Research Article

Hybrid Models for Weather Parameter Forecasting

Rashmi Bhardwaj and **Varsha Duhoon**

USBAS, Nonlinear Dynamics Research Lab, Guru Gobind Singh Indraprastha University, Sec-16C, Dwarka,
New Delhi 10078, India

Correspondence should be addressed to Rashmi Bhardwaj; rashmib22@gmail.com

Received 30 May 2021; Accepted 29 October 2021; Published 25 November 2021

Academic Editor: Roberto Natella

Copyright © 2021 Rashmi Bhardwaj and Varsha Duhoon. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The objective of the paper is to compare hybrid conjunction models with conventional models for the reduction of errors in weather forecasting. Besides the simple models like RBF model, SMO model, and LibSVM model, different hybrid conjunction models have been used for forecasting under different schemes. The forecasts from these models are further compared on the basis of errors calculated and time taken by the hybrid models and simple models in order to forecast weather parameters. In this paper, conjunction models over the conventional models are designed for forecasting the weather parameters for the reduction of error. India is a tropical country with variations in weather conditions. The objective is to build a conjunction model with less error to forecast weather parameters. A hybrid conjunction model is developed and analysed for different weather parameters for different metropolitan cities of India. Performance measurement is analysed for weather parameters. It is observed that, on the basis of error comparison and time taken by the models, the hybrid wavelet-neuro-RBF model gives better results as compared to the other models due to lower values of determined errors, better performance, and lesser time taken. The study becomes significant as weather forecasting with accuracy is a complex task along with the reduction of prediction error by the application of different models and schemes. It is concluded that the proposed hybrid model is helpful for forecasting and making policies in advance for the betterment of the human being, farmers, tourists, and so on as in all these activities, weather forecast plays an important role.

1. Introduction

The forecasting of weather forecasting is an important task as the economy of a country, especially the one which is more agriculture-based, depends on the agriculture yield, which further is dependent on weather parameters. There have been constant efforts in the area of weather forecasting to increase prediction accuracy.

The basic schematic of application of simple and hybrid models is explained through a flowchart in Figure 1.

In the recent past, research work has been significantly carried out on the time-series analysis of weather parameters. Bhardwaj and Duhoon [1] studied the persistence behavior of time series of monthly rainfall and temperature for India by applying dispersion analysis on the weather dataset from 1901 to 2015. Similarly, for ecological data like that of wildlife poaching, the persistence behavior of studied time series was assessed by using dispersion analysis in [2].

Furthermore, in [3] using time-series analysis, the change in the predictability of the temperature was studied for the factors which may lead to unpredictable or antipersistent behavior in time series leading to an increase in death rates due to random temperature fluctuations under phenomena of heatstroke. Antipersistence behavior was observed by the authors in the studied data, which is not a good indicator in the direction of lowering of temperature levels despite global initiatives like the Paris Climate Agreement 2015 being taken [3].

Different soft computing techniques to forecast weather parameters have been studied in [4]. In [5], it has been demonstrated that, on integrating ANFIS (Adaptive Neuro-Fuzzy Inference System) with Sugeno model and applying the resultant ANFIS-Sugeno model on weather time-series data like daily temperature, the error in the output gets reduced. Further in [6], single NARX (Nonlinear Autoregressive Exogenous) model and conjunction model

FIGURE 1: Flowchart for modeling of problems.

W-NARX (Wavelet-NARX) were studied by Bhardwaj and Duhoon for temperature data time series, and on comparing the output results, it was concluded that W-NARX showed better efficiency in learning the behavior from the input parameters with least MSE (Mean Square Error) of Training, Validating, and Testing as $8.12376e-1$, $4.86326e-1$, and $4.79787e-1$, respectively. The R values for Training, Validating, and Testing obtained were $9.88728e-1$, $9.26185e-1$, and $9.94526e-1$, respectively.

Various machine learning strategies for temperature forecasting were studied by Cifuentes et al. [7], where the machine learning techniques proved to be better for accurate prediction. In comparison with traditional artificial neural networks, the deep learning strategies reported small errors. Both clustering and classification techniques using weather parameters have been studied. Naive Bayes provides better results as compared to others on the basis of statistical outcomes of Kappa Statistics and estimated errors. In [8], K-means clustering, EM clustering, hierarchical clustering were used as the clustering methods, and on the basis of the least time taken, it was concluded that K-means clustering is an efficient clustering technique [8].

For a monthly forecast of precipitation, Kalteh [9] studied artificial neural networks and conjunction of ANN (Artificial Neural Network) and singular spectrum analysis models. The conjunction model so formed was compared with the single ANN model on the basis of error estimation by calculating RMSE (Root Mean Square Error) and coefficient of efficiency. The results showed that the conjunction

model performed way superior to the single ANN model. Further conjunction model of wavelet and SVM (Support Vector Machine) was studied by Kisi and Cimen [10] for forecasting daily precipitation where the discrete wavelet and SVM were combined to form a conjunction model. A single SVM was compared, and the conjunction model was observed to give the best results on the basis of error calculation.

On the other hand, Oana and Spataru [11] studied the application of genetic algorithms in conjunction with WRF numerical weather prediction systems for optimizing and forecasting using GA. It was observed that the conjunction model was efficient and showed less error. Comparison of ARIMA (Autoregressive Integrated Moving Average) with exponential smoothing models, which is Holt-Winters model and ETS model (i.e., Error, Trend, and Seasonality model), was carried out by Guizzi et al. [12] in order to forecast weather parameters: temperature, pressure, and humidity for Italy for one month. The Holts-Winters model gave better results as compared to the other two models. For weather parameter data of Saudi Arabia, individual MLP (Multiple Layer Perceptron) and RBF (Radial Basis Function) models were studied by Saba et al. [13]. They compared the obtained results with the results of the hybrid neural model, which was a combination of MLP and RBF, in order to improve the forecast. The RMSE, correlation coefficient, and scatter index were compared in order to see which model performed better with lesser error. It was observed that the hybrid model showed better results as compared to

individual MLP and RBF models. Sepideh et al. [14] discussed Neuro-Fuzzy and Wavelet Neuro-Fuzzy conjunction models for short- and long-term forecast of air temperature. The comparison of ANFIS and WANFIS showed that conjunction models performed better than the usual ANFIS model. The comparison was conducted by comparing the coefficient of determination and RMSE.

Reddy and Jung predicted load using wavelet and ANN [15]. Shrimohammadi et al. applied the wavelet-ANFIS model to forecast drought [16]. Turgay et al. applied the wavelet neural network to predict precipitation [17]. Piasecki et al. predicted water level using wavelet ANN [18]. Khan et al. studied the comparison among wavelet ANN and ANN models [19]. Karami et al. studied and applied the ANN model to predict whether data [20]. Kanna et al. applied an adaptive neural network for predicting wind power [21]. Adamowski et al. forecasted river flow using wavelet and neural networks [22]. Araghi et al. studied a wavelet-based hybrid model [23]. Liu et al. analysed wavelet and ANN for rainfall prediction [24].

Based on the above-mentioned studies [25], in this paper, we attempt to form a new model which can forecast weather parameters and take lesser time in forecasting. The daily data of weather parameters, namely, maximum temperature, minimum temperature, rainfall, and wind speed, for Delhi and Mumbai from January 1, 2017, up till May 30, 2018, have been considered, and for Chennai, the daily data from January 1, 2020, up till February 28, 2021, have been studied in order to study the independent data sets for daily weather parameters. The objective of the study is to study the different metropolitan cities at different locations on the map of the country. Delhi, the capital of India, faces all kinds of seasons. Mumbai is near the sea and has a lot of moisture present there, and it has limited seasons. Chennai all together is at the different side and faces limited seasons. These states are dense in terms of population as many people from rural and urban areas come together for their living. Hence, in order to study the independent data sets for daily weather parameters, these states have been taken.

To the considered data plain, LibSVM (Library Support Vector Machine), RBF, and SMO (Sequential Minimal Optimization) methods are applied. Next, the different hybrid conjunction models are studied under different schemes in which the time series of weather parameters are denoised using Haar wavelet and trained using NARX before obtaining the forecasts for the forecasting period. The study includes (1) wavelet-RBF, wavelet-SMO, and wavelet-LibSVM, (2) Neuro-RBF, Neuro-SMO, and Neuro-LibSVM, and (3) wavelet-neuro-RBF, wavelet-neuro-SMO, and wavelet-neuro-LibSVM. The outputs are compared, and on the basis of the errors and time taken in seconds by the models, the most suitable model is chosen. Weka software has been used for the study.

2. Data Set

The climate in Delhi is a combination of monsoon-influenced humid subtropical and semiarid climate, having a variation between summer and winter temperatures and

precipitation. Summer starts in early April and peaks in late May further to early June, showing an average temperature near 38°C, and at times it can rise to 45°C. Winter on the other hand starts in November and peaks in January, showing an average temperature of 6–7°C. Delhi's proximity to the Himalayas results in cold waves leading to a lower temperature. The extreme temperature has ranged from –2.2 to 48.4°C.

The climate of Mumbai is a tropical, wet, and dry climate. The climatic condition of Mumbai is moderately hot with a high level of humidity. Its coastal nature and tropical location ensure that temperatures do not fluctuate much through the year. Mumbai experiences three distinct seasons: (1) winter (October–February) temperature: 15 to 20°C, (2) summer (March–May) temperature: 27–30°C, and (3) monsoon (June–September) temperature: 24–29°C. The extreme temperature ranges from 14 to 30°C.

The climate of Chennai is a tropical wet and dry climate. Chennai lies on the thermal equator and is also coastal, which prevents extreme variation in seasonal temperature. The weather in Chennai is hot and humid. The hottest part of the year is May and early June, with a maximum temperature of 38–42°C. The coolest part of the year is January, with a minimum temperature of 18–20°C. The extreme temperature ranges from 13.9 to 45°C.

Based on the data in Table 1, the model developed in the study is validated for the above data points.

3. Methodology

3.1. Wavelet Method. Wavelet transformation is used for denoising signals, which means regenerating or reconstructing a signal from the noisy one. Wavelet transformation applies interdependent analysis in order to match the output with the input signals. Now, if the charge of a signal is filled in a small wavelets dimension, its coefficient will be comparatively larger in comparison to disturbances; hence, energy spreads over a large number of coefficients. Further, reducing wavelets transformation will eliminate the low amplitude disturbance or signals which are not required in the wavelet domain.

The Haar wavelet's mother wavelet function $\varnothing(t)$ can be written as

$$\varnothing(t) = \begin{cases} 1, & 0 \leq t < 1, \\ -1, & \frac{1}{2} \leq t < 1, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Its scaling function $\lambda(t)$ can be written as

$$\lambda(t) = \begin{cases} 1, & 0 \leq t < 1, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

The steps of the denoising scheme are as follows:

Step 1: decompose input time-series signal using DWT (Discrete Wavelet Transform) selecting wavelet. In the

TABLE 1: Details of the data set

S. no.	Cities	Longitude	Latitude	Time period	Parameters	No. of days
1.	Delhi	N 28° 38' 37.266"	E 77° 11' 12.9084"	01.01.2017–30.05.2018	Maximum temperature, minimum temperature, evaporation, and wind speed	515
2.	Mumbai	N 19° 13' 43.7700"	E 72° 51' 14.8248"	01.01.2017–30.05.2018	Maximum temperature, minimum temperature, and wind speed	515
3.	Chennai	N 13° 4' 2.7804"	E 80° 14' 15.4212"	01.01.2020–28.02.2021	Maximum temperature, minimum temperature, and wind speed	425

study, the Haar wavelet is used, whose function is given in equations (1) and (2).

Step 2: by acting on the detail wavelet coefficient, the threshold function is used to reduce noise in the signal processed in Step 1. The coefficients are scaled or shrunk depending on the chosen threshold function. In the study, a soft thresholding technique has been used for which the function is as follows:

$$R = \begin{cases} \text{sign}(s) \cdot (|s| - |\text{Th}|), & |s| \geq \text{Th}, \\ 0, & |s| < \text{Th}, \end{cases} \quad (3)$$

where s and Th represent the wavelet coefficient and threshold value.

Step 3: extract the denoised time series by the inverse wavelet transformation of the output signal in Step 2.

3.2. NARX Method. NARX is a recurrent neural network with feedback connections spread over several layers of the network. It is used mostly in modeling of time series and is based on the linear ARX (Autoregressive Exogenous) model. It is defined by the following equation:

$$Y(t) = f(y(t-1), y(t-2), \dots, y(t-n_y), u(t-1), u(t-2), \dots, u(t-n_u)), \quad (4)$$

where on the past values of the output signal $Y(t)$ and the exogenous independent input signal, the next values of the output signal are regressed. By using a feed-forward neural network to approximate f , the NARX model is implemented. Besides being used as a predictor, NARX is used for non-linear filtering, where the target output is the noise-free version of the input signal.

The steps of the NARX scheme are as follows:

Step 1: load the decomposed output time-series signal denoised using DWT (Discrete Wavelet Transform).

Step 2: choose the time delay and the number of hidden neurons. Then, use Levenberg–Marquardt method to train the time series.

Step 3: train the time series and obtain the output.

In NARX, using Levenberg–Marquardt backpropagation method, the training of data is done. It reduces the error to the minimum. The algorithm is used for training the time series for the purpose of error reduction after being denoised using a wavelet.

3.3. Soft Computing Techniques. In the study, the kernel-based soft computing methods which have been used for forecasting and compared are described as follows.

3.3.1. RBF (Radial Basis Function) Method. The RBF kernels are the most widely used kernels because of having similarity with the Gaussian distribution. The RBF kernel computes the similarity or closeness between two data points and can be mathematically be presented as

$$K(y_1, y_2) = \exp\left(-\frac{\|y_1 - y_2\|^2}{2\sigma^2}\right), \quad (5)$$

where σ is the variance and the hyperparameter and $\|y_1 - y_2\|$ is the Euclidean (L_2 norm) distance between the two data points y_1 and y_2 .

Now linear regression is applied on RBF transformed data mentioned as above, and then we get

$$Y_n = [\text{RBF}(y_n, y_1), \text{RBF}(y_n, y_2), \text{RBF}(y_n, y_3), \dots, \text{RBF}(y_n, y_N)], \quad (6)$$

where N is the total number of data points. On the wavelet denoised and NARX trained data, the regression-based RBF conjunction model is applied to obtain the forecasted RBF data Y_n .

The steps of the RBF scheme are as follows:

Step 1: load the decomposed output time-series signal denoised using DWT (Discrete Wavelet Transform) and then train time series using NARX.

Step 2: choose the machine learning method-RBF for training the time series using the algorithm.

Step 3: obtain the desired output.

3.3.2. LibSVM Method. As an optimization algorithm, SVM (Support Vector Machine) was introduced by Shirmohammadi [16] to find the hyperplane with the maximum discriminating margin between two classes of data. To evaluate an input data sample y_t , the decision function used has the general form as follows:

$$Y_t = f(y_t) = \text{sign} \left(\sum_{i=1}^N z_i \alpha_i K(y_t, y_i) - b \right), \quad (7)$$

where $\{y_i, z_i\}_{i=1, \dots, N}$ is the training sample, $y_i \in \mathcal{R}^d$ is the input sample data from sample i of size d , and z_i takes two possible values 1 or -1. Kernel function $K(y_t, y_i)$ estimates the similarity between the i and t samples in the feature space with b as the threshold constant of the function. The coefficient values α_i correspond to the Lagrange multiplier of quadratic programming (QP) problem, which are obtained by minimizing the objective function as follows:

$$\underset{\alpha \in \mathcal{R}^N}{\text{minimize}} \psi(\alpha) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N z_i z_j K(y_i, y_j) \alpha_i \alpha_j - \sum_{i=1}^N \alpha_i, \quad 0 \leq \alpha_i \leq C, \quad \forall i, \quad \sum_{i=1}^N z_i \alpha_i = 0, \quad (8)$$

where the compromise between trading error minimization and the margin maximization is given by constant C and a large C involves high error penalization. The QP problem is divided into QP subproblems under the technique of chunking, where QP subproblems use a subset of non-zero α_i 's and the training samples (M) that violate the Karush-Kuhn-Tucker (KKT) condition. There are several SVM techniques like one-class SVM, nu-SVM, and R-SVM. LibSVM is an integrated software for SV classification, regression, and distribution estimation, including different SVM formulations.

The steps of the LibSVM scheme are as follows:

Step 1: load the decomposed output time-series signal denoised using DWT (Discrete Wavelet Transform) and then train time series using NARX.

Step 2: choose the machine learning method-LibSVM for training the time series using the algorithm.

Step 3: obtain the desired output.

LibSVM supports multiclass classification providing a simple interface where users can easily link it with their own programs. Weka software has been used for the study. Weka library for SVM has been used.

3.3.3. SMO (Sequential Minimal Optimization) Method. In each iteration, by making use of only two Lagrange multipliers, the SMO algorithm pushes the chunking method to the smallest possible expression. The optimal values of the multiplier are determined, and the SVM framework is updated until the QP problem is solved. Analytically, the optimization subproblem can be solved for two Lagrange multipliers which make SMO advantageous. In the methodology, the algorithm chooses two training samples (y_1, z_1) and (y_2, z_2) , whose associated Lagrange

multipliers are α_1 and α_2 . The optimization of the objective function in equation (8) for variables α_1 and α_2 now becomes

$$\begin{aligned} \underset{\alpha_1, \alpha_2}{\text{minimize}} \psi(\alpha) &= \frac{1}{2} (\alpha_1^2 K_{11} + 2s\alpha_1\alpha_2 K_{12} + \alpha_2^2 K_{22}) - \alpha_1 - \alpha_2, \\ 0 &\leq \alpha_1 \leq C, \\ 0 &\leq \alpha_2 \leq C, \\ s\alpha_2 + \alpha_1 &= \gamma, \\ s &= z_1 \cdot z_2, \\ \gamma &\in \mathcal{R}. \end{aligned} \quad (9)$$

The main routine initializes the SVM algorithm for two classification problems and evaluates all the samples and the associated Lagrange multipliers α_i . In the whole iteration, the value of the Lagrange multipliers does not change anymore and then the routine finishes. The output of the procedure is bias b and a list of multipliers α_i and any input data time series can be evaluated and forecasted using equation (7) of SVM.

The steps of the SMO scheme are as follows:

Step 1: load the decomposed output time-series signal denoised using DWT (Discrete Wavelet Transform) and then train time series using NARX.

Step 2: choose the machine learning method-SMO for training the time series using the algorithm.

Step 3: obtain the desired output.

Following the methodology, results generated are denoised by wavelet, trained by neural networks, and forecasted using kernel-based techniques of soft computing RBF, LibSVM, and SMO along with their conjunction

model. The error between forecasted time series (FTS) and original time series (OTS) is calculated by the following:

Performance Errors

- (a) Mean Absolute Error (MAE): MAE is used for measuring the average magnitude of errors in the set of predicted values. It measures efficiency for continuous variables. MAE is the average of the differences between observed values and actual values. MAE calculates the difference between actual values and forecasted values:

$$\text{MAE} = \sum_{i=1}^k \frac{\text{abs}(X_i - \alpha(X))}{n}, \quad (10)$$

where X_i is the predicted value; X is the actual value; N is the number of terms.

- (b) Relative Absolute Error (RAE): RAE is the same as RMSE but slightly smaller, and also it is calculated to compare the models to see which has better performance.

$$\text{RAE} = \frac{\sum |X_i - X|}{\sum |Y_i - X|}, \quad (11)$$

where X_i is the predicted values; Y_i is the previous target value; X is the actual value.

- (c) Root Mean Squared Error (RMSE): it measures the average magnitude of errors. The errors are squared before they are averaged, and the RMSE is relatively high weight to large errors.

$$\text{RMSE} = \sqrt{\frac{\sum (X_i - X)^2}{N}}, \quad (12)$$

where X is the actual value; X_i is the predicted values; N is the number of observations.

- (d) Root Relative Squared Error (RRSE): it is a tool to calculate the error, and the error is normalized by taking the square root.

$$\text{RRSE} = \frac{\sqrt{\sum \text{abs}(\beta - \alpha)^2 / N}}{\sqrt{\sum \text{abs}(\gamma - \alpha)^2 / N}}, \quad (13)$$

where β is the predicted values; γ is the previous target value; α is the actual value; N is the number of observations.

4. Results and Discussion

The data has been trained, tested, and validated as 70%, 15%, and 15% of all the data while using the NARX method to study the pattern of the time series. The parameters studied are maximum temperature, minimum temperature, and wind speed. It is observed that the denoised plus trained signal deviates least from the original data trajectory. For the

forecasting period June 1, 2018, to June 7, 2018, for Delhi and Mumbai and March 1, 2021, to March 7, 2021, for Chennai, the original data signaled-noised data signal, trained data signal, and the denoised plus trained data signal are fed to RBF, SMO, and LibSVM models to get the simple model forecasts under Scheme 1 and hybrid conjunction model forecasts under Schemes 2, 3, and 4, respectively. The comparison plot of simple and hybrid forecasts with the original data for the forecasting periods is shown in Figures 2(a)–2(l). For different simple and wavelet conjunction hybrid models, the error and time taken are compared in Tables 1–7 for the metropolitan cities and Figures to select the most efficient model.

The above error estimations and time taken by simple and conjunction models show the efficiency of the hybrid models, and hence, it can be concluded from Figures 6–8 and Tables 1–8 that the Wavelet + Neuro + RBF model shows the best results for the forecasting of weather parameters. RBF method has advantages of easy design, better generalization, and strong tolerance to input noise, and its learning ability makes it very suitable to design flexible control systems, whereas SMO is an iterative algorithm for solving problems. SMO breaks this problem into a series of the smallest possible subproblems, which are then solved analytically. Though LibSVM works stepwise, first by training the data to obtain model and then using the model to predict information of testing data set, RBF is a better method as compared to it as RBF works faster by calculating the distance between the data values and then studying the pattern of the data to make the forecasts accordingly. SMO and LibSVM are methods that take more time to study the pattern of time series. In the case of SMO, it takes more time to break the data set into smaller and smaller sets, due to which some points are left out, and hence, the error chances increase. Similarly, for LibSVM, as it includes multiple algorithms in it, it takes time to understand the pattern and then choose the correct model. RBF predictions are thus observed to be more accurate than LibSVM and SMO methods. In the case of LibSVM and SMO, the forecasts and errors from the hybrid model are not much different from the conventional model forecasts and errors, even though the time taken (TT) is significantly reduced using the hybrid scheme. But applying wavelet and NARX and then feeding the denoised plus trained data to the RBF yield better predictions for the forecasting period compared to the predictions obtained by directly feeding the data to RBF. It is observed that when the data input signal is first denoised using the Haar wavelet and then trained using the neural networks (NARX), the double processing of data reduces the error in the forecasts obtained after feeding the denoised plus trained data signal to plain RBF model. The hybrid or conjunction model of Wavelet + Neuro + RBF provides better predictions with the least error and time taken to build the model [15, 24] for the considered weather parameter data time series.

Time taken is in seconds, RRSE and RAE are taken in percentage, and MAE and RMSE have been taken as units.

FIGURE 2: Continued.

FIGURE 2: Comparison between actual and forecasted time series for Delhi. (a) Comparison plot between forecasted and actual data for maximum temperature (RBF). (b) Comparison plot between forecasted and actual data for maximum temperature (SMO). (c) Comparison plot between forecasted and actual data for maximum temperature (LibSVM). (d) Comparison plot between forecasted and actual data for minimum temperature (RBF). (e) Comparison plot between forecasted and actual data for minimum temperature (SMO). (f) Comparison plot between forecasted and actual data for minimum temperature (LibSVM). (g) Comparison plot between forecasted and actual data for wind speed (RBF). (h) Comparison plot between forecasted and actual data for wind speed (SMO). (i) Comparison plot between forecasted and actual data for wind speed (LibSVM). (j) Comparison plot between forecasted and actual data for evaporation (RBF). (k) Comparison plot between forecasted and actual data for evaporation (SMO). (l) Comparison plot between forecasted and actual data for evaporation (LibSVM).

TABLE 2: Comparison of errors and time taken for maximum temperature, Delhi.

Methods	Time taken	MAE	RMSE	RAE (%)	RRSE (%)
RBF	0.26	1.97	2.18	42.51	46.88
W-RBF	0.22	1.17	1.08	42.40	46.80
N-RBF	0.18	1.09	1.07	42.24	46.88
W-N-RBF	0.14	1.07	1.06	42.14	46.72
SMO	0.32	3.08	3.86	45.54	50.56
W-SMO	0.3	2.78	3.47	45.06	51.23
N-SMO	0.28	2.75	3.32	43.10	48.12
W-N-SMO	0.28	2.71	3.45	41.97	46.90
LibSVM	0.48	2.74	3.50	102.89	141.89
W-LibSVM	1.75	2.75	3.45	101.45	141.85
N-LibSVM	1.72	2.74	3.45	100.55	141.55
W-N-LibSVM	1.97	2.72	3.44	100.11	141.43

TABLE 3: Comparison of errors and time taken for minimum temperature, Delhi.

Methods	Time taken	MAE	RMSE	RAE (%)	RRSE (%)
RBF	0.27	2.33	1.44	35.23	38.16
W-RBF	0.25	1.58	1.17	33.18	37.64
N-RBF	0.24	1.15	1.18	32.14	36.51
W-N-RBF	0.24	1.10	1.15	31.40	35.74
SMO	0.26	2.12	2.58	36.81	38.16
W-SMO	0.26	1.88	2.32	35.09	38.19
N-SMO	0.26	1.87	2.35	33.11	35.42
W-N-SMO	0.22	1.86	2.32	32.35	34.75
LibSVM	0.31	2.03	2.59	100.08	141.40
W-LibSVM	1.04	2.41	2.65	141.23	141.92
N-LibSVM	1.05	1.91	2.46	141.88	141.88
W-N-LibSVM	1.06	1.81	2.39	141.40	141.43

TABLE 4: Comparison of errors and time taken for wind speed, Delhi.

Methods	Time taken	MAE	RMSE	RAE (%)	RRSE (%)
RBF	0.96	1.08	3.13	83.03	89.43
W-RBF	0.85	1.02	2.06	81.47	90.14
N-RBF	0.26	1.02	2.17	81.11	88.05
W-N-RBF	0.22	1.00	2.16	80.07	86.11
SMO	0.18	1.56	2.08	89.12	94.35
W-SMO	0.15	1.45	1.98	89.10	90.94
N-SMO	0.15	0.99	1.88	85.14	90.88
W-N-SMO	0.14	0.89	1.39	83.88	90.79
LibSVM	1.22	1.59	2.06	99.23	140.81
W-LibSVM	1.02	1.45	1.98	99.01	141.73
N-LibSVM	1.02	0.98	1.77	100.65	141.55
W-N-LibSVM	1.06	0.85	1.32	100.11	141.43

TABLE 5: Comparison of errors and time taken for evaporation, Delhi.

Methods	Time taken	MAE	RMSE	RAE (%)	RRSE (%)
RBF	0.11	0.71	1.60	48.99	51.07
W-RBF	0.12	0.65	1.05	45.79	47.05
N-RBF	0.11	0.62	1.04	42.89	47.12
W-N-RBF	0.08	0.60	1.00	40.67	45.66
SMO	0.16	0.73	0.93	50.49	52.88
W-SMO	0.2	0.70	0.99	48.18	51.12
N-SMO	0.24	0.71	0.90	47.19	51.55
W-N-SMO	0.23	0.69	0.89	47.03	50.43
LibSVM	0.96	2.25	0.90	138.80	142.96
W-LibSVM	1.96	1.99	0.99	104.50	142.05
N-LibSVM	1.22	1.89	0.85	102.88	141.96
W-N-LibSVM	1.21	1.88	0.81	100.12	141.43

TABLE 6: Comparison of errors and time taken for maximum temperature, Mumbai and Chennai.

Methods	Mumbai					Chennai				
	TT	MAE	RMSE	RAE	RRSE	TT	MAE	RMSE	RAE	RRSE
RBF	0.36	0.48	0.61	70.62	71.12	0.66	1.99	2.19	42.51	46.88
W-RBF	0.22	0.89	0.12	89.22	136.74	0.87	1.97	1.08	44.40	46.80
N-RBF	0.28	0.78	1.03	68.39	67.26	0.28	1.41	1.07	45.24	46.88
W-N-RBF	0.14	0.77	0.11	58.39	58.87	0.14	1.02	1.06	38.14	45.72
SMO	0.32	0.04	0.07	65.87	96.46	0.37	2.01	3.86	55.54	59.56
W-SMO	0.23	0.45	0.61	89.87	70.68	0.43	2.74	3.47	59.07	61.33
N-SMO	0.28	0.73	1.02	63.94	66.82	0.88	2.78	3.32	63.10	49.12
W-N-SMO	0.26	0.75	1.20	62.85	68.64	0.96	2.79	3.45	61.97	56.91
LibSVM	0.38	0.99	0.99	99.70	102.56	0.58	2.99	3.50	92.89	121.84
W-LibSVM	1.75	0.89	1.99	98.45	100.32	1.75	2.78	3.46	102.45	101.85
N-LibSVM	1.82	0.87	1.97	99.54	100.99	1.92	2.89	3.45	100.99	101.55
W-N-LibSVM	1.07	0.91	1.09	101.23	99.63	1.07	2.01	3.44	100.19	99.43

Figure 3 shows a comparison of actual and forecasted values of time series of Delhi.

Figures 3–5 are the time-series plot Delhi, Chennai, and Mumbai of original and denoised trained time-series.

Figure 6 shows the error plots of the statistical calculations done in Tables 2–5 for comparison of the models. The time-series plot of maximum temperature, minimum

temperature, and wind speed of Delhi, Mumbai, and Chennai are as follows.

In order to validate the observation based on the initial study of weather parameters on the daily data set for Delhi region from January 1, 2017, up till May 30, 2018, as obtained from IARI (Indian Agricultural Research Institute, Delhi), the study is further applied to independent data sets of daily

TABLE 7: Comparison of errors and time taken for minimum temperature, Mumbai and Chennai.

Methods	Mumbai					Chennai				
	TT	MAE	RMSE	RAE	RRSE	TT	MAE	RMSE	RAE	RRSE
RBF	0.47	2.30	2.74	81.10	73.88	0.27	2.33	1.44	35.23	38.16
W-RBF	0.35	2.21	2.62	83.9	74.69	0.25	1.58	1.17	33.19	37.64
N-RBF	0.24	2.03	2.44	82.99	74.11	0.24	1.15	1.18	32.14	36.52
W-N-RBF	0.22	2.00	2.14	70.50	73.10	0.04	1.01	1.15	30.41	32.74
SMO	0.46	2.15	2.75	75.82	74.12	0.96	2.99	2.58	46.81	48.17
W-SMO	0.29	2.06	2.67	78.05	76.19	0.56	1.80	2.32	45.10	58.19
N-SMO	0.16	1.89	2.49	77.61	75.30	0.87	1.80	2.35	53.12	55.42
W-N-SMO	0.22	2.02	2.62	77.46	75.54	0.62	1.87	2.33	52.35	94.75
LibSVM	0.31	2.98	2.99	98.45	80.98	0.31	2.03	2.59	101.09	101.40
W-LibSVM	1.44	2.89	2.14	88.23	81.96	1.94	2.41	2.65	131.23	100.92
N-LibSVM	1.15	1.99	2.98	89.63	82.99	1.15	1.19	2.47	121.88	101.88
W-N-LibSVM	1.09	2.97	2.66	102.54	99.54	1.76	1.70	2.39	121.40	99.43

(a)

(b)

(c)

FIGURE 3: Continued.

FIGURE 3: Original and denoised trained data time series for Delhi. (a) Maximum temperature data. (b) Minimum temperature data. (c) Wind speed data. (e) Evaporation data.

FIGURE 4: Original and denoised trained time series for Chennai. (a) Maximum temperature data, Chennai. (b) Minimum temperature data, Chennai. (c) Wind speed data, Chennai.

FIGURE 5: Original and denoised trained time series for Mumbai. (a) Maximum temperature data, Mumbai. (b) Minimum temperature data, Mumbai. (c) Wind speed data, Mumbai.

FIGURE 6: Continued.

FIGURE 6: Comparison plot of forecasting errors for Delhi data. (a) TT comparison. (b) MAE comparison. (c) RMSE comparison. (d) RAE comparison. (e) RRSE comparison.

TABLE 8: Comparison of errors and time taken for wind speed, Mumbai and Chennai.

Methods	Mumbai					Chennai				
	TT	MAE	RMSE	RAE	RRSE	TT	MAE	RMSE	RAE	RRSE
RBF	0.96	3.28	4.28	73.40	72.07	0.96	1.08	3.13	83.03	89.43
W-RBF	0.75	2.48	3.27	81.14	74.41	0.85	1.02	2.06	81.47	90.14
N-RBF	0.16	2.15	1.80	73.31	67.32	0.26	1.02	2.17	81.11	88.05
W-N-RBF	0.12	1.48	1.65	65.39	63.19	0.12	1.00	2.16	80.07	88.11
SMO	0.62	3.01	4.31	67.32	67.32	0.18	1.86	2.08	99.12	94.35
W-SMO	0.88	2.36	3.40	77.25	77.25	0.15	1.45	1.98	99.11	98.94
N-SMO	0.78	1.41	1.89	69.97	69.97	0.15	1.99	1.88	95.14	97.88
W-N-SMO	0.85	2.02	2.64	76.06	76.06	0.14	1.89	1.39	83.88	97.80
LibSVM	0.98	3.01	1.99	88.74	89.41	1.72	1.99	2.06	98.23	145.81
W-LibSVM	0.92	2.01	1.87	80.21	90.37	1.42	1.45	1.98	89.01	101.74
N-LibSVM	0.42	2.90	1.97	85.21	98.41	1.92	1.98	1.99	101.65	101.55
W-N-LibSVM	0.94	1.98	2.01	99.12	99.60	1.46	1.95	1.38	102.11	100.43

FIGURE 7: Comparison plot of forecasting errors for Chennai data. (a) TT comparison. (b) MAE comparison. (c) RMSE comparison. (d) RAE comparison. (e) RRSE comparison.

maximum temperature, minimum temperature, and wind speed of Mumbai region from January 1, 2017, up till May 30, 2018, and Chennai region from January 1, 2020, up till

February 28, 2021, as obtained from available meteoblue weather records online. The results are tabulated in Tables 6–8, which reconfirm our earlier observation of RBF

FIGURE 8: Comparison plot of forecasting errors for Mumbai data. (a) TT comparison. (b) MAE comparison. (c) RMSE comparison. (d) RAE comparison. (e) RRSE comparison.

producing better prediction than LibSVM and SMO and Wavelet + Neuro + RBF to be the most efficient method in comparison to the rest of hybrid and conventional schemes.

Figures 7 and 8 show the plot of error comparison of the models. The bias of the original and forecasted value for maximum, minimum temperature, and wind speed is

FIGURE 9: Bias of Delhi, Mumbai, and Chennai. (a) Maximum temperature. (b) Minimum temperature. (c) Wind speed.

calculated for Delhi, Mumbai, and Chennai. The histogram of the bias is shown in Figure 9.

5. Conclusion

In this paper, for Delhi (maximum temperature, minimum temperature, evaporation, and wind speed) and Mumbai (maximum temperature, minimum temperature, and wind speed) from January 1, 2017, up till May 30, 2018, the daily data of weather parameters of have been considered, and for Chennai (maximum temperature, minimum temperature, and wind speed) from January 1, 2020, up till February 28, 2021, the daily data of weather parameters have been considered in the first phase of the study. In the second phase, the wavelet-neuroconjunction models are studied. Using wavelet transformation, the time series of weather parameters was denoised, after which the denoised data time-series signal was trained using NARX, and finally, the denoised plus trained time series was fed to Lib SVM, RBF, and SMO to obtain novel hybrid forecasts for the forecasting period. These forecast outputs have been compared for the time taken by the models, and errors, namely, MAE, RMSE, RAE, and RRSE, have been calculated. In the case of LibSVM and SMO, the forecasts and errors from the hybrid model are not

much different from the conventional model forecasts and errors even though the time taken (TT) is significantly reduced using the hybrid scheme. RBF predictions are observed to be more accurate than LibSVM and SMO methods. Based on the error calculation and the time taken, the hybrid scheme using the wavelet-neuroconjunction model gives better output as compared to the simple application of the conventional RBF method. Hybrid models, due to proper denoising and training of the time-series signal using wavelet and NARX, respectively, yield better results compared to the forecasts obtained by directly feeding data to the RBF model, thus verifying the study on independent data sets of weather parameters for Mumbai from January 1, 2017, up till May 30, 2018, and Chennai from January 1, 2020, up till February 28, 2021. It is concluded that Wavelet + Neuro + RBF model shows better results for the forecasting of weather parameters in comparison to all rest hybrid and conventional models. It can finally be concluded that Wavelet + Neuro + RBF model shows better results for the different data values and for the different time periods as well. The study will help the concerned authorities for future planning and take preventive steps for future coming calamities if any. It will also help the government to make effective policies.

Data Availability

The data used to support this study are from IARI Meteorological Database System Division of Agricultural Physics, IARI, New Delhi (https://www.iari.res.in/index.php?option=com_content&view=article&id=450&Itemid=224).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors are thankful to Guru Gobind Singh Indraprastha University for the research facility and financial support.

References

- [1] R. Bhardwaj and V. Duhoon, "Dispersion analysis of monthly rainfall and temperature time series, 1901-2015," *Indian Journal of Industrial and Applied Mathematics*, vol. 11, no. 1, pp. 91-100, 2020.
- [2] R. Bhardwaj and S. Das, "Fractal analysis of Indian rhinoceros poaching at Kaziranga," *Jñānābha*, vol. 48, pp. 54-60, 2018.
- [3] R. Bhardwaj and V. Duhoon, "Time series analysis of heat stroke," *Jñānābha*, vol. 49, no. 1, pp. 01-10, 2019.
- [4] R. Bhardwaj and V. Duhoon, "Weather forecasting using soft computing technique," in *Proceedings of the IEEE International Conference on Computing, Power and Communication Technologies 2018 (GUCON)*, pp. 1111-1115, September 2018, Greater Noida, India.
- [5] R. Bhardwaj and V. Duhoon, "Real time weather parameter forecasting using anfis-sugeno," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 1, pp. 461-469, 2019.
- [6] R. Bhardwaj and V. Duhoon, "Weather forecasting based on hybrid wavelet-neural model," *Italian Journal of Pure and Applied Maths*, 2017.
- [7] J. Cifuentes, G. Marulanda, A. Bello, and J. Reneses, "Air temperature forecasting using machine learning techniques: a review," *Energies*, vol. 13, no. 16, pp. 1-28, 2020.
- [8] R. Bhardwaj and V. Duhoon, "Classification & clustering of time series of weather data," in *Proceedings of the ICICC-2020-International Conference on Innovative Computing and Communication*, pp. 257-270, February 2020, New Delhi, India.
- [9] A. M. Kalteh, "Enhanced monthly precipitation forecasting using artificial neural network and singular spectrum analysis conjunction models," *INAE Letters*, vol. 2, no. 3, pp. 73-81, 2017.
- [10] O. Kisi and M. Cimen, "Precipitation forecasting by using wavelet-support vector machine conjunction model," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 4, pp. 783-792, 2012.
- [11] L. Oana and A. Spataru, "Use of genetic algorithms in numerical weather prediction," in *Proceedings of the 18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pp. 456-461, Timisoara, Romania, September 2016.
- [12] G. Guizzi, C. Silvestri, E. Romano, and R. Revetria, "A comparison of forecast models to predict weather parameters," *Advances in Energy and Environmental Science and Engineering*, pp. 88-96, 2015.
- [13] T. Saba, A. Rehman, and J. S. AlGhamdi, "Weather forecasting based on hybrid neural model," *Applied Water Science*, vol. 7, no. 7, pp. 3869-3874, 2017.
- [14] K. Sepideh, K. Ozgur, S. Jalal, and M. Oleg, "A wavelet and neuro-fuzzy conjunction model to forecast air temperature variations at coastal sites," *The International Journal of Ocean and Climate Systems*, vol. 6, no. 4, pp. 159-172.
- [15] S. S. Reddy and C. M. Jung, "Short-term load forecasting using artificial neural networks and wavelet transform," *International Journal of Applied Engineering Research*, vol. 11, no. 19, pp. 9831-9836, 2016.
- [16] B. Shirmohammadi, H. Moradi, and V. Moosavi, "Forecasting of meteorological drought using Wavelet-ANFIS hybrid model for different time steps (case study: southeastern part of east Azerbaijan province, Iran)," *Natural Hazards*, vol. 69, no. 389, 2013.
- [17] T. Partal and H. K. Cigizoglu, "Prediction of daily precipitation using wavelet-neural networks," *Hydrological Sciences Journal*, vol. 54, no. 2, pp. 234-246, 2009.
- [18] A. Piasecki, J. Jurasz, and J. F. Adamowski, "Forecasting surface water-level fluctuations of a small glacial lake in Poland using a wavelet-based artificial intelligence method," *Acta Geophysica*, vol. 66, no. 5, pp. 1093-1107, 2018.
- [19] M. M. H. Khan, N. S. Muhammad, and A. El-Shafie, "Wavelet-ANN versus ANN-based model for hydrometeorological drought forecasting," *Water (Switzerland)*, vol. 10, no. 8, 2018.
- [20] F. Karami and A. B. Dariane, "Optimizing signal decomposition techniques in artificial neural network-based rainfall-runoff model," *International Journal of River Basin Management*, vol. 15, no. 1, pp. 1-8, 2017.
- [21] B. Kanna and S. N. Singh, "Long term wind power forecast using adaptive wavelet neural network," in *Proceedings of the 2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON)*, pp. 671-676, Varanasi, India, December 2016.
- [22] K. Adamowski J& Sun, "Development of a coupled wavelet transform and neural network method for flow forecasting of non-perennial rivers in semi-arid watersheds," *Journal of Hydrology*, vol. 390, no. 1-2, pp. 85-91, 2010.
- [23] A. Araghi, J. Adamowski, and C. J. Martinez, "Comparison of wavelet-based hybrid models for the estimation of daily reference evapotranspiration in different climates," *Journal of Water and Climate Change*, vol. 11, no. 1, pp. 39-53, 2018.
- [24] H. Liu, H.-q. Tian, D.-f. Pan, and Y.-f. Li, "Forecasting models for wind speed using wavelet, wavelet packet, time series and Artificial Neural Networks," *Applied Energy*, vol. 107, no. C, pp. 191-208, 2013.
- [25] V. Vapnik, *Estimation of Dependences Based on Empirical Data*, Springer-Verlag New York, Inc., New York, NY, USA, 1982.