AI and deep learning (DL) approaches have gained popularity in predicting wind speed [31]. The results of these studies dependent on several factors including variation in prediction window, different metrics and types of data used [54]. The future trends in wind speed forecasting will be architecture of input system, wind speed features, and effective utilization of metrics in improving accuracy.

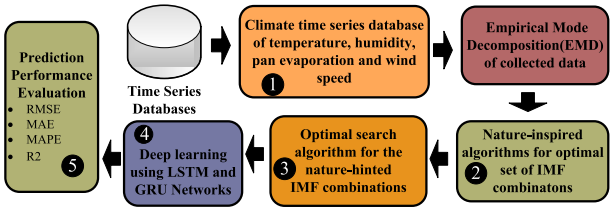


FIGURE 1. Optimization flow diagram of the proposed strategy for wind speed forecasting.

B. NOVEL CONTRIBUTIONS

Weather prediction is a complex endeavor, requiring knowledge and prediction of different atmospheric conditions. Wind speed is a critical element in predicting weather as it can affect wind power generation. Nonetheless, accurately predicting wind speeds, particularly during highly fluctuating conditions, is a challenge. In order to address the previously stated challenges, the following contributions have been claimed:

- To address the challenges, the proposed work utilized algorithms inspired by natural processes **(Fig. 1 step 2)**. Thirteen such algorithms have been evaluated for wind speed prediction. The aim of these algorithms is to discover the relationship between distinctive weather variables and produced wind speed. This is achieved by inspecting the correlation between combinations of Intrinsic Mode Functions (IMFs) of these variables. IMFs refer to the basic constitutive elements resulting from empirical mode decomposition (EMD) of time series data which are especially suitable for studying intricate and nonlinear systems such as the atmosphere. The proposed work aims to imitate the adaptability and efficiency of natural systems by using algorithms inspired by nature when dealing with complex interactions present in atmospheric data. These algorithms may be based on various natural phenomena like bird swarming, foraging by ants, or processes of optimization seen in biological evolution. The proposed technique aims to make wind

TABLE 1. Literature Review-state of the art.

Reference	Time interval	Model	Input parameters	Performance metric
Liu et al. [14]	15 min	CNNGRU	Wind speed	MAE 2.41, MAPE 0.97
Wang et al. [15]	10 min	DBN	Wind speed Direction	MAPE 1.1739
Yildiz et.al [16]	1 hour	RB-CNN	Wind speed, wind direction , wind power	MAE 0.0376, SMAPE 0.253
Z hang et al [17]	15 min	SW-LSTM	Wind speed	RMSE 0.174
Liu et al [18]	10 min	SDAE-LSTM	Wind speed, temperature, wind direction, and air pressure.	RMSE 0.3880, MAE 0.3066
Xu et al [19]	10 min	Adaptive LSTM	Wind speed, wind direction , temperature, wind generator data	MAPE 2.7
Huang et al [20]	1 hour	EEMD-LSTM	Wind speed	RMSE 0.83, MAE 0.71
Lu et al [21]	15 min	E-DLSTM	wind speed, temperature, air pressure, air density, wind direction	RMSE 2.6
Niu et al [22]	1 hour	A-GRU	wind power, wind speed temperature, air pressure and air density	MAPE 4.25, MAPE 14.94
Shivam et al [23]	10 min	Res-CNN	wind speed, wind direction, and wind power	RMSE 1.342, MAPE 0.272

speed forecasts more accurate and reliable by blending such algorithms with sophisticated modeling techniques and large data sets.

- An optimal search algorithm **(Fig. 1 step 3)** has been proposed to further refine the output of nature-hinted IMF combinations for wind speed predictions. This algorithm

ensures the most suitable and minimum number of IMF combinations sets for restoration of predictable wind speed data.
- Deep learning approaches **(Fig. 1 step 4)** like the LSTM and GRU networks have been employed for training and testing on IMF combinations outcomes of above mentioned optimal search algorithm.
- An atmospheric dataset **(Fig. 1 step 1)** collected from the Climate, Energy, and Water Research Institute (CEWRI), NARC, Islamabad has been analyzed for wind speed prediction.
- Subsequently, the experimental analysis **(Fig. 1 step 5)** indicates a higher predictive performance advantage as compared to a direct method over atmospheric and wind speed data recorded.

The proposed method outperformed conventional on various metrics for both LSTM and GRU networks. For LSTM, it achieved RMSE, MAE, MAPE, R^2 values of 3.72, 1.22, 1.53×10^{-3}, and 0.8, respectively. For GRU, it achieved RMSE, MAE, MAPE, R^2 values of 3.85, 1.22, 0.07, and 0.98, respectively. The three-wavelet approach yielded RMSE, MAE, MAPE, R^2 values of 12.7, 9.03, 0.03, and 0.79, respectively, for LSTM, and RMSE, MAE, MAPE, R^2 values of 12.56, 9.23, 0.51, and 0.79, respectively, for GRU. The direct approach had lower performance, with RMSE, MAE, MAPE, R^2 19.78, 14.38, 0.06, and 0.50 for LSTM, and 18.89, 13.67, 0.68, and 0.52 for GRU.

This paper has been divided into different sections. Every section offers the particulars of the corresponding part of the proposed method. Section I describes the introduction, literature review, contributions, motivation, and research challenges. The overview of the proposed prediction approach is presented in section II. Details of nature-inspired algorithms and their objective function are also provided in this section. Section III gives the details of experimental analysis including data acquisition, empirical mode decomposition, performance measures, and experimental results discussion. The proposed work is concluded in section IV.

II. PREDICTION FRAMEWORK

A. OVERVIEW

The proposed forecasting method is based on extracting the best time series signals from the recorded atmospheric variables. The proposed analysis of input weather features and wind speed begin with essential phase of time series data decomposition. Empirical mode decomposition (EMD) is one of the practical approach for data decomposition. To simplify the investigation, the EMD is employed **(Fig. 2 step 1)** which converts input weather variables $f_1, f_2, f_3, \ldots, f_n$ data into their corresponding intrinsic mode functions and residuals. Consequently, each weather feature f including wind speed data yields IMFs $e_1, e_2, e_3, \ldots, e_n$. In order to explore the expressive time series data, the approach explores each possible combination of the given $e_1, e_2, e_3, \ldots, e_n$ components for each weather feature. All the possible additions **(Fig. 2 step 2)** for given IMFs belonging to each feature are computed without repetition.

All the IMF combinations for each feature are gathered to find the correlation between input weather variables and wind speed data. To minimize the objective function, thirteen nature-inspired algorithms **(Fig. 2 step 3)** are evaluated for the given combinations belonging to each feature. This objective function ensures the selection of inputs that have maximum similarity with IMF combinations of wind speed data. λ is varied from $1.5, 2, 3, \ldots, 15$ for each algorithm evaluation. The outcome of each evaluation is an index row of six elements. Each index belongs to a specific IMF combination of a given input feature. The outcome is a 15×6 row matrix for each. As there are thirteen algorithms, the outcome is a 196×6 matrix. Each row of this matrix has an overall correlation, entropy, and number of combined IMFs of wind speed associated with it. Algorithm 01 **(Fig. 2 step 4)** ensures the selection of appropriate output and input signals from it. These signals are fed into deep learning **(Fig. 2 step 5)** algorithms for prediction. Where hyper-parameters of each deep learning algorithm are optimized using Particle Swarm Optimization. After combining each signal, the final step evaluates **(Fig. 2 step 6)** the prediction performance using RMSE, MAE, MAPE, and R2 measures.

B. NATURE-INSPIRED ALGORITHMS

In recent years, to mitigate environmental challenges and meet electricity demands, the grid increasingly requires renewable sources like wind to produce electricity. In case of wind energy, electricity generation is considered as of high potential due to its sustainable use as well as expandable capacity. Nonetheless, the way in which wind energy is utilized will depend largely on how well we understand the intricate interaction existing between atmospheric input parameters and wind speed produced.

The thirteen optimization algorithms are selected due to their ability to balance exploration and exploitation, robustness, versatility, adaptability, convergence time, adaptive mechanisms, implementation simplicity, and handling complex problems compared to other algorithms. These are all well-established swarm-based algorithms, with good performance in the literature

Each of these algorithms holds different strengths, and using them collectively provides quite a powerful search. DF, WOA, PSO, and Firefly algorithms provide a good balance between exploration and exploitation, which helps avoid local optima and improves the likelihood of finding the global optimum. GA, ABC, and CUCKOO algorithms are robust for different types of optimization problems (both continuous and discrete). It is their versatility that enables them to adjust to distinct problem features and limitations. Coot, MAO, and JSO are more adaptable to dynamic environments, where challenges evolve with time. JSO, PSO, and Firefly algorithms have fast convergence compared to other techniques. BSO, MAO, and Harmony algorithms maintain diversity among populations, avoid premature convergence, and through exploration. GA,

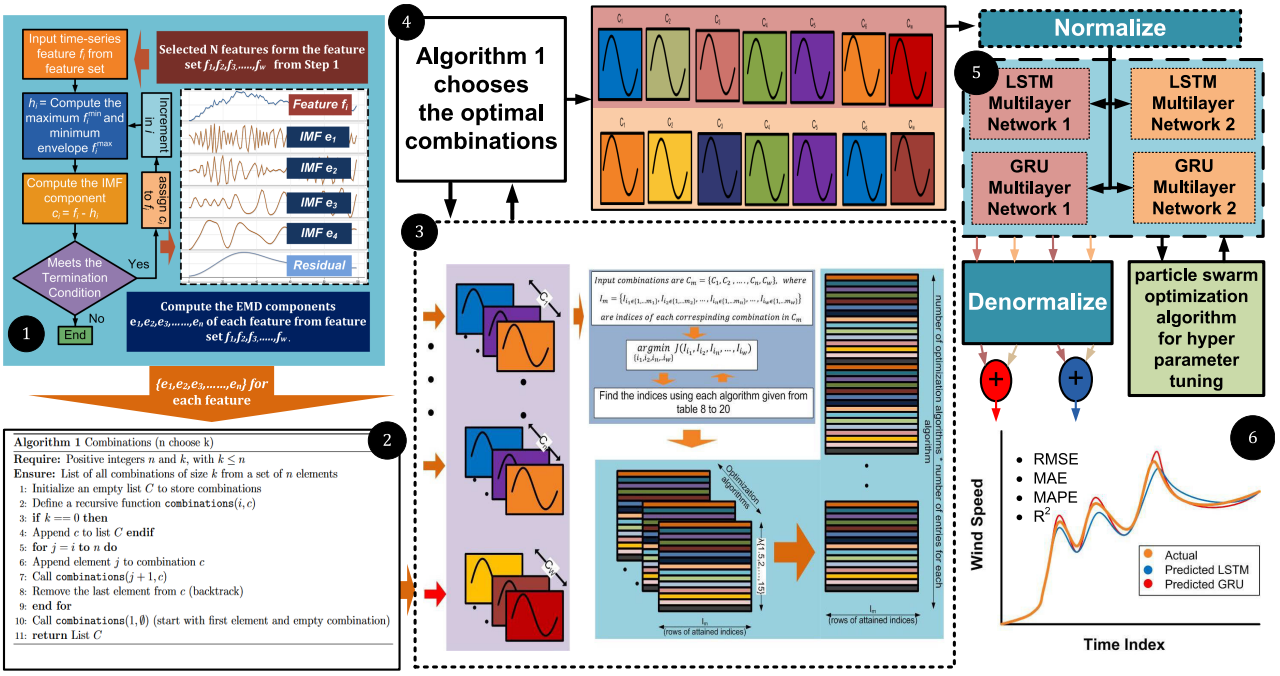


FIGURE 2. Block diagram of the step-by-step implementation of the proposed prediction technique.

PSO, and ABC algorithms are simple to implement and flexible, which makes them practical for various problems. DFA, WOA, Firfly, and Cuckoo algorithms are more suitable for handling complex and high dimensions problems. Coot, Harmony, and MAO algorithms can change their search strategies according to the characteristics of problems or the performance of the algorithm.

The connection between input and output prediction variables is important in raising the reliability of wind speed prediction. A high correlation between input versus output data implies any changes in input will influence observable outputs in a predictable manner making predictive ability better.

The same idea about a correlation can be extended to Intrinsic Mode Functions (IMFs) and their combination of atmosphere variables. Fig. 3 shows the atmospheric data versus day index (daily atmospheric data for the five-year horizon, i-e., 2016-2020). While the Fig. 5 shows selected IMFs of climate variables. These IMFs represent the inherent oscillatory modes within the atmospheric input data, offering a nuanced understanding of its dynamic nature. The study uses dot product analysis to determine the effect of different IMF combinations of input data on wind speed. The dot product is utilized as a measure to determine the relationship between IMF combinations of atmospheric input data and target wind speed data. Element-wise multiplication followed by the addition of the two vectors determines the similarity between them. The higher value of the scaler product identifies more similarity between two vectors. The R ratio is obtained from the scalar product divided by the modulus (magnitude) of the two vectors providing important information on how much stronger or weaker these atmospheric input quantities are related to wind speed data hence guiding predictive modeling as well as a decision-making process.

$$R = \frac{C.W}{|C||W|} \tag{2}$$

Eqn. 2 utilizes two variables C and W in optimization of the correlation between weather and wind speed. Here C denotes IMFs combinations of weather factors whereas W denotes IMFs combinations. The correlation measure ranges from -1 to $+1$, signifying relationship strength between the two variables. A score approaching $+1$ implies a strong positive correlation while one close to -1 indicates negativity. On the other side, values close to zero or precisely zero imply a lack of connection between them. The main idea of this analysis is to determine the optimal IMFs combinations C and W, thereby maximizing the correlation between weather conditions and wind speed, crucial for enhancing efficiency and performance in renewable energy systems.

Algorithm 1 initializes with matrix, M' where each row represents indices of combinations of input and wind features for a lambda value using a specific optimization algorithm. Each row has a corresponding Entropy, E and objective criteria value ξ. Algorithm 1 starts **(line 3)** by separating the indices of wind data from the given matrix, M' which is performed by collecting all the rows with a fixed column no. 6. The next step **(line 4)** involves retrieving the set of numbers that contains the constituent IMF numbers for each index of wind data. Each set length is different. The maximum length is chosen in the next step **(line 5)**. There are multiple indices as there are more than one maximum length exists in the input vector. These indices are further narrowed down by looking for a minimum value of the objective for the above indices. Line 6 provides these indices where objective values are minimum. Lastly, resultant

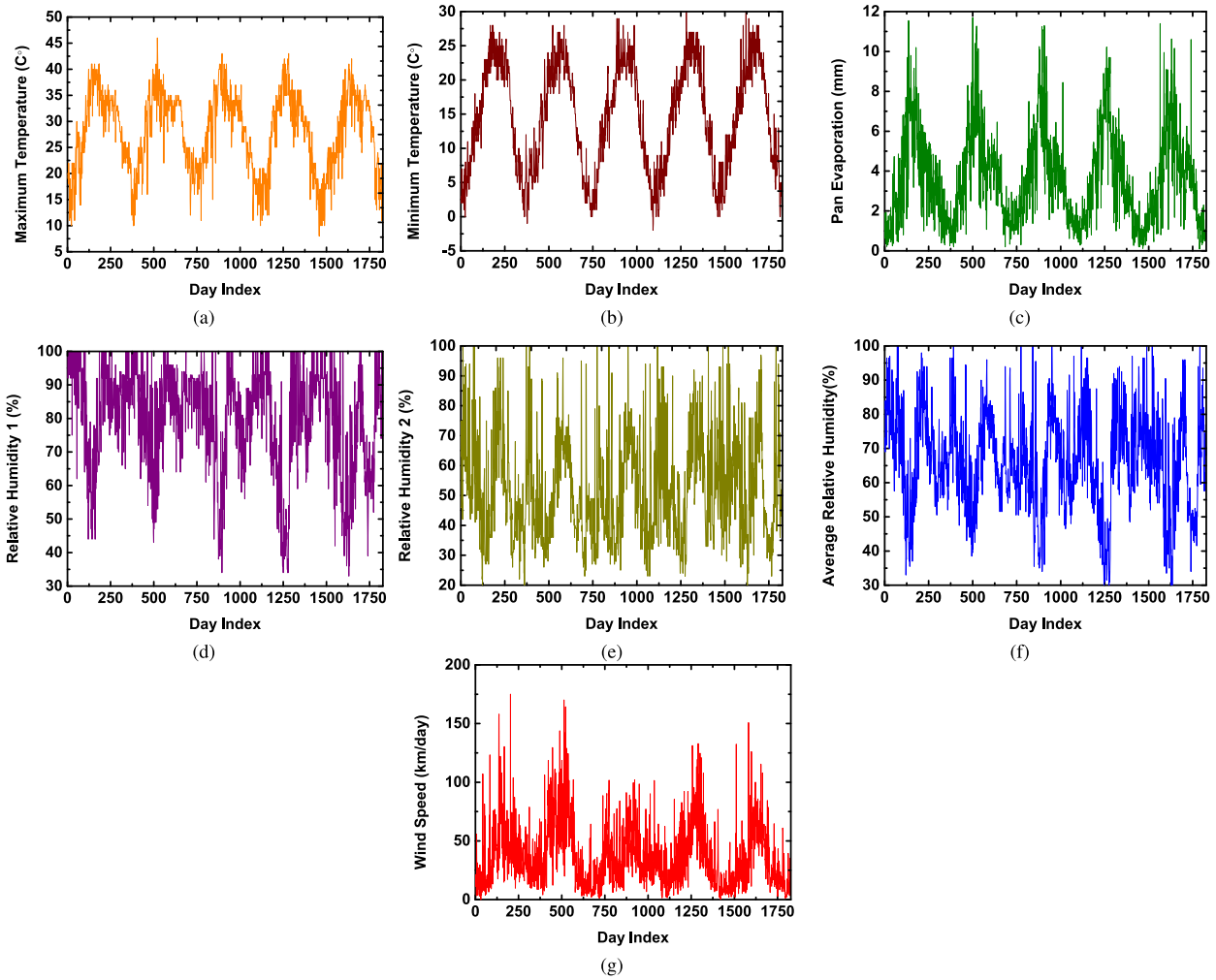


FIGURE 3. Graphs of data acquisition through different atmospheric sensors for duration of 2016 to 2020 are presented in (a) Maximum Temperature (b) Minimum Temperature (c) Pan Evaporation (d) Relative Humidity Recording 1 (e) Relative Humidity Recording 2 (f) Average Humidity Reading (g) Wind Speed.

TABLE 2. Metrics used in algorithm 1.

Variables	Metrics	Comments
M'	Matrix	Selected indices of each weather feature by Nature-inspired algorithms
N	Vector	Number of row of M'
w_i	Vector	Indices of Wind speed
p'_i	Matrix	Constituent IMFs sets related to w_i
e''_i	Vector	Indices of maximum length sets p'_i
c'''_i	Vector	Indices of minimum objective function of IMF combinations pointed by e''_i
l	Index	Index of maximum entropy of IMF combination pointed by c'''_i
f	IMFs set	IMFs set pointed out by l
s_f	IMFs set	set of complete IMFs related to wind speed
s'	IMFs set	set difference of s_f and s
s	IMFs set	Union between s and f
w'	Indices vector	Indices of all IMF combination other than f
o	index	Entropy
E	value	Entropy
ξ	value	Objective value

indices **(line 7)** are fed to the entropy tensor to provide a single index l.

By indexing l into p, an IMF numbers set is retrieved into set f. This set is subtracted **(line 9)** from s_f in order to find out the IMF number other than f. The resultant set **(line 11)** is given to *invconstituants* to find the combination waveform w'. **Line 10** takes union with set f and s to add the number of set f into set s. **Line 12** calls the function step 2, which returns the new M', E and ξ. After the first iteration, set s will become populated. Therefore, in the second iteration *else* section of the algorithm executes. However, this time the algorithm only picks the index of minimum objective function value. **Line 15 to 17** find the appropriate set f like in *if* part of the code. **Line 18** performs the union s_f and f. Unlike *if* section, the **line 19** identifies the subtraction of s from s_f. If the result in s' is non-empty, **lines 21 and 22** will execute, otherwise, they skipped. While loop will run until s is not equal to s_f.

EMD algorithm decomposes the data regarding climate conditions of minimum and maximum temperature, pan evaporation, relative humidity 1, relative humidity 2, average relative humidity, and wind speed into 7, 8, 10, 9, 9, 9, and 10 individual components. The proposed approach should

Algorithm 1 Optimal Search Algorithm (OAIMF) for the Nature-Hinted IMF Combinations of Wind Speed

Input: Matrix M', vectors E, ξ, sets s, s_f and initialize s=$\varnothing$
Output: *o=invconstiuants(f)*

while $s \neq s_f$ **do**
  if s=$\varnothing$ **then**
    $w_{i=1,2,\ldots,N} \leftarrow M'_{:,7}$
    $p_{i'=1,2,\ldots,N,j} \leftarrow constituants(w_{i=1,2,\ldots,N})$
    $e_{i''=1,2,\ldots,N} \leftarrow \text{argmax}_{\forall i'}(length(p_{i'=1,2,\ldots,N,j}))$
    $c_{i'''=1,2,\ldots,N'} \leftarrow \text{argmin}_{\forall i''}(\xi(e''_i))$
    $l \leftarrow \text{argmax}_{\forall i'''}(E(c_{i'''}))$
    $f \leftarrow p_{l,j}$
    $s' \leftarrow s_f - f$
    $s \leftarrow s \cup f$
    $w' \leftarrow invconstituants(s')$
    $M', E, \xi \leftarrow functionsetp2(w')$
  else
    $ind \leftarrow \text{argmin}_{\forall i''''}(\xi_{i''''})$
    $w_{v=1,2,\ldots,N} \leftarrow M'_{:,7}$
    $p_{i=1,2,\ldots,N,j} \leftarrow constituants(w_{v=1,2,\ldots,N})$
    $f \leftarrow p_{ind,j}$
    $s \leftarrow s \cup f$
    $s' \leftarrow s_f - s$
    if $s' \neq \varnothing$ **then**
      $w' \leftarrow invconstituants(s')$
      $M', E, \xi \leftarrow functionsetp2(w')$
    end if
  end if
end while

explore all the possible combinations of these components. Therefore, there are 119, 246, 1012, 501, 501, 1012, and 1012 numbers for combination for each climatic condition. In total, the proposed approach has to pick optimal sets from $119 \times 246 \times 1012 \times 501 \times 501 \times 501 \times 1012 \approx 3.77 \times 10^{18}$ number of possibilities. This selection is performed through a naturally inspired algorithm and proposed search algorithm 1.

The primary selection of IMF combinations is performed by naturally inspired optimization algorithms. These algorithms explore IMF combinations with the optimal value of the objective function. These multiple approaches range λ from 1.5 to 15, resulting in multiple choices of IMF combinations for each climate feature. Therefore, there is a requirement to find the most appropriate IMF combinations for the feature of wind speed, because, the sum of these IMF combinations should reconstruct the original wind speed data. Algorithm 1 selects two combinations of IMFs related to wind speed data. It ensures the minimization of the number of combinations. Each wind-related IMF combination will require a separate training mechanism for prediction.

Minimizing these numbers of wind speed-related IMF combinations will directly reduce the computational requirements. However, that is only one requirement, the second is to have weather-related IMF combinations that have maximum similarity with these two selected IMF combinations related to wind speed data.

Table 3 identifies these choices where the first selection corresponds to wind speed combination index 1002. This combination index relates to the combination of IMF numbers from 3 to 10. The corresponding objective function value, ξ is 7.26. The correlation coefficient, R of minimum and maximum temperature, pan evaporation, relative humidity 1, and relative average humidity is around 0.8, which shows the high similarity between these variables and wind speed data. While the relative humidity 2 data exhibits a correlation value below 0.8. Table 3 also shows corresponding constituent IMF sets for each case. Subsequently, algorithm 1 is left with only the choice to pick the combination index 1. The signal of this index has the constituent IMF set of {1, 2}. The given method outcomes in weather-related IMF combinations with small similarities or even with opposite polarity. The corresponding objective function value for this selection is 97.18, which is way higher compared to the first IMF combination index of 1002. Figs. 6 and 7 are illustrating the selected combinations and their constituent IMFs set corresponding to each atmospheric variable for wind speed related 1 and 2001 index, respectively.

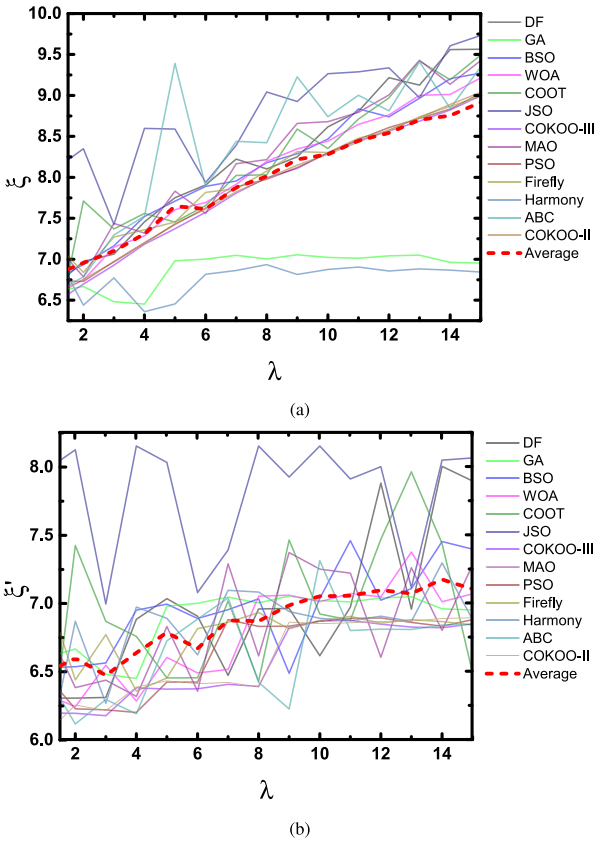


FIGURE 4. Representation of a relationship between regularization factor λ and objective function value ξ in (a) and ξ' in (b) of thirteen nature-inspired algorithms and their mean value.

C. FORMULATION OF OPTIMIZATION FUNCTION

Nature-inspired algorithms can mimic the principles and behaviors seen in nature to address difficult optimization or

search problems. These algorithms draw ideas from natural evolution processes such as survival of the fittest breed and inheritance of genes as well as from social animals' generic interactions like ants, bees, and birds. Inspired by nature, these algorithms are powerful tools for addressing a wide range of challenges.

Nature-based algorithms have the ability to adapt and improve on their own with time. They respond to varying situations and adjust their strategies dynamically like the animals habituate themselves to the new conditions. Additionally, algorithms inspired by nature are frequently strong as well as flexible, so they are the right choice for solving massive optimization problems in complicated systems where traditional methods may struggle.

In order to facilitate the prediction algorithm, inputs with high correlation to output should be selected. However, in addition to that, a minimum number of outputs for training is required, which is only possible when we have an output IMF combination of more IMFs. In objective function, nature-inspired algorithms search to minimize the reciprocals of the sum of correlations between inputs and output and the weighted reciprocal of the number of IMFs in the output IMF combination. This formulation will ensure heightened correlations between inputs and output along with the selection of an output IMF combination with a high number of IMFs.

In our case, we have various climate features which include different types of temperature, pan evaporation, and humidity readings. To analyze their behavior, the EMD method decomposes them into different IMFs. The proposed method inclines to find the maximum similarity between input feature and wind speed data. Therefore, the approach dives to consider the combinations of these IMFs, and for all seven variables we have nearly 3.77×10^{18} possibilities of combinations. In order to find the most suitable consolidation, the approach should have an objective function. Nature-inspired algorithms should minimize the following objective function, ξ:

$$\xi = \sum_{i}^{N} \frac{1}{R_i} + \frac{\lambda}{n_w} \tag{3}$$

$$\xi' = \sum_{i}^{N} \frac{1}{R_i} \tag{4}$$

In Eqn. 3, ξ is the objective function, and in Eqn. 4, ξ' denotes the objective function without the regulation term. N is the total number of input weather features. Whereas, R_i denotes the correlation value of wind speed IMF combination and IMF combination related to other weather variables. n_w denotes the number of constituents IMFs for a particular IMF combination related to wind speed data. λ signifies the regularization term which ranges from 1.5 to 15. The larger value of λ forces these algorithms to select the IMF combination with the large number of constituents IMFs related to wind speed which is the desired goal. However, increasing the values of λ will also increase the average objective function values, ξ and xi' in Fig. 4(a,b) for all the nature-inspired algorithms. Therefore, there is a necessity to seek a better optimization approach for selecting the best group of IMFs for all weather parameters. Algorithm 1 searches the best group of IMFs among many optimized IMF combinations. Tables 8-20 show the results of searching of these algorithms. Further, table 3 demonstrates entries from all these algorithms based on the best objective value, ξ.

III. EXPERIMENTAL RESULTS AND DISCUSSION

A. DATA ACQUISITION

In order to predict the wind speed for a short term, a duration of 24 hours is considered. The data gathered at the Climate, Energy and Water Research Institute (CEWRI) field station, National Agricultural Research Centre (NARC), Islamabad, Pakistan. The data accumulation is performed on daily basis at 0900. The latitude and longitude coordinates for data registration are 33.4^o North and 73.8^o East at an altitude of 1632 feet. The atmospheric field station is equipped with multiple sensors, which record the data of wind speed, maximum temperature, minimum temperature, pan evaporation, relative humidity at two different times e.g., humidity 1, humidity 2, and their average relative humidity. Fig. 8 shows some of these sensors. For pre-processing and training purpose, the considered duration is five years. The considered daily data range from 2016 to 2020.

B. EMD

In signal processing and data analysis, where complex datasets are involved, they cannot be analyzed using traditional methods due to their non-linearity and non-stationary nature. As an effective technique, Empirical Mode Decomposition (EMD) is seen as having the ability to unfold complex dynamics within such datasets. In the late 1990s, Huang et al. developed EMD as a method of decomposition signals into a finite number of intrinsic mode functions (IMFs) as well as the residual. Unlike Fourier-based methods which use predetermined basic functions, EMD decomposes signals into IMFs that are capable of capturing localized oscillations adaptively. The key steps involved in EMD include:

1) Data Preparation: The entire process starts with the decomposition of a one-dimensional signal $x(t)$ into its internal oscillatory elements.
2) Identification of Extrema: The second step includes the identification of all the local maxima and minima related to that signal $x(t)$. These points are referred to as extrema.
3) Construction of Upper and Lower Envelopes: Attach the maxima and minima to create the upper and lower envelopes respectively. These envelopes, however, should be equisurfaces passing through the maxima and minima.
4) Calculation of Mean Envelope: Calculate a mean m envelope using the upper $N_u(t)$ and lower $N_l(t)$ ones. The signal's local trend can be illustrated by this mean

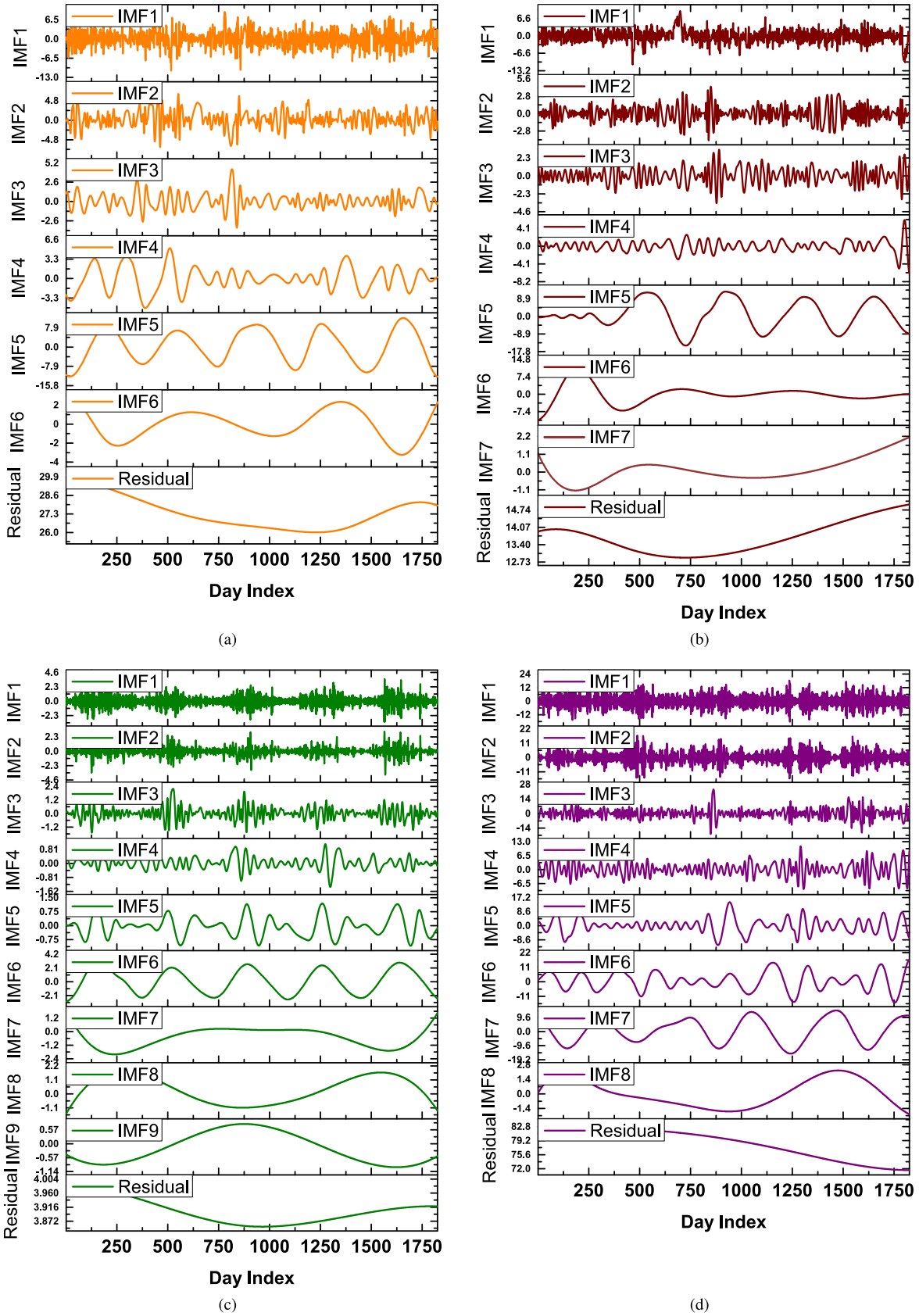
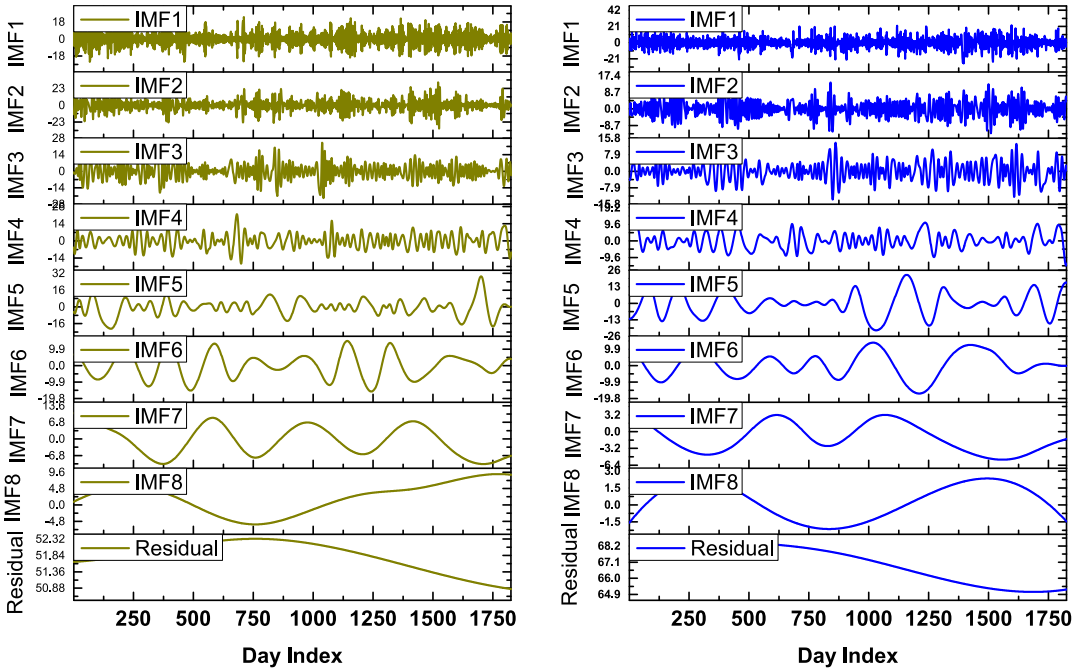
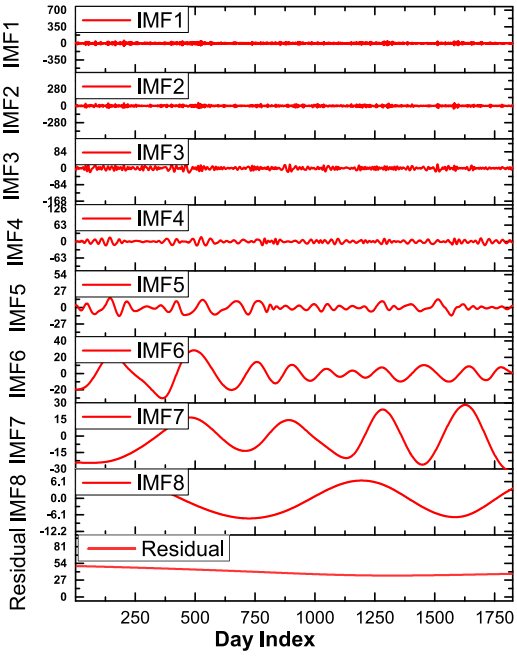


FIGURE 5. Representation of IMFs and Residual decomposed components by Empirical Mode Decomposition algorithm of different atmospheric sensors for duration of 2016 to 2020 in (a) Maximum Temperature (b) Minimum Temperature (c) Pan Evaporation (d) Relative Humidity Recording 1.



(e) (f)



(g)

FIGURE 5. ***(Continued.)*** **Representation of IMFs and Residual decomposed components by Empirical Mode Decomposition algorithm of different atmospheric sensors for duration of 2016 to 2020 in (e) Relative Humidity Recording 2 (f) Average Humidity Reading (g) Wind Speed.**

envelope. Eqn. 5 represents the same.

$$m = \frac{N_u(t) + N_l(t)}{2} \tag{5}$$

5) Extraction of the First Intrinsic Mode Function (IMF): In Eqn. 6, after subtraction of the mean envelope from original signal, which is denoted as $I_1(t)$ resonates with the fast oscillations that are found in the original signal. This is first IMF.

$$I_i = x - m \tag{6}$$

6) Repeat the EMD Process: Consider $I_i(t)$ as a fresh signal and employ steps 2-5 again to extract another IMF at t, where this process is iterative until a given stopping

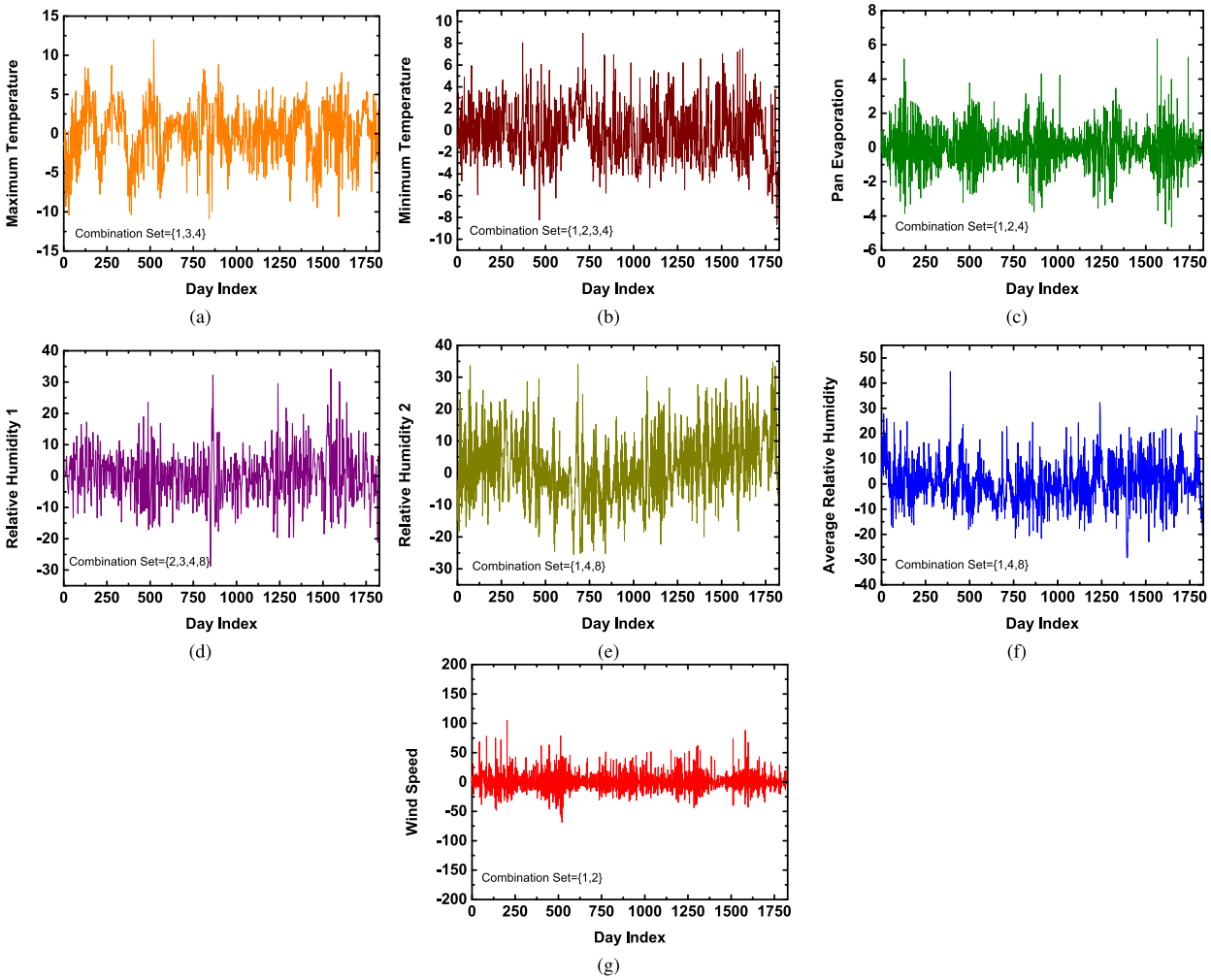


FIGURE 6. Graphs of 1st optimal IMFs combination chosen by nature-inspired algorithms and algorithm 1 of different atmospheric sensors for duration of 2016 to 2020 are presented in (a) Maximum Temperature (b) Minimum Temperature (c) Pan Evaporation (d) Relative Humidity Recording 1 (e) Relative Humidity Recording 2 (f) Average Humidity Reading (g) Wind Speed.

TABLE 3. Selection of IMFs combinations of wind speed data and weather variables with corresponding similarities, objective value ξ, and entropy.

	Input Features						Output	Performance Criterion's	
Features	**Maximum Temperature**	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	ξ	**Entropy**
Combination Index	54	137	627	237	120	220	1002	7.265522	0.45
Correlation Value, R	0.8487	0.8304	0.8726	0.8077	0.7981	0.8026			
IMF Set	{4,5,7}	{2,5,6,8}	{6,7,8,9,10}	{4,5,8,9}	{7,8,9}	{3,4,7,9}	{3,4,5,6,7,8,9,10}		
Combination Index	27	85	47	180	53	53	1	97.1845	1.61
Correlation Value, R	0.0413	0.0689	0.1202	-0.0649	-0.047	-0.0746			
IMF Set	{1,3,4}	{1,2,3,4}	{1,2,4}	{2,3,4,8}	{1,4,8}	{1,4,8}	{1,2}		

rule is achieved. Frequently, this rule can be the point at which a set number of IMFs are achieved or when the residue is too tiny.

7) Residue Extraction: In Eqn. 7, by subtracting the sum of all extracted IMFs from the original signal, the residue is derived. Residue refers to the remaining signal or low-frequency components that were not caught by IMFs.

$$Residual = x - \sum_{i}^{N} I_i \tag{7}$$

8) Repeat EMD on the Residue: Sometimes, it is advantageous to apply EMD recursively on the residue,

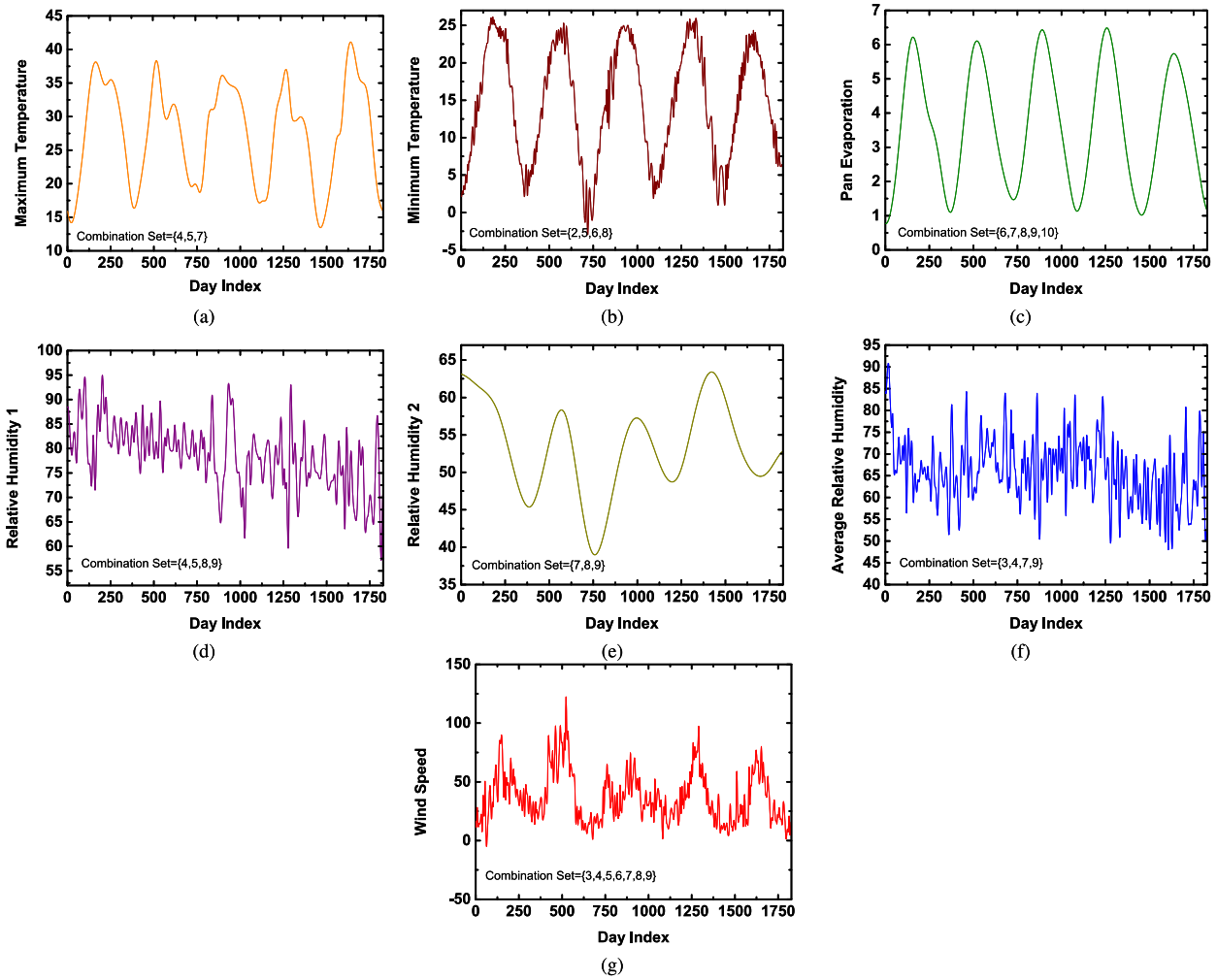


FIGURE 7. Graphs of 2^{nd} optimal IMFs combination chosen by nature-inspired algorithms and algorithm 1 of different atmospheric sensors for duration of 2016 to 2020 are presented in (a) Maximum Temperature (b) Minimum Temperature (c) Pan Evaporation (d) Relative Humidity Recording 1 (e) Relative Humidity Recording 2 (f) Average Humidity Reading (g) Wind Speed.

TABLE 4. Experimental results of the proposed technique and their corresponding time cost.

	GRU Network			LSTM Network		
	Three Wavelet	Direct	Proposed	Three wavelet	Direct	Proposed
RMSE	12.56	18.89	3.85	12.70	19.78	3.72
MAE	9.23	13.67	1.22	9.03	14.38	1.22
MAPE	0.51	0.68	0.07	0.03	0.06	1.53×10^{-3}
R-seq	0.79	0.52	0.98	0.79	0.50	0.80
Time Cost (msec)	217.8	80.95	1257.6	204.5	260.31	1309.2

TABLE 5. PSO optimized hyper-parameters for LSTM and GRU.

	Output Combination	LSTM /GRU Units	FC Units	L2 Regularization	Initial Learning Rate
LSTM	Wind Speed-1	461	295	0.01	7.88×10^{-3}
	Wind Speed-2	423	312	1.36×10^{-2}	7.61×10^{-3}
GRU	Wind Speed-1	474	1	0.01	9.97×10^{-3}
	Wind Speed-2	137	1	0.01	0.01

TABLE 6. Computational resources for LSTM and GRU models training.

Resource	Specifications
Software Tool	Matlab 9.14.0
Memory (RAM)	8GB
CPU	Intel(R) Core(TM) i5-2500K CPU @ 3.30GHz
GPU	NVIDIA GeForce GTX 1070 8GB

such that more IMFs and one residue are obtained. This process can be continued until the residue is insignificant or until some decomposition target is achieved. However, the step is optional.

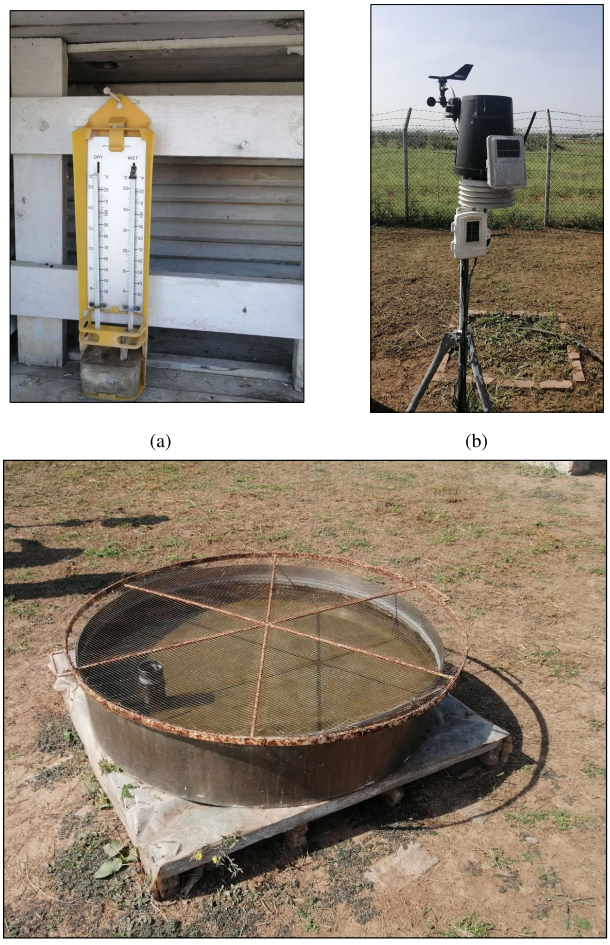


FIGURE 8. Data sensors are located at CEWRI field station, NARC, Islamabad, Pakistan for atmospheric data collection (a) Temperature Sensor (b) Wind and humidity sensors (c) Pan Evaporation.

TABLE 7. Architecture of LSTM or GRU models.

Layer Name	Description
Sequence Input	Sequence input with 7 dimensions
LSTM/GRU	LSTM or GRU layer
Fully Connected	Fully connected layer
ReLU	ReLU layer
Fully Connected	1 neruron fully connected layer
Regression Output	mean-squared-error with response 'Response'

9) Termination Condition: The EMD process ends if the defined stop rule is reached when further decomposition does not yield significant additional information.
10) Post-processing: When the decomposition process is completed, some post-processing techniques may be used, including filtering and reconstruction of the extracted IMFs along with the residue, according to the application requirements.

By adhering to these measures, Empirical Mode Decomposition (EMD) facilitates the splitting up of signals into intrinsic oscillatory components giving beneficial perspectives to the fundamental dynamics of data. EMD enables to effectively analyze intricate signals that have a nonlinear characteristic. This method gives us an idea about different datasets' basic dynamics. EMD helps us remove useful data by separating the signals into what we call IMFs adaptively. It means that by using EMD, one has an opportunity to get useful information for different signal-processing applications.

C. PERFORMANCE MEASURES

In mathematical modeling, Root Mean Square Error (RMSE) can be thought of as a foundation stone, because, it acts as a strong measure that assesses how far apart forecasts from reality are. The basic meaning of RMSE is the average difference between estimator predictions and actual observations. The disparity between data inputs and the regression line epitomizes a pivotal point when assessing how well a predictive model would perform in the future. This difference computes unambiguously as the Root Mean Square Error (RMSE) which is also the standard deviation of these differences known as the residuals. In other words, the residuals are the differences between the actual values and the ones forecasted by a model. The RMSE functions as a gauge that measures how closely the data points fall on the regression line. When the RMSE is close to one, it means that there are almost no residuals and, consequently, the data points are tightly clustered around the regression line. In other words, if you have a small RMSE value, then your model is more accurate than when you have a higher one. RMSE measures are not only statistical but enable to predict with greater accuracy.

$$RMSE = \sqrt{\frac{\sum_i^n (y_i - x_i)^2}{n}} \tag{8}$$

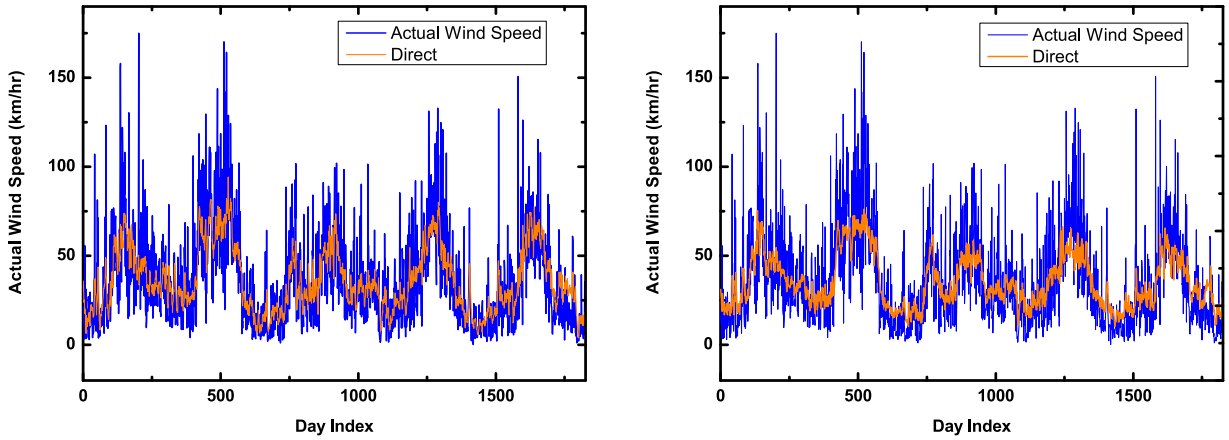
In Eqn. 8, y_i denotes the original or ground truth values while x_i represents the output values of the prediction model and n denotes the total length of y_i or x_i.

In the realm of data analytics and predictive modeling, Mean Absolute Error (MAE) is a simple and fundamental metric that describes the accuracy of predictions. At its core, MAE measures the degree to which predicted values deviate from their actual values. Essentially, MAE computes differences between individual predicted values (forecasted observations) against actual ones without considering their directions. This property allows MAE to handle exceptions or extreme values more safely than others since it gives the same value for every deviation without considering its size and sign. Mathematically, the formula for MAE is straightforward:

$$MAE = \frac{1}{n}\sum_{i}^{n}|y_i - y_i'| \tag{9}$$

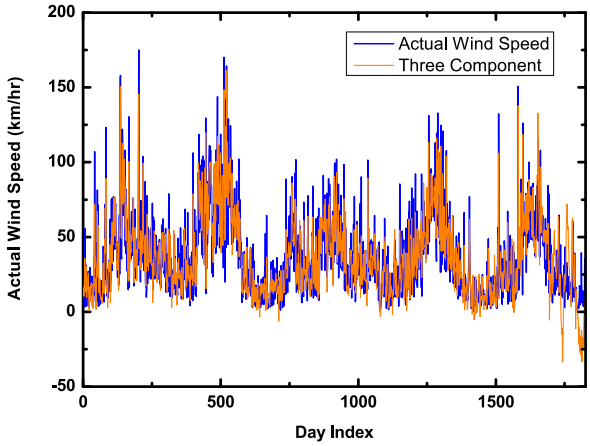
whereas, in Eqn. 9, n represents the total number of data points and y_i denotes the original or true value of the i^{th} data point. While y_i' represents the predicted value for the i^{th} data point and $|.|$ denotes the absolute value operator.

MAE provides an aggregate view of the total predictive performance of a model by taking the sum of the absolute

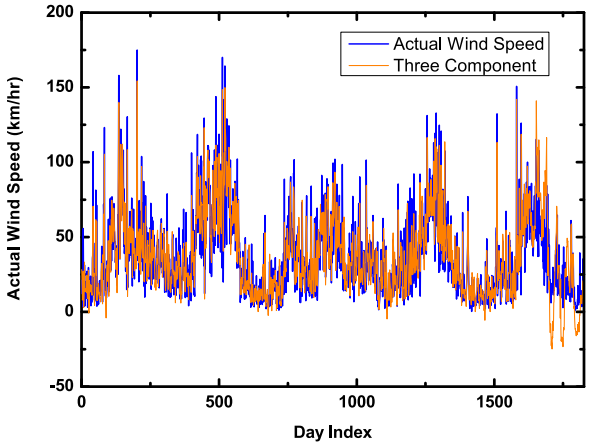


(a) GRU-based direct training results

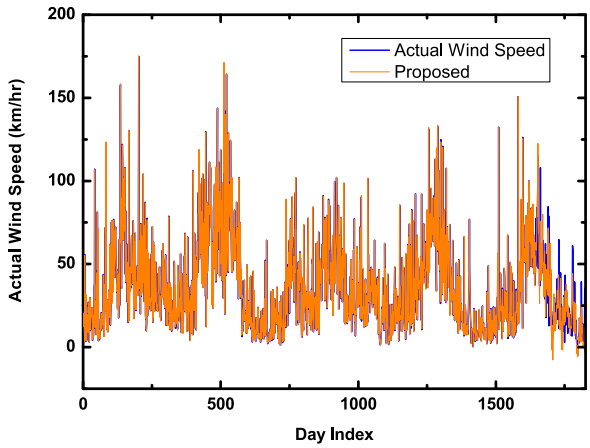
(b) LSTM-based direct training results



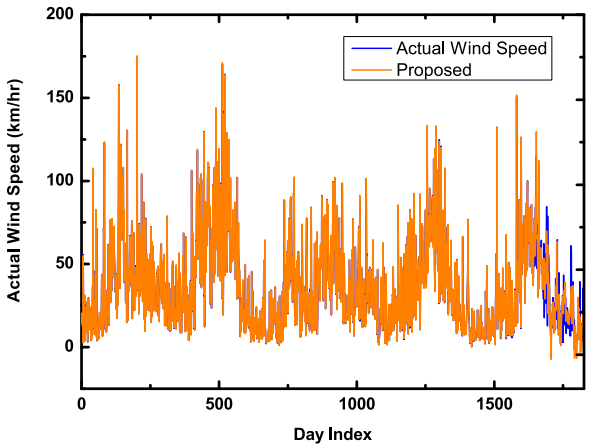
(c) GRU-based training results using three decomposed signals



(d) LSTM-based training results using three decomposed signals



(e) GRU-based training results using proposed method

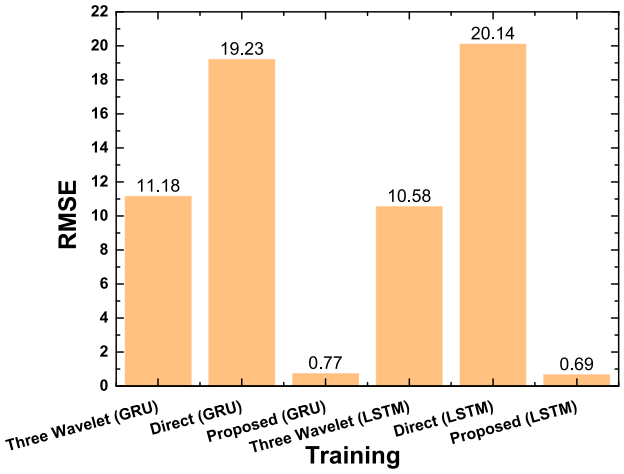


(f) LSTM-based training results using proposed method

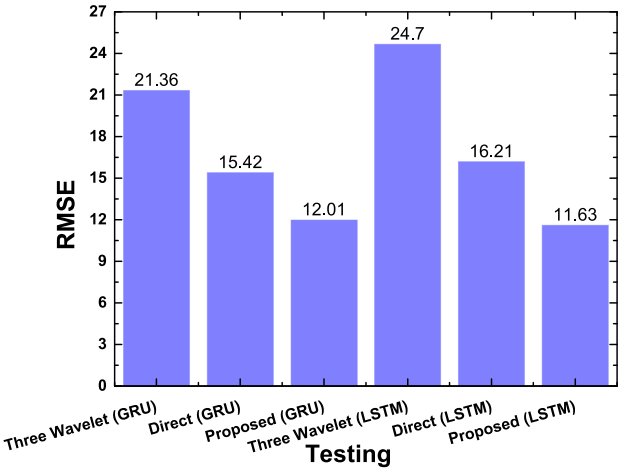
FIGURE 9. **Results graphs of the proposed and comparative techniques.**

differences between predicted values and true values for each point and then averaging them. One of the key advantages of MAE is its interpretability. The mean absolute error (MAE) in contrast to other error metrics is not squared, meaning that it preserves the original scale of a dataset which makes it more intuitive to interpret prediction errors as well as being directly interpretable into original units. MAE is a robust framework to evaluate model performance across various datasets and contexts, because, this method is simple to calculate and easy to understand.

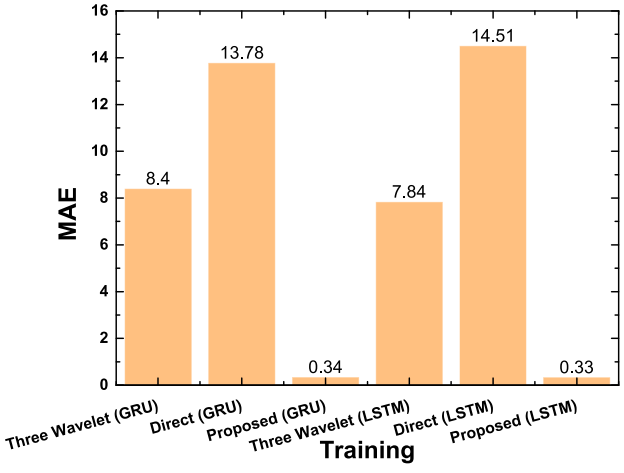
The mean absolute percentage error (MAPE) serves as a crucial metric in forecasting and prediction. Its significance



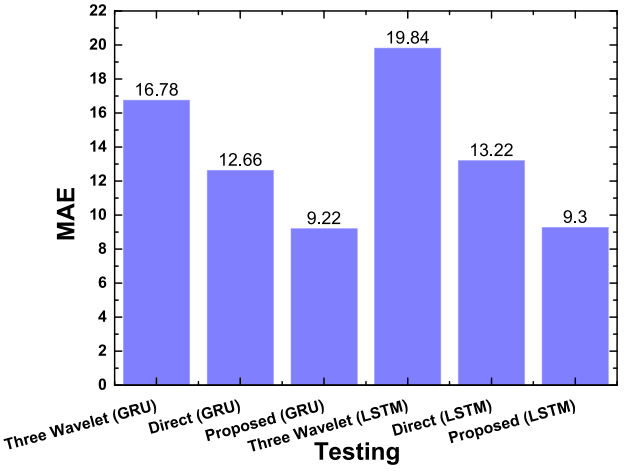
(a) RMSE of training data



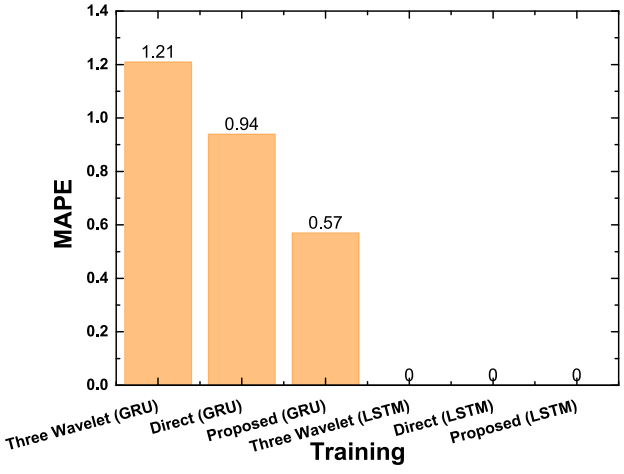
(b) RMSE of testing data



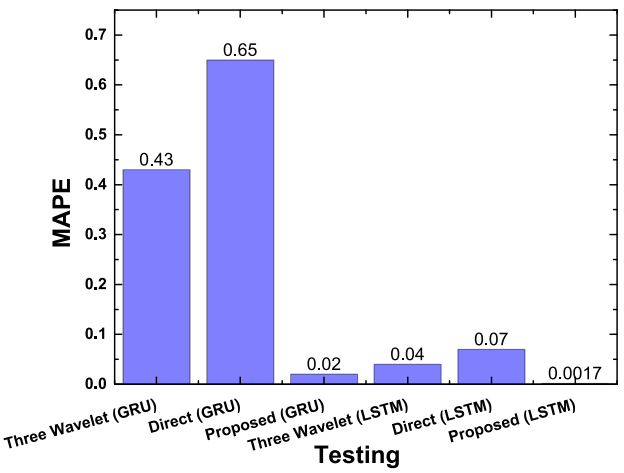
(c) MAE of training data



(d) MAE of testing data



(e) MAPE of training data

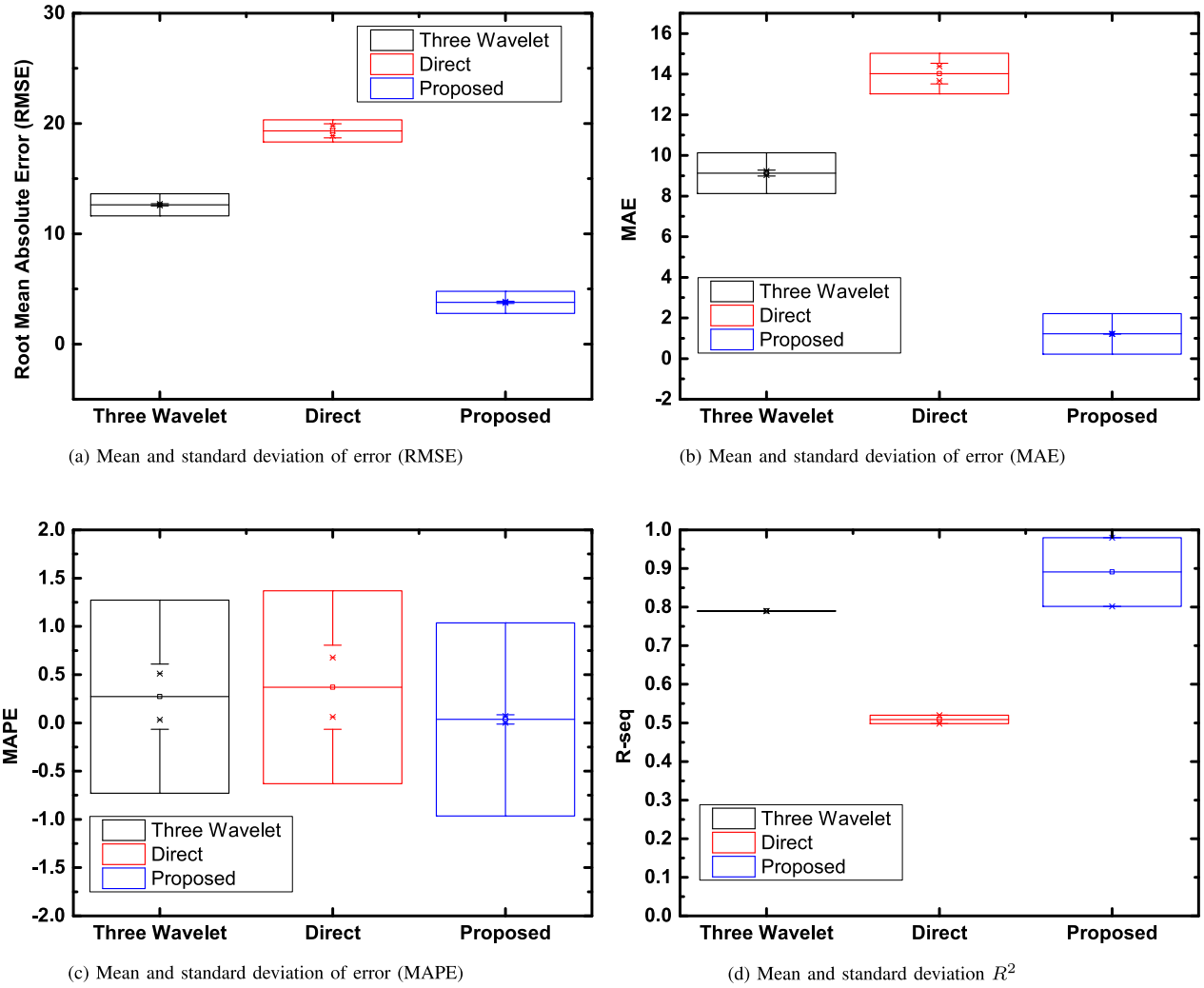


(f) MAPE of testing data

FIGURE 10. **Error graphs of training and testing data for GRU and LSTM networks.**

lies in its ability to offer a relative measure of error with different datasets and forecasting scenarios. MAPE facilitates easy comparisons and evaluations of forecasting performance by expressing the error as a percentage of the actual values.

One critical benefit of MAPE is its adaptability; in addition to indicating how well a model predicts, it closely resembles the loss functions defined relative to regression training sets. Consequently, the use of MAPE within machine learning allows for optimizing prediction accuracy through shrinkage



(a) Mean and standard deviation of error (RMSE)

(b) Mean and standard deviation of error (MAE)

(c) Mean and standard deviation of error (MAPE)

(d) Mean and standard deviation R^2

FIGURE 11. Graphs of errors and R^2 across GRU and LSTM.

property between estimated (output) and measured (true) values. The main focus of the MAPE algorithm is on appraising prediction techniques so as to connect the gap between theoretical projections and actual results; which makes it a compelling benchmark as well as an effective measure of accuracy. This approach drives precise forecasts that are reliable predictions across diverse domains.

Mathematically, MAPE can be expressed as,

$$MAPE = \frac{1}{n}\sum_{i}^{n}\left|\frac{y_i - x_i}{y_i}\right| \tag{10}$$

In Eqn. 10, y_i denotes the vector having actual values, x_i is the vector of predicted values and n denotes the total length of y_i or x_i.

R^2 performance measure has a significant impact on regression or prediction models' evaluation. It gauges the level of explanations given by the independent variable(s) in a model to variations of the dependent variable. In other words, R^2 measures the proportion of variability in the dependent variable that is attributable to the independent variable(s) in the model. R^2 is a measure that ranges from zero to one and indicates how well a model fits data. It provides a clear signal of how well a model fits data. Large values of R^2 signify a model that fits well while small values suggest poor fit. It is significant to note that R^2 is useful for measuring accuracy but not precision in terms of data prediction. In other words, it is usually used together with other evaluation metrics when appraising how well a model work.

The computation of R^2 is based on a mathematical framework that entails contrasting the variance of the original data set against that predicted by a specific model in terms of prediction accuracy. This comparison allows for the determination of the proportion of variability in the dependent variable that is accounted for by the independent variable(s) in the model. The mathematical framework of R^2 is as

$$R^2 = 1 - \frac{\sum_i^n (y_i - x_i)}{\sum_i^n (y_i - \hat{y})} \tag{11}$$

In Eqn. 11, y_i denotes the vector of actual values, x_i is vector of the predicted values, and $\hat{y}$ denotes the average over y.

TABLE 8. Dragon fly.

	Input Features						Outputs	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	21	246	526	350	424	398	165	1.5	6.8075	6.3075
R	0.9973	0.8632	0.9393	0.9882	0.9559	0.9771				
Indices	94	246	204	501	400	388	165	2	6.9737	6.307
R	0.9848	0.8632	0.9533	0.9821	0.9603	0.9760				
Indices	43	246	817	70	434	501	375	3	7.4562	6.8848
R	0.9748	0.8946	0.9591	0.9748	0.9487	0.9562				
Indices	119	39	626	474	300	155	954	4	7.7503	7.036
R	0.8758	0.8662	0.8762	0.8708	0.8649	0.8752				
Indices	119	246	144	501	501	501	954	5	7.9072	6.9072
R	0.8758	0.8155	0.8821	0.8534	0.8387	0.8546				
Indices	119	246	368	107	417	314	835	6	8.2232	6.4732
R	0.8771	0.8113	0.8907	0.8892	0.8703	0.8788				
Indices	55	208	506	185	281	49	368	7	8.1024	6.9596
R	0.9436	0.8876	0.9229	0.9464	0.9197	0.9441				
Indices	65	170	651	218	501	421	954	8	8.2493	6.9635
R	0.8781	0.8619	0.8540	0.8734	0.8387	0.8680				
Indices	6	242	99	425	172	406	954	9	8.6174	6.6174
R	0.8789	0.8231	0.8686	0.8725	0.8654	0.8635				
Indices	118	246	971	360	281	501	626	10	8.8061	6.9727
R	0.9327	0.8591	0.8676	0.9411	0.9245	0.9222				
Indices	119	225	899	501	402	237	837	11	9.2157	7.8824
R	0.9036	0.8681	0.8949	0.8148	0.8397	0.8486				
Indices	119	151	562	326	241	221	1012	12	9.1258	6.9591
R	0.7832	0.7644	0.7989	0.7426	0.7344	0.7477				
Indices	119	219	1012	501	434	251	837	13	9.1258	6.9591
R	0.9036	0.8727	0.9019	0.8148	0.8278	0.8602				
Indices	51	83	666	476	188	126	1012	14	9.5609	8.0053
R	0.7847	0.7704	0.7879	0.7001	0.7228	0.7398				
Indices	86	132	510	437	36	311	1012	15	9.5641	7.8975
R	0.7860	0.7673	0.7942	0.7303	0.7472	0.7379				

TABLE 9. Genetic algorithm.

	Input Features						Output	Parameters		
	Maximum Temperature	Minimum Temperature	Pan Evaporation	Relative Humidity 1	Relative Humidity 2	Average Relative Humidity	Wind Speed	λ	ξ	ξ'
Indices	118	242	1011	486	500	500	370	1.5	7.0095	6.6345
R	0.9217	0.9042	0.8147	0.9436	0.9154	0.9399				
Indices	89	246	1012	501	500	496	626	2	7.0672	6.6672
R	0.9444	0.8591	0.8538	0.9235	0.9017	0.9249				
Indices	75	46	606	310	288	500	370	3	7.2516	6.4516
R	0.9589	0.8446	0.9420	0.9519	0.9272	0.9399				
Indices	119	128	817	271	221	243	626	4	7.6943	6.9800
R	0.9332	0.8958	0.9413	0.9465	0.9406	0.9246				
Indices	88	118	773	288	368	381	954	5	7.8591	7.0020
R	0.8726	0.8377	0.8683	0.8722	0.8390	0.8695				
Indices	86	209	808	495	448	501	954	6	8.0471	7.0471
R	0.8829	0.8250	0.8597	0.8694	0.8520	0.8546				
Indices	116	241	757	495	481	433	954	7	8.1471	7.0042
R	0.8795	0.8295	0.8197	0.8694	0.8524	0.8611				
Indices	105	244	303	280	479	325	954	8	8.3401	7.0544
R	0.8789	0.8162	0.8732	0.8727	0.8315	0.8713				
Indices	117	244	501	496	500	497	954	9	8.4505	7.0220
R	0.8766	0.8162	0.8523	0.8606	0.8410	0.8591				
Indices	119	246	947	385	437	484	954	10	8.5838	7.0124
R	0.8758	0.8155	0.8547	0.8678	0.8596	0.8560				
Indices	49	135	844	369	453	357	954	11	8.7554	7.0412
R	0.8776	0.8679	0.7896	0.8739	0.8583	0.8734				
Indices	118	242	727	501	499	496	954	12	8.9075	7.0504
R	0.8740	0.8231	0.8668	0.8534	0.8348	0.8629				
Indices	115	246	947	241	500	500	954	13	8.9075	7.0504
R	0.8744	0.8155	0.8547	0.8664	0.8410	0.8567				
Indices	101	241	269	220	357	412	954	14	8.9614	6.9614
R	0.8811	0.8295	0.8692	0.8635	0.8665	0.8634				
Indices	89	209	616	64	496	426	954	15	9.0959	6.9530
R	0.8765	0.8250	0.8853	0.8823	0.8456	0.8662				

TABLE 10. Bird swarm optimization.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	79	242	998	494	454	290	626	1.5	6.8332	6.5332
R	0.9470	0.8835	0.9051	0.9204	0.9206	0.9366				
Indices	90	83	625	301	301	224	626	2	6.9369	6.5369
R	0.9381	0.8888	0.8932	0.9343	0.9278	0.9275				
Indices	88	206	623	291	255	100	625	3	7.5223	6.9508
R	0.9335	0.8587	0.8906	0.9467	0.9120	0.9496				
Indices	106	145	913	434	452	476	954	4	7.7092	6.9949
R	0.8757	0.8659	0.8463	0.8790	0.8547	0.8586				
Indices	115	224	819	465	480	468	954	5	7.8904	6.8904
R	0.8744	0.8252	0.8782	0.8716	0.8492	0.8504				
Indices	99	246	336	364	427	241	837	6	7.9554	6.9554
R	0.9056	0.8672	0.9089	0.8518	0.8294	0.8673				
Indices	110	180	829	422	487	481	954	7	8.1781	7.0353
R	0.8807	0.8596	0.8668	0.8558	0.8469	0.8668				
Indices	114	241	727	476	500	498	954	8	8.2891	6.4891
R	0.8717	0.8295	0.8668	0.8558	0.8410	0.8537				
Indices	60	119	919	204	402	141	626	9	8.4673	7.0388
R	0.9463	0.8865	0.9132	0.9372	0.9217	0.9458				
Indices	105	241	1007	499	492	291	954	10	8.8353	7.4603
R	0.8789	0.8295	0.8275	0.8534	0.8467	0.8816				
Indices	119	246	627	497	501	438	1002	11	8.7387	7.0244
R	0.8451	0.8188	0.8726	0.7537	0.7500	0.8002				
Indices	104	170	969	474	456	459	954	12	8.9668	7.1096
R	0.8740	0.8619	0.8182	0.8708	0.8479	0.8547				
Indices	119	246	1012	469	500	501	954	13	8.9668	7.1096
R	0.8758	0.8155	0.8120	0.8689	0.8410	0.8546				
Indices	117	245	1002	237	500	500	1002	14	9.2041	7.4541
R	0.8461	0.8151	0.8622	0.8077	0.7511	0.7599				
Indices	108	243	830	434	496	461	1002	15	9.2736	7.3986
R	0.8440	0.8201	0.8676	0.8041	0.7609	0.7789				

TABLE 11. Whales optimization algorithm.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	36	236	972	480	481	482	375	1.5	6.6665	6.2915
R	0.9729	0.9174	0.9582	0.9565	0.9565	0.9631				
Indices	104	216	706	345	427	361	375	2	6.7444	6.2444
R	0.9687	0.9574	0.9488	0.9728	0.9482	0.9699				
Indices	116	241	990	484	370	61	626	3	7.2854	6.2854
R	0.9361	0.8775	0.9028	0.9237	0.9249	0.9351				
Indices	96	199	819	395	405	405	375	4	7.6050	6.6050
R	0.9652	0.9256	0.9672	0.9671	0.9440	0.9601				
Indices	109	242	998	494	494	494	626	5	7.6917	6.4917
R	0.9427	0.8835	0.9051	0.9204	0.8855	0.9160				
Indices	88	169	742	354	365	342	626	6	7.9161	6.5161
R	0.9335	0.8796	0.9386	0.9370	0.9241	0.9357				
Indices	105	219	903	446	428	489	626	7	8.1960	7.0531
R	0.9363	0.8825	0.9312	0.9375	0.9084	0.9314				
Indices	119	246	1012	501	231	326	954	8	8.3464	7.0607
R	0.8758	0.8155	0.8120	0.8534	0.8736	0.8795				
Indices	117	235	1002	496	496	496	954	9	8.4417	7.0131
R	0.8766	0.8312	0.8241	0.8606	0.8456	0.8629				
Indices	114	207	971	413	421	361	954	10	8.6458	7.0744
R	0.8717	0.8301	0.8213	0.8798	0.8554	0.8787				
Indices	111	231	401	404	473	466	954	11	8.7629	7.0486
R	0.8780	0.8165	0.8486	0.8516	0.8248	0.8729				
Indices	117	243	999	495	495	156	954	12	9.0019	7.3769
R	0.8766	0.8190	0.8207	0.8694	0.8431	0.8833				
Indices	118	241	1002	495	241	496	1002	13	9.0019	7.3769
R	0.8452	0.8178	0.8622	0.7874	0.7931	0.7812				
Indices	112	241	954	468	453	471	954	14	9.0129	7.0129
R	0.8759	0.8295	0.8598	0.8505	0.8583	0.8608				
Indices	116	241	995	492	492	492	954	15	9.2144	7.0716
R	0.8795	0.8295	0.8235	0.8552	0.8467	0.8588				

TABLE 12. COOT.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	91	216	998	201	494	99	626	1.5	6.7987	6.4987
R	0.9352	0.9270	0.9051	0.9485	0.8855	0.9414				
Indices	117	223	208	443	311	346	956	2	7.7103	7.4246
R	0.8484	0.8181	0.8463	0.7548	0.7881	0.8013				
Indices	90	209	129	446	261	230	835	3	7.5587	6.7587
R	0.8829	0.8219	0.8924	0.8881	0.8761	0.8830				
Indices	56	58	746	172	419	274	627	4	7.4543	6.4543
R	0.9366	0.8872	0.9309	0.8935	0.8375	0.8501				
Indices	75	242	819	446	301	301	626	5	7.6560	6.4560
R	0.9425	0.8835	0.9485	0.9375	0.9278	0.9410				
Indices	119	195	614	260	497	501	626	6	8.0219	7.0190
R	0.9332	0.9312	0.9519	0.9318	0.9070	0.9222				
Indices	103	209	1012	401	156	392	954	7	8.0311	6.4311
R	0.8707	0.8250	0.8210	0.8732	0.8762	0.8747				
Indices	75	135	947	364	402	361	626	8	8.5903	7.4653
R	0.9425	0.9342	0.9182	0.9386	0.9217	0.9432				
Indices	118	206	242	207	500	335	1002	9	8.3516	6.9230
R	0.8452	0.8208	0.8312	0.8058	0.7511	0.7762				
Indices	90	153	692	349	291	181	954	10	8.7117	6.8783
R	0.8812	0.8378	0.8612	0.8715	0.8730	0.8767				
Indices	82	145	596	365	291	300	835	11	8.9703	7.4703
R	0.8887	0.8767	0.8309	0.8811	0.8764	0.8806				
Indices	45	80	762	332	470	215	998	12	9.4208	7.9640
R	0.8312	0.8053	0.8069	0.8076	0.7809	0.8060				
Indices	118	96	219	291	126	346	1012	13	9.4208	7.9640
R	0.7845	0.7715	0.7420	0.7462	0.7307	0.7412				
Indices	116	209	498	480	497	465	1002	14	9.1907	7.4407
R	0.8443	0.8266	0.8508	0.7888	0.7650	0.7716				
Indices	119	129	998	501	500	231	626	15	9.4930	6.4930
R	0.9332	0.9380	0.9051	0.9235	0.9017	0.9445				

TABLE 13. Jump spider optimization.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	119	246	1012	501	21	501	1012	1.5	8.2139	8.0473
R	0.7832	0.7590	0.7974	0.6939	0.7481	0.7038				
Indices	119	25	1012	501	501	501	1012	2	8.3487	8.1264
R	0.7832	0.7753	0.7974	0.6939	0.6927	0.7038				
Indices	82	231	929	398	406	437	954	3	8.5986	8.1541
R	0.8779	0.8165	0.8597	0.8794	0.8491	0.8658				
Indices	119	246	1012	501	501	501	1012	4	8.5894	8.0339
R	0.7832	0.7590	0.7974	0.6939	0.6927	0.7038				
Indices	119	236	1012	26	501	501	1012	5	7.9372	7.0800
R	0.7832	0.7652	0.7974	0.7510	0.6927	0.7038				
Indices	76	242	987	497	492	496	954	6	8.3916	7.3916
R	0.8805	0.8231	0.8203	0.8544	0.8467	0.8629				
Indices	91	139	884	398	382	495	957	7	9.0430	8.1541
R	0.8562	0.7904	0.8325	0.8203	0.7979	0.7782				
Indices	119	246	1012	501	501	501	1012	8	8.9269	7.9269
R	0.7832	0.7590	0.7974	0.6939	0.6927	0.7038				
Indices	84	246	821	227	311	365	1012	9	9.2652	8.1541
R	0.7849	0.7590	0.7962	0.7449	0.7272	0.7344				
Indices	119	246	1012	501	501	501	1012	10	9.2876	7.9126
R	0.7832	0.7590	0.7974	0.6939	0.6927	0.7038				
Indices	103	28	526	160	188	354	990	11	9.3361	8.0028
R	0.7653	0.7705	0.7362	0.7699	0.7542	0.7548				
Indices	119	246	1012	501	326	49	1012	12	8.9731	7.1159
R	0.7832	0.7590	0.7974	0.6939	0.7305	0.7439				
Indices	109	225	1009	496	480	498	954	13	8.9731	7.1159
R	0.8758	0.8212	0.8029	0.8606	0.8492	0.8537				
Indices	119	246	1012	501	100	501	1012	14	9.6060	8.0504
R	0.7832	0.7590	0.7974	0.6939	0.7463	0.7038				
Indices	119	246	1012	501	501	231	1012	15	9.7333	8.0667
R	0.7832	0.7590	0.7974	0.6939	0.6927	0.7499				

TABLE 14. COKOO-III.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	55	58	305	237	345	402	375	1.5	6.5721	6.1971
R	0.9721	0.9690	0.9705	0.9771	0.9555	0.9653				
Indices	119	123	340	218	246	346	375	2	6.6959	6.1959
R	0.9612	0.9664	0.9789	0.9718	0.9618	0.9705				
Indices	60	18	360	207	371	61	375	3	7.1832	6.3832
R	0.9746	0.9732	0.9838	0.9762	0.9564	0.9630				
Indices	46	129	591	277	271	454	626	4	7.3732	6.3732
R	0.9442	0.9380	0.9484	0.9448	0.9309	0.9338				
Indices	60	216	370	434	100	488	626	5	7.5749	6.3749
R	0.9463	0.9270	0.9587	0.9450	0.9446	0.9277				
Indices	94	164	616	36	437	382	626	6	7.8083	6.4083
R	0.9458	0.9298	0.9577	0.9510	0.9205	0.9434				
Indices	21	195	564	94	261	447	626	7	7.9928	6.3928
R	0.9462	0.9312	0.9340	0.9466	0.9296	0.9304				
Indices	119	64	616	271	342	482	626	8	8.1086	6.8229
R	0.9332	0.9406	0.9577	0.9465	0.9240	0.9301				
Indices	101	18	370	131	241	271	954	9	8.3002	6.8717
R	0.8811	0.8775	0.8894	0.8797	0.8714	0.8775				
Indices	35	64	601	499	100	421	954	10	8.4481	6.8767
R	0.8774	0.8748	0.8831	0.8534	0.8830	0.8680				
Indices	91	99	621	215	26	391	954	11	8.5637	6.8494
R	0.8781	0.8649	0.8696	0.8798	0.8775	0.8654				
Indices	89	64	601	462	438	64	954	12	8.6851	6.8279
R	0.8765	0.8748	0.8831	0.8781	0.8612	0.8826				
Indices	101	115	165	221	119	64	954	13	8.6851	6.8279
R	0.8811	0.8622	0.8913	0.8822	0.8737	0.8826				
Indices	49	18	370	434	191	326	954	14	8.8220	6.8220
R	0.8776	0.8775	0.8894	0.8790	0.8741	0.8795				
Indices	105	123	360	495	357	100	954	15	8.9948	6.8520
R	0.8786	0.8674	0.8870	0.8694	0.8665	0.8853				

TABLE 15. MAO.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	81	154	819	215	371	241	374	1.5	6.9998	6.6248
R	0.9274	0.8912	0.8963	0.9136	0.8906	0.9162				
Indices	79	22	155	172	231	84	626	2	6.7839	6.3839
R	0.9470	0.9410	0.9279	0.9462	0.9357	0.9416				
Indices	79	18	275	398	64	346	152	3	7.3191	6.3191
R	0.9376	0.9324	0.9235	0.9353	0.9292	0.9340				
Indices	33	145	360	85	240	326	370	4	7.8319	6.8319
R	0.9552	0.9429	0.9463	0.9598	0.9374	0.9560				
Indices	105	13	228	146	241	221	590	5	7.5579	6.3579
R	0.8842	0.8799	0.8750	0.8790	0.8694	0.8822				
Indices	43	58	817	221	221	141	626	6	8.1657	7.2907
R	0.9481	0.9378	0.9413	0.9487	0.9406	0.9458				
Indices	54	137	356	326	120	220	1002	7	8.2171	6.6171
R	0.8487	0.8304	0.8552	0.8065	0.7981	0.8026				
Indices	54	25	942	160	291	84	627	8	8.6586	7.3729
R	0.9371	0.9191	0.9302	0.8923	0.8734	0.8918				
Indices	91	204	415	85	120	49	953	9	8.6802	7.2517
R	0.8387	0.7998	0.8140	0.8126	0.8077	0.8110				
Indices	76	158	830	307	120	392	957	10	8.7939	7.2225
R	0.8556	0.7994	0.8671	0.8216	0.8121	0.8129				
Indices	51	82	823	215	64	357	957	11	9.0058	6.6058
R	0.8574	0.8342	0.8574	0.8221	0.8108	0.8056				
Indices	90	132	575	185	90	326	627	12	9.4285	7.2619
R	0.9350	0.9135	0.9380	0.8933	0.8833	0.8896				
Indices	91	58	914	332	190	241	834	13	9.4285	7.2619
R	0.8462	0.8129	0.8226	0.8276	0.8172	0.8318				
Indices	74	154	956	30	109	241	837	14	9.1367	6.8034
R	0.9076	0.8773	0.9200	0.8693	0.8537	0.8673				
Indices	91	218	568	104	90	8	952	15	9.4261	7.2832
R	0.8509	0.8245	0.8642	0.8053	0.7984	0.8039				

TABLE 16. Particle swarm optimization.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	119	93	1012	75	497	452	375	1.5	6.7330	6.3583
R	0.9612	0.9599	0.8724	0.9755	0.9374	0.9633				
Indices	119	195	819	487	454	438	375	2	6.7292	6.2292
R	0.9612	0.9628	0.9672	0.9689	0.9513	0.9680				
Indices	49	103	24	323	449	207	375	3	7.2021	6.2021
R	0.9749	0.9595	0.9693	0.9700	0.9435	0.9702				
Indices	66	180	616	487	326	326	375	4	7.4244	6.4244
R	0.9709	0.9571	0.9764	0.9689	0.9581	0.9735				
Indices	94	242	349	398	241	211	626	5	7.6240	6.4240
R	0.9458	0.8835	0.9522	0.9442	0.9370	0.9447				
Indices	104	235	616	221	438	361	626	6	7.8821	6.8821
R	0.9421	0.8923	0.9577	0.9487	0.9230	0.9432				
Indices	21	195	370	480	361	500	954	7	7.9758	6.8330
R	0.8811	0.8634	0.8894	0.8712	0.8699	0.8567				
Indices	110	129	601	291	361	221	954	8	8.1195	6.8338
R	0.8807	0.8705	0.8831	0.8811	0.8699	0.8834				
Indices	101	195	601	357	221	231	954	9	8.2986	6.8701
R	0.8811	0.8634	0.8831	0.8810	0.8789	0.8808				
Indices	54	170	601	462	438	291	954	10	8.4629	6.8915
R	0.8748	0.8619	0.8831	0.8781	0.8612	0.8816				
Indices	119	123	536	474	326	481	954	11	8.6090	6.8947
R	0.8758	0.8674	0.8727	0.8708	0.8704	0.8668				
Indices	110	164	954	462	454	291	954	12	8.7190	6.8619
R	0.8807	0.8630	0.8598	0.8781	0.8589	0.8816				
Indices	116	170	616	413	417	417	954	13	8.7190	6.8619
R	0.8795	0.8619	0.8853	0.8798	0.8638	0.8766				
Indices	76	64	616	462	326	241	954	14	8.8341	6.8341
R	0.8805	0.8748	0.8853	0.8781	0.8704	0.8788				
Indices	101	135	589	255	382	470	954	15	9.0241	6.8813
R	0.8811	0.8679	0.8762	0.8765	0.8663	0.8638				

TABLE 17. Fire fly algorithm.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	115	195	44	462	482	110	835	1.5	7.0383	6.7883
R	0.8844	0.8741	0.8964	0.8906	0.8626	0.8962				
Indices	15	242	591	462	417	382	626	2	6.8403	6.4403
R	0.9498	0.8835	0.9484	0.9432	0.9252	0.9434				
Indices	115	99	601	452	36	370	835	3	7.3591	6.3591
R	0.8844	0.8761	0.8891	0.8887	0.8944	0.8839				
Indices	119	68	954	332	438	241	370	4	7.4543	6.4543
R	0.9219	0.9465	0.9386	0.9559	0.9393	0.9600				
Indices	15	235	819	326	454	501	626	5	7.8165	6.8165
R	0.9498	0.8923	0.9485	0.9471	0.9206	0.9222				
Indices	118	198	1002	261	326	36	837	6	7.8618	6.8618
R	0.9050	0.8760	0.9157	0.8674	0.8487	0.8720				
Indices	116	195	616	474	382	291	954	7	8.0761	6.9333
R	0.8795	0.8634	0.8853	0.8708	0.8663	0.8816				
Indices	116	123	1007	237	438	417	954	8	8.3143	6.8143
R	0.8795	0.8674	0.8275	0.8826	0.8613	0.8766				
Indices	11	216	360	495	482	481	835	9	8.3038	6.8753
R	0.8939	0.8713	0.8918	0.8856	0.8626	0.8788				
Indices	76	135	626	495	402	100	954	10	8.4772	6.9058
R	0.8805	0.8680	0.8762	0.8694	0.8574	0.8853				
Indices	119	195	819	495	361	500	954	11	8.5706	6.8563
R	0.8758	0.8634	0.8782	0.8694	0.8699	0.8567				
Indices	116	22	614	398	489	221	954	12	8.7386	6.8814
R	0.8795	0.8750	0.8835	0.8794	0.8507	0.8834				
Indices	116	216	819	110	497	100	954	13	8.7386	6.8814
R	0.8795	0.8596	0.8782	0.8839	0.8464	0.8853				
Indices	105	135	580	382	366	241	954	14	8.8678	6.8678
R	0.8789	0.8680	0.8748	0.8793	0.8624	0.8788				
Indices	91	64	817	398	100	332	954	15	8.9864	6.8435
R	0.8781	0.8748	0.8751	0.8794	0.8830	0.8701				

TABLE 18. Harmony optimization.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	21	220	427	200	480	63	165	1.5	6.8283	6.3283
R	0.9973	0.9380	0.8479	0.9878	0.9489	0.9864				
Indices	91	58	594	230	431	331	358	2	7.3708	6.8708
R	0.8894	0.8903	0.8004	0.9042	0.8718	0.8922				
Indices	26	147	80	416	370	181	165	3	7.6401	6.9734
R	0.9878	0.8687	0.9474	0.9872	0.9763	0.9877				
Indices	51	73	706	279	277	459	824	4	8.1421	6.8921
R	0.8817	0.8404	0.8558	0.8599	0.8637	0.8620				
Indices	20	241	485	255	360	241	372	5	8.1246	6.6246
R	0.9295	0.8966	0.7929	0.8828	0.8510	0.8834				
Indices	112	134	411	21	155	15	374	6	8.4962	7.0962
R	0.9365	0.8882	0.8769	0.9175	0.9032	0.9146				
Indices	45	244	841	481	419	85	581	7	8.4168	7.0835
R	0.8753	0.7997	0.7907	0.8801	0.8472	0.8914				
Indices	86	209	918	131	174	494	817	8	8.2268	6.9411
R	0.8803	0.8193	0.8150	0.8777	0.8479	0.8466				
Indices	43	96	682	433	175	221	954	9	8.3208	6.8922
R	0.8793	0.8393	0.8595	0.8604	0.8661	0.8834				
Indices	104	198	154	156	70	191	954	10	8.4441	6.8727
R	0.8740	0.8360	0.8831	0.8808	0.8700	0.8812				
Indices	49	39	30	480	110	376	954	11	8.6213	6.9070
R	0.8776	0.8662	0.8778	0.8712	0.8805	0.8651				
Indices	100	18	255	21	210	451	954	12	8.7268	6.8697
R	0.8760	0.8775	0.8576	0.8809	0.8652	0.8555				
Indices	30	89	601	371	26	54	954	13	8.7268	6.8697
R	0.8774	0.8660	0.8831	0.8590	0.8775	0.8780				
Indices	76	137	289	58	291	109	1002	14	9.0479	7.2979
R	0.8478	0.8304	0.8583	0.8048	0.7935	0.8025				
Indices	36	22	44	413	425	376	954	15	9.0243	6.8814
R	0.8776	0.8750	0.8853	0.8798	0.8496	0.8651				

D. EXPERIMENTAL SETUP

Deep learning approaches of LSTM and GRU networks are implemented for experimental verification and evaluation of the proposed pre-processing technique with comparative methods. Two other methods are considered for comparison, one is the direct method which is without any pre-processing atmospheric data is fed into the deep learning models for training. The second method [55] involves sample entropy-based reconstruction of segregated IMFs of atmospheric variables. This technique results in three frequency-based signals that are low-frequency component, periodic component, and high-frequency component. These signals are trained separately on deep learning models to predict each component. In order to assess the overall wind speed, these three components are utilized to reconstruct a signal. However, more components introduce higher reconstruction error as well as more training and testing resources. This limitation is addressed by our method by minimizing the components to two sub-signals for reconstruction.

The deep learning approaches of LSTM and GRU utilize Adaptive Moment Estimation (ADAM) as the default optimizer. The mechanism of the ADAM's [56] updating is very similar to the RMSP, which is an optimization method commonly used for training deep learning networks. The model parameters updating of both methods are performed at each iteration through moving averages. ADAM, however, takes an additional step by adding a moment term in its computation. ADAM employs parameter gradients and their corresponding squared values for running average calculations at each element. This utilization of parameter gradients and their squared terms in overall calculation allows for finer-grained as well as dynamic changes in models' weights. These squared values enable the ADAM to adaptively adjust the learning rates for different parameters in the model. This can be advantageous in instances, where some parameters require faster updates while others don't need such rapid adjustments during training.

$$M_{i+1} = \beta_1 M_i + (1 - \beta_1)\nabla L(\theta_i) \tag{12}$$

$$v_{i+1} = \beta_2 v_i + (1 - \beta_2)[\nabla L(\theta_i)]^2 \tag{13}$$

$$\theta_{i+1} = \theta_i - \frac{lM_l}{\sqrt{u_i} + \varepsilon} \tag{14}$$

In Eqn. 12 and 13, β_1 and β_2 are the decay rates of gradients and their squares, respectively. Eqn. 14 illustrates the parameter updating procedure similar to RMSP, θ_i denotes training parameter i, and l represents the learning rate, which should be greater than zero. M describes the momentum while $\nabla L(\theta_i)$ represents the gradient of the loss function. Whereas, in Eqn. 14, ε is the small value for prevention if the square root value of v_i is zero.

The experimental analysis demonstrates the lead of the proposed method over the state-of-the-art deep learning models like LSTM and GRU. For the analysis purpose, the performance criterion of RMSE, MAE, MAPE, and R^2 are considered in the table 4. Further, Fig. 9 illustrates the

TABLE 19. Artificial bee colony.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	82	89	45	119	118	110	370	1.5	6.6822	6.3072
R	0.9600	0.9447	0.9446	0.9543	0.9444	0.9601				
Indices	52	103	515	116	109	90	165	2	6.7840	6.1173
R	0.9965	0.9765	0.9443	0.9929	0.9871	0.9896				
Indices	85	109	88	119	110	110	155	3	7.5257	6.1923
R	0.9594	0.9448	0.9368	0.9565	0.9570	0.9628				
Indices	49	54	60	100	110	110	163	4	9.3924	6.7258
R	0.9734	0.9677	0.9502	0.9794	0.9663	0.9772				
Indices	86	64	374	119	119	85	142	5	7.8859	6.8859
R	0.8999	0.8884	0.8679	0.9020	0.8948	0.9005				
Indices	88	18	35	43	110	119	837	6	8.4407	7.0407
R	0.9085	0.8625	0.8853	0.8688	0.8609	0.8450				
Indices	15	204	729	469	482	119	578	7	8.4237	6.4237
R	0.8763	0.8297	0.8332	0.8658	0.8444	0.8659				
Indices	110	103	109	90	119	119	368	8	9.2278	6.2278
R	0.9372	0.9265	0.9207	0.9492	0.9313	0.9399				
Indices	18	74	381	119	75	119	163	9	8.7418	7.3133
R	0.9746	0.9653	0.9323	0.9772	0.9621	0.9705				
Indices	56	22	30	90	26	116	957	10	9.0023	6.8023
R	0.8582	0.8104	0.8103	0.8225	0.8152	0.8082				
Indices	109	89	17	26	36	85	614	11	8.8117	6.8117
R	0.8811	0.8732	0.8722	0.8920	0.8852	0.8888				
Indices	86	74	115	119	119	100	824	12	9.4096	6.8096
R	0.8873	0.8763	0.8720	0.8822	0.8794	0.8880				
Indices	81	54	80	85	119	110	590	13	9.4096	6.8096
R	0.8873	0.8759	0.8793	0.8817	0.8802	0.8823				
Indices	45	34	99	207	21	100	954	14	8.8421	6.8421
R	0.8799	0.8679	0.8686	0.8803	0.8798	0.8853				
Indices	75	7	115	119	119	109	819	15	9.3471	6.8471
R	0.8825	0.8726	0.8723	0.8781	0.8696	0.8827				

prediction responses of each method. The proposed method achieved 3.72, 1.22, 1.53 $\times$ 10^{-3}, and 0.8 values of RMSE, MAE, MAPE, and R^2 for the LSTM network. While 3.85, 1.22, 0.07, and 0.98 values of RMSE, MAE, MAPE, and R^2 for the GRU network. The three-wavelet approach acquired 12.7, 9.03, 0.03, and 0.79 values of RMSE, MAE, MAPE, and R^2 for the LSTM network. On the GRU network, 12.56, 9.23, 0.51, and 0.79 values of RMSE, MAE, MAPE, and R^2 are achieved. The direct approach obtains 19.78, 14.38, 0.06, and 0.50 values of RMSE, MAE, MAPE, and R^2 for the LSTM network. While 18.89, 13.67, 0.68, and 0.52 values of RMSE, MAE, MAPE, and R^2 for the GRU network.

PSO is the algorithm that mimics the social behavior of individual birds in a swarm to find the best solution to a particular problem. Each swarm individual is called a particle, which represents a potential solution to the problem. Particles roam through the search space also known as possible solutions looking for the most suitable solution. The key advantage of PSO is its simplicity and straightforwardness to implement. It can work without gradient data, which makes it good for improvement issues without gradients or with poor computation properties. Given these properties of the PSO, the approach employed it to find hyper-parameters tuning of LSTM and GRU networks. Table 5 demonstrates the values of the optimized hyperparameters for both LSTM and GRU networks. Table 6 describes the computational resources utilized to train these networks. Table 7 provides the details of the architecture of both networks.

The experiment outcomes demonstrate enhanced performance as compared to the other methods for each performance measure. Fig. 9 illustrates the responses of these proposed and comparative techniques. The figure includes the first column of the GRU network while the second column illustrates the responses of the LSTM network. While the first row belongs to the direct approach, the second row shows the responses to the prediction of the three-wavelet method and the last row demonstrates the responses of the proposed method. The graphs evidently show the proposed technique's performance, which is far better than its counterpart techniques.

Fig. 10 illustrates the error graphs of both training and testing, separately. Overall, the training errors are for less than testing errors for the LSTM and GRU networks. Error analysis of the proposed method and comparative methods is provided in Fig. 11. The error and its corresponding standard deviation are considered for these methods across GRU and LSTM networks. The direct, three-wavelet, and proposed methods represent the error averages over both networks. The direct method demonstrates the maximum error in all error measures. Fig. 11(a) illustrates the averages and their corresponding standard deviations related to the mean square error. Fig. 11(b) represents the same mean average error. Fig. 11(c) demonstrates the means and their standard deviations for mean average percentage error. Lastly, the R^2 value demonstrates the best value for the proposed technique in Fig. 11(d). Overall, the proposed method, has less error and a higher R^2 value than its counterparts.

TABLE 20. COKOO-II.

	Input Features						Output	Parameters		
	Maximum Temperature	**Minimum Temperature**	**Pan Evaporation**	**Relative Humidity 1**	**Relative Humidity 2**	**Average Relative Humidity**	**Wind Speed**	λ	ξ	ξ'
Indices	56	189	338	414	352	335	165	1.5	6.6822	6.3072
R	0.9941	0.9377	0.9818	0.9887	0.9732	0.9851				
Indices	111	139	31	435	427	86	376	2	6.7840	6.1173
R	0.9624	0.9205	0.9740	0.9737	0.9482	0.9783				
Indices	115	14	616	501	221	488	376	3	7.5257	6.1923
R	0.9599	0.9728	0.9764	0.9557	0.9669	0.9608				
Indices	21	64	557	389	153	336	627	4	9.3924	6.7258
R	0.9462	0.9406	0.9420	0.9438	0.9212	0.9420				
Indices	76	146	893	291	222	252	627	5	7.8859	6.8859
R	0.9366	0.9339	0.8987	0.9467	0.9406	0.9408				
Indices	77	75	329	501	22	464	626	6	8.4407	7.0407
R	0.9366	0.9409	0.9460	0.9235	0.9423	0.9249				
Indices	70	129	858	227	490	241	627	7	8.4237	6.4237
R	0.9462	0.9380	0.9178	0.9475	0.9120	0.9455				
Indices	115	159	44	357	417	261	627	8	9.2278	6.2278
R	0.9417	0.9295	0.9521	0.9467	0.9223	0.9441				
Indices	102	115	820	191	120	417	954	9	8.7418	7.3133
R	0.8811	0.8622	0.8782	0.8800	0.8758	0.8676				
Indices	18	64	616	132	427	120	955	10	9.0023	6.8023
R	0.8799	0.8748	0.8853	0.8797	0.8533	0.8809				
Indices	119	123	599	100	311	361	954	11	8.8117	6.8117
R	0.8758	0.8674	0.8808	0.8820	0.8643	0.8787				
Indices	117	59	600	191	489	232	954	12	9.4096	6.8096
R	0.8795	0.8720	0.8808	0.8800	0.8507	0.8808				
Indices	111	140	368	205	85	221	955	13	9.4096	6.8096
R	0.8807	0.8418	0.8883	0.8704	0.8764	0.8834				
Indices	119	129	931	399	219	342	954	14	8.8421	6.8421
R	0.8758	0.8705	0.8657	0.8794	0.8633	0.8697				
Indices	119	135	857	405	9	141	955	15	9.3471	6.8471
R	0.8758	0.8679	0.8541	0.8702	0.8740	0.8791				

IV. CONCLUSION

Wind energy is a readily available and clean renewable energy source (RES) that could play a vital role in the sustainable growth of developing countries like Pakistan. However, wind energy generation forecasting is quite difficult due to the unpredictable nature of the wind speed. Management of energy utilization can become very efficient if knowledge about the wind blowing is available in advance. This work represents a framework of a deep learning-based approach to wind prediction, which involves a pre-processing step of feature selection using nature-inspired algorithms. Our proposed technique can speculate the wind speed in advance with higher accuracy than the other state-of-the-art techniques. The experiment is evaluated on available recorded atmospheric data from 2016 to 2020 at the Climate, Energy, and Water Research Institute (CEWRI), NARC, Islamabad, Pakistan. The results demonstrate the superiority of the proposed approach over other deep-learning methods. The proposed method achieved an RMSE of 3.72 for the LSTM network and an RMSE of 3.85 for the GRU network. The nearest performance of RMSE based on the LSTM network is 12.70, while for the GRU network, the RMSE is equal to 12.56. However, the time cost of the proposed technique is 5× more than the three-wavelet approach due to the computation requirements of optimized deep learning networks architecture. Our proposed method can be implemented in a field where it can effectively predict the wind speed by considering the other atmospheric variables. The computation time of our proposed architecture can be reduced by utilizing different the network cut and precision reduction of trained parameters such as pruning and quantization, which can be taken up as a future work.

APPENDIX A
TABLES

See Tables 8–20.

ACKNOWLEDGMENT

The authors would like to thank the Climate, Energy and Water Research Institute (CEWRI), NARC, Islamabad, Pakistan.

REFERENCES

[1] A. Qazi, F. Hussain, N. A. Rahim, G. Hardaker, D. Alghazzawi, K. Shaban, and K. Haruna, "Towards sustainable energy: A systematic review of renewable energy sources, technologies, and public opinions," *IEEE Access*, vol. 7, pp. 63837–63851, 2019.

[2] M. Amer and T. U. Daim, "Selection of renewable energy technologies for a developing county: A case of Pakistan," *Energy Sustain. Develop.*, vol. 15, no. 4, pp. 420–435, Dec. 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0973082611000767

[3] K. Ullah, M. A. Tunio, Z. Ullah, M. T. Ejaz, M. J. Anwar, M. Ahsan, and R. Tandon, "Ancillary services from wind and solar energy in modern power grids: A comprehensive review and simulation study," *J. Renew. Sustain. Energy*, vol. 16, no. 3, May 2024, Art. no. 032701.

[4] H. Holttinen, "Estimating the impacts of wind power on power systems—Summary of IEA wind collaboration," *Environ. Res. Lett.*, vol. 3, no. 2, Apr. 2008, Art. no. 025001, doi: 10.1088/1748-9326/3/2/025001.

[5] I. Hussain, A. Haider, Z. Ullah, M. Russo, G. M. Casolino, and B. Azeem, "Comparative analysis of eight numerical methods using Weibull distribution to estimate wind power density for coastal areas in Pakistan," *Energies*, vol. 16, no. 3, p. 1515, Feb. 2023.

[6] S. Saeed, R. Asghar, F. Mehmood, H. Saleem, B. Azeem, and Z. Ullah, "Evaluating a hybrid circuit topology for fault-ride through in DFIG-based wind turbines," *Sensors*, vol. 22, no. 23, p. 9314, Nov. 2022.

[7] K. Ullah, Z. Ullah, A. Basit, and G. Gruosso, "Optimal utilization of frequency ancillary services in modern power systems," *e-Prime-Adv. Electr. Eng., Electron. Energy*, vol. 9, Sep. 2024, Art. no. 100755.

[8] J. Lerner, M. Grundmeyer, and M. Garvert, "The importance of wind forecasting," *Renew. Energy Focus*, vol. 10, no. 2, pp. 64–66, Mar. 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1755008409700924

[9] R. Asghar, F. Rehman, Z. Ullah, A. Aman, K. Iqbal, and A. Ali Nawaz, "Modified switch type fault current limiter for low-voltage ride-through enhancement and reactive power support of DFIG-WT under grid faults," *IET Renew. Power Gener.*, vol. 14, no. 9, pp. 1481–1490, Jul. 2020.

[10] I. Colak, G. Fulli, S. Bayhan, S. Chondrogiannis, and S. Demirbas, "Critical aspects of wind energy systems in smart grid applications," *Renew. Sustain. Energy Rev.*, vol. 52, pp. 155–171, Dec. 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032115007091

[11] C. A. Lopez-Villalobos, O. Martínez-Alvarado, O. Rodriguez-Hernandez, and R. Romero-Centeno, "Analysis of the influence of the wind speed profile on wind power production," *Energy Rep.*, vol. 8, pp. 8079–8092, Nov. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352484722011854

[12] M. D. Sabir, K. Hafeez, S. Batool, G. Akbar, L. Khan, G. Hafeez, and Z. Ullah, "Prediction of solar PV power using deep learning with correlation-based signal synthesis," *IEEE Access*, vol. 12, pp. 40736–40751, 2024.

[13] K. Ullah, Z. Ullah, S. Aslam, M. S. Salam, M. A. Salahuddin, M. F. Umer, M. Humayon, and H. Shaheer, "Wind farms and flexible loads contribution in automatic generation control: An extensive review and simulation," *Energies*, vol. 16, no. 14, p. 5498, Jul. 2023.

[14] H. Liu, X. Mi, Y. Li, Z. Duan, and Y. Xu, "Smart wind speed deep learning based multi-step forecasting model using singular spectrum analysis, convolutional gated recurrent unit network and support vector regression," *Renew. Energy*, vol. 143, pp. 842–854, Dec. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960148119306998

[15] K. Wang, X. Qi, H. Liu, and J. Song, "Deep belief network based k-means cluster approach for short-term wind power forecasting," *Energy*, vol. 165, pp. 840–852, Dec. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544218318826

[16] C. Yildiz, H. Acikgoz, D. Korkmaz, and U. Budak, "An improved residual-based convolutional neural network for very short-term wind power forecasting," *Energy Convers. Manage.*, vol. 228, Jan. 2021, Art. no. 113731. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0196890420312553

[17] Z. Zhang, L. Ye, H. Qin, Y. Liu, C. Wang, X. Yu, X. Yin, and J. Li, "Wind speed prediction method using shared weight long short-term memory network and Gaussian process regression," *Appl. Energy*, vol. 247, pp. 270–284, Aug. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261919306932

[18] X. Liu, H. Zhang, X. Kong, and K. Y. Lee, "Wind speed forecasting using deep neural network with feature selection," *Neurocomputing*, vol. 397, pp. 393–403, Jul. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231220304148

[19] G. Xu and L. Xia, "Short-term prediction of wind power based on adaptive LSTM," in *Proc. 2nd IEEE Conf. Energy Internet Energy Syst. Integr. (EI)*, Oct. 2018, pp. 1–5.

[20] Y. Huang, S. Liu, and L. Yang, "Wind speed forecasting method using EEMD and the combination forecasting method based on GPR and LSTM," *Sustainability*, vol. 10, no. 10, p. 3693, Oct. 2018. [Online]. Available: https://www.mdpi.com/2071-1050/10/10/3693

[21] K. Lu, W. X. Sun, X. Wang, X. R. Meng, Y. Zhai, H. H. Li, and R. G. Zhang, "Short-term wind power prediction model based on encoder–decoder LSTM," *IOP Conf. Ser., Earth Environ. Sci.*, vol. 186, Oct. 2018, Art. no. 012020, doi: 10.1088/1755-1315/186/5/012020.

[22] Z. Niu, Z. Yu, W. Tang, Q. Wu, and M. Reformat, "Wind power forecasting using attention-based gated recurrent unit network," *Energy*, vol. 196, Apr. 2020, Art. no. 117081. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544220301882

[23] K. Shivam, J.-C. Tzou, and S.-C. Wu, "Multi-step short-term wind speed prediction using a residual dilated causal convolutional network with nonlinear attention," *Energies*, vol. 13, no. 7, p. 1772, Apr. 2020. [Online]. Available: https://www.mdpi.com/1996-1073/13/7/1772

[24] X. Guo, C. Zhu, J. Hao, and S. Zhang, "Multi-step wind speed prediction based on an improved multi-objective seagull optimization algorithm and a multi-kernel extreme learning machine," *Int. J. Speech Technol.*, vol. 53, no. 13, pp. 16445–16472, Jul. 2023, doi: 10.1007/s10489-022-04312-7.

[25] F. Theuer, M. F. van Dooren, L. von Bremen, and M. Kühn, "LiDAR-based minute-scale offshore wind speed forecasts analysed under different atmospheric conditions," *Meteorologische Zeitschrift*, vol. 31, pp. 13–29, Oct. 2021. [Online]. Available: https://elib.dlr.de/147104/

[26] X. Luo and D. Zhang, "An adaptive deep learning framework for day-ahead forecasting of photovoltaic power generation," *Sustain. Energy Technol. Assessments*, vol. 52, Aug. 2022, Art. no. 102326. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2213138822003782

[27] B. B. Hazarika, D. Gupta, and N. Natarajan, "Wavelet kernel least square twin support vector regression for wind speed prediction," *Environ. Sci. Pollut. Res.*, vol. 29, no. 57, pp. 86320–86336, Dec. 2022.

[28] N. Amjady, F. Keynia, and H. Zareipour, "Wind power prediction by a new forecast engine composed of modified hybrid neural network and enhanced particle swarm optimization," *IEEE Trans. Sustain. Energy*, vol. 2, no. 3, pp. 265–276, Jul. 2011.

[29] M. Ahmadi and M. Khashei, "Current status of hybrid structures in wind forecasting," *Eng. Appl. Artif. Intell.*, vol. 99, Mar. 2021, Art. no. 104133. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197620303705

[30] L. Zhang, J. Wang, and X. Niu, "Wind speed prediction system based on data pre-processing strategy and multi-objective dragonfly optimization algorithm," *Sustain. Energy Technol. Assessments*, vol. 47, Oct. 2021, Art. no. 101346. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2213138821003568

[31] H. Liu and Z. Zhang, "Development and trending of deep learning methods for wind power predictions," *Artif. Intell. Rev.*, vol. 57, no. 5, p. 112, Apr. 2024, doi: 10.1007/s10462-024-10728-z.

[32] D. Wei, J. Wang, X. Niu, and Z. Li, "Wind speed forecasting system based on gated recurrent units and convolutional spiking neural networks," *Appl. Energy*, vol. 292, Jun. 2021, Art. no. 116842. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261921003378

[33] Z. Sun and M. Zhao, "Short-term wind power forecasting based on VMD decomposition, ConvLSTM networks and error analysis," *IEEE Access*, vol. 8, pp. 134422–134434, 2020.

[34] A. Forestiero, "Bio-inspired algorithm for outliers detection," *Multimedia Tools Appl.*, vol. 76, no. 24, pp. 25659–25677, Dec. 2017, doi: 10.1007/s11042-017-4443-1.

[35] N. Shaukat, A. Ahmad, B. Mohsin, R. Khan, S. U.-D. Khan, and S. U.-D. Khan, "Multiobjective core reloading pattern optimization of PARR-1 using modified genetic algorithm coupled with Monte Carlo methods," *Sci. Technol. Nucl. Installations*, vol. 2021, pp. 1–13, Apr. 2021, doi: 10.1155/2021/1802492.

[36] L. Abualigah, M. A. Elaziz, N. Khodadadi, A. Forestiero, H. Jia, and A. H. Gandomi, *Aquila Optimizer Based PSO Swarm Intelligence for IoT Task Scheduling Application in Cloud Computing*. Cham, Switzerland: Springer, 2022, pp. 481–497, doi: 10.1007/978-3-030-99079-4_19.

[37] D. KaraboÃŸa. (2005). *An Idea Based on Honey Bee Swarm for Numerical Optimization*. [Online]. Available: https://api.semanticscholar.org/CorpusID

[38] A. H. Gandomi and A. H. Alavi, "Krill herd: A new bio-inspired optimization algorithm," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 17, no. 12, pp. 4831–4845, Dec. 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1007570412002171

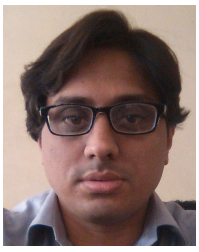
[39] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," *Neurocomputing*, vol. 172, pp. 371–381, Jan. 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231215010504

[40] J. Wang, T. Niu, H. Lu, W. Yang, and P. Du, "A novel framework of reservoir computing for deterministic and probabilistic wind power forecasting," *IEEE Trans. Sustain. Energy*, vol. 11, no. 1, pp. 337–349, Jan. 2020.

[41] S. Buhan, Y. Özkazanç, and I. Çadirci, "Wind pattern recognition and reference wind mast data correlations with NWP for improved wind-electric power forecasts," *IEEE Trans. Ind. Informat.*, vol. 12, no. 3, pp. 991–1004, Jun. 2016.

[42] M. Nazari-Heris, B. Mohammadi-Ivatloo, and G. B. Gharehpetian, "Short-term scheduling of hydro-based power plants considering application of heuristic algorithms: A comprehensive review," *Renew. Sustain. Energy Rev.*, vol. 74, pp. 116–129, Jul. 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032117302605

[43] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," *Proc. Roy. Soc. London. Ser. A, Math., Phys. Eng. Sci.*, vol. 454, no. 1971, pp. 903–995, Mar. 1998. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rspa.1998.0193
[44] Z. Wu and N. E. Huang, "Ensemble empirical mode decomposition: A noise-assisted data analysis method," *Adv. Adapt. Data Anal.*, vol. 1, no. 1, pp. 1–41, Jan. 2009, doi: 10.1142/s1793536909000047.
[45] M. E. Torres, M. A. Colominas, G. Schlotthauer, and P. Flandrin, "A complete ensemble empirical mode decomposition with adaptive noise," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 4144–4147.
[46] N. Rehman and D. P. Mandic, "Multivariate empirical mode decomposition," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 466, no. 2117, pp. 1291–1302, 2010. [Online]. Available: https://api.semanticscholar.org/CorpusID:1658732
[47] G. Rilling, P. Flandrin, P. Goncalves, and J. M. Lilly, "Bivariate empirical mode decomposition," *IEEE Signal Process. Lett.*, vol. 14, no. 12, pp. 936–939, Dec. 2007.
[48] D. Iatsenko, P. V. E. McClintock, and A. Stefanovska, "Nonlinear mode decomposition: A noise-robust, adaptive decomposition method," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 92, no. 3, Sep. 2015, Art. no. 032916.
[49] F. Rodríguez, F. Martín, L. Fontán, and A. Galarza, "Ensemble of machine learning and spatiotemporal parameters to forecast very short-term solar irradiation to compute photovoltaic generators' output power," *Energy*, vol. 229, Aug. 2021, Art. no. 120647. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544221008963
[50] L. Li, Y.-Q. Liu, Y.-P. Yang, S. Han, and Y.-M. Wang, "A physical approach of the short-term wind power prediction based on CFD pre-calculated flow fields," *J. Hydrodynamics*, vol. 25, no. 1, pp. 56–61, Feb. 2013, doi: 10.1016/s1001-6058(13)60338-8.
[51] Y. Yu, X. Han, M. Yang, and J. Yang, "Probabilistic prediction of regional wind power based on spatiotemporal quantile regression," in *Proc. IEEE Ind. Appl. Soc. Annu. Meeting*, Sep. 2019, pp. 1–16.
[52] S. Zolfaghari, G. H. Riahy, and M. Abedi, "A new method to adequate assessment of wind farms' power output," *Energy Convers. Manage.*, vol. 103, pp. 585–604, Oct. 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0196890415006524
[53] B. K. Saxena, S. Mishra, and K. V. S. Rao, "Offshore wind speed forecasting at different heights by using ensemble empirical mode decomposition and deep learning models," *Appl. Ocean Res.*, vol. 117, Dec. 2021, Art. no. 102937. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0141118721004041
[54] D. Alves, F. Mendonça, S. S. Mostafa, and F. Morgado-Dias, "The potential of machine learning for wind speed and direction short-term forecasting: A systematic review," *Computers*, vol. 12, no. 10, p. 206, Oct. 2023. [Online]. Available: https://www.mdpi.com/2073-431X/12/10/206
[55] Z. Li, R. Xu, X. Luo, X. Cao, S. Du, and H. Sun, "Short-term photovoltaic power prediction based on modal reconstruction and hybrid deep learning model," *Energy Rep.*, vol. 8, pp. 9919–9932, Nov. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S235248472201441X
[56] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent.*, San Diego, CA, USA, Y. Bengio and Y. LeCun, Eds., 2015, pp. 1–13.



MUHAMMAD DILSHAD SABIR received the Graduate degree in computer engineering from COMSATS University Islamabad and the M.S. and Ph.D. degrees in CE from the National University of Sciences and Technology, Islamabad, in 2012 and 2022, respectively. He currently holds a faculty member position of COMSATS University Islamabad. His research interests include the Internet of Things (IoT), deep learning, correlation pattern recognition, convolution neural networks, and network security.

LAIQ KHAN received the B.Sc. degree (Hons.) in electrical engineering from Khyber Pakhtoon Kwah University of Engineering and Technology, Peshawar, Pakistan, in 1996, and the M.S.-leading Ph.D. degrees in power system dynamics and control from the University of Strathclyde, Glasgow, U.K., in 2003. Before the Ph.D. degree, he was with Siemens, Pakistan, as a Field Engineer for two years. He was an Assistant Professor with the Faculty of Electronic Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Swabi, Pakistan, until 2008. Then, he joined the Faculty of Electrical Engineering, COMSATS University Islamabad, Abbottabad Campus, Pakistan, as an Associate Professor. He was also a Professor with Islamic University Madinah, Saudi Arabia, for three years. Currently, he is a Professor in power system dynamics and control with COMSATS University Islamabad, Islamabad Campus. He has published more than a 100 publications in highly reputable international conferences and peer-reviewed impact factor journals. His research interests include power system stability and control (FACTS controllers and HVDC), robust control theory, intelligent control systems, nonlinear control of WECS and photovoltaic systems, distributed MPC control of microgrid, hybrid electric storage systems for EVs, machine learning/deep learning applications in microgrids/smart grids, and solar cell synthesis & characterization.

KAMRAN HAFEEZ received the M.S. degree in electrical power engineering from UET Peshawar, in 2008, and the Ph.D. degree in electrical engineering from COMSATS University Islamabad, in 2020. He is currently an Assistant Professor with the ECE Department, COMSATS University Islamabad. His research interests include power converters, HVDC systems, and electric power systems.

ZAHID ULLAH (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering from UET Peshawar, in 2014, and the M.S. degree in electrical engineering from COMSATS University Islamabad, Abbottabad Campus, Abbottabad, Pakistan, in 2017. He is currently pursuing the Ph.D. degree in electrical engineering with the Politecnico di Milano, Italy. He was a Lecturer with UMT Lahore, Pakistan, from 2017 to 2020. He has published various articles in reputed journals and conference proceedings. His research interests include machine learning, deep learning, digital twins, smart grid, renewable energy systems, and ICTs.

STANISLAW CZAPP (Member, IEEE) received the M.S. degree from Gdansk University of Technology, Poland, in 1996, and the Ph.D. and D.Sc. degrees, in 2002 and 2010, respectively. He is currently a Professor with the Faculty of Electrical and Control Engineering, Gdansk University of Technology. He is the author or co-author of many articles, conference papers, and unpublished studies, such as designs and expert evaluations and opinions. He is an Expert of the SEP Association of Polish Electrical Engineers in Section Eight Electrical Installations and Devices. His research and teaching interests include power systems, electrical installations and devices, electric lighting, and electrical safety.

Open Access funding provided by 'Politecnico di Milano' within the CRUI CARE Agreement