Article

Balanced Spider Monkey Optimization with Bi-LSTM for Sustainable Air Quality Prediction

Chelladurai Aarthi [1], Varatharaj Jeya Ramya [2], Przemysław Falkowski-Gilski [3,*] and Parameshachari Bidare Divakarachari [4,*]

[1] Department of Electronics and Communication Engineering, Sengunthar Engineering College, Tiruchengode 637205, Tamil Nadu, India

[2] Department of Electronics and Communication Engineering, Panimalar Engineering College, Chennai 600123, Tamil Nadu, India

[3] Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology, Narutowicza 11/12, 80-233 Gdansk, Poland

[4] Department of Electronics and Communication Engineering, Nitte Meenakshi Institute of Technology, Bengaluru 560064, Karnataka, India

[*] Correspondence: przemyslaw.falkowski@eti.pg.edu.pl (P.F.-G.); paramesh@nmit.ac.in (P.B.D.)

Abstract: A reliable air quality prediction model is required for pollution control, human health monitoring, and sustainability. The existing air quality prediction models lack efficiency due to overfitting in prediction model and local optima trap in feature selection. This study proposes the Balanced Spider Monkey Optimization (BSMO) technique for effective feature selection to overcome the local optima trap and overfitting problems. The air quality prediction data were collected from the Central Pollution Control Board (CPCB) from four cities in India: Bangalore, Chennai, Hyderabad, and Cochin. Normalization is performed using Min-Max Normalization and fills the missing values in the dataset. A Convolutional Neural Network (CNN) is applied to provide deep representation of the input dataset. The BSMO technique selects the relevant features based on the balancing factor and provides the relevant features for the Bi-directional Long Short-Term Memory (Bi-LSTM) model. The Bi-LSTM model provides the time series prediction of air quality for four cities. The BSMO model obtained higher feature selection performance compared to existing techniques in air quality prediction. The BSMO-BILSTM model obtained 0.318 MSE, 0.564 RMSE, and 0.224 MAE, whereas Attention LSTM reached 0.699 MSE, 0.836 RMSE, and 0.892 MAE. Our solution may be of particular interest to various governmental and non-governmental institutions focused on maintaining high Quality of Life (QoL) on the local or state level.

Keywords: air quality prediction; balanced spider monkey optimization; bi-directional long short-term memory; convolutional neural network; sustainability

1. Introduction

Air contamination affects the human health, it is considered as one of the major problems in recent days. The ever-growing population and motorization in cities tend to increase the traffic volume which leads to higher gas emissions, etc. [1]. Air pollution has been a major problem over the past few decades in most developing countries. High levels of air pollutants lead to chronic diseases, such as chronic respiratory diseases, heart failure, bronchitis, etc. People with diabetics, heart and lung disease, and children and elderly people are vulnerable to health effects related to air pollution. Additionally, air pollutants and derivatives can cause many adverse effects related with, e.g., water quality and environment degradation, global climate change, acid deposition, visibility impairment, and plants [2].

Multivariate Time Series (MTS) is gaining more attention and importance as time series data generation in the Internet of Things (IoT) era advances. A deep learning (DL)

architecture is applied for time series forecasting and is still an active research area for different studies. Most researchers focus on applying the single architecture to solve problems in time series forecasting [3]. Accurate air quality prediction is key to improve the local government's rapid response [4]. Particulate Matter (PM) is divided into three major groups, namely: coarse particles (PM_{10}), fine particles ($PM_{2.5}$), and ultrafine particles ($PM_{0.1}$). Overall, their sizes vary in source and health effects. In particular, $PM_{2.5}$ particles are more active compared to larger pollutants that can spread quickly and remain in the air for a longer time. It also carries substances that affects human health and the surrounding biological environment [5].

In feature selection, optimization-based methods are used for the intelligent behavior of the spider monkey, which inspired us to develop mathematical models that follow the Fission Fusion Social Structure (FFSS). Effective feature selection techniques help to solve overfitting in classifiers by removing irrelevant features from the extracted ones. Convolutional Neural Network (CNN) based models are applied for feature extraction and classification in various fields of prediction and image processing due to its efficiency [6–13]. Accurate air quality prediction models help to control and prevent air pollution and protect residents' health. Time series based classical statistical models were applied for air quality prediction, they include: Multiple Linear Regression (MLR), and Autoregressive Integrated Moving Average (ARIMA) [14]. Many researchers focus on time series data for the air quality prediction. Several scientists successfully applied machine learning (ML) models for this purpose. The Support Vector Regression (SVR) is a machine learning technique used to minimize the structural risk based on statistical learning [15,16]. Recurrent Neural Network (RNN) models were extensively applied for learning of time series data and Long Short-Term Memory (LSTM) is RNN model to learn long-term temporal dependencies [17,18]. Efficient air quality prediction model helps in various fields, including pollution control, sustainability, human health, and government policies. Various models were applied for the air quality prediction process, yet most models face limitations related with the overfitting problem.

The main contributions of this paper are as follows:

1. The Balanced Spider Monkey Optimization (BSMO) model is proposed in this research for feature selection in air quality prediction. The balancing factor in the BSMO model maintains a tradeoff between the exploration and exploitation that helps to overcome the overfitting and local optima trap.
2. The Convolutional Neural Network (CNN) model is applied for feature extraction purposes to provide hidden representation of the input dataset. The Bi-directional Long Short-Term Memory (Bi-LSTM) model is used for the prediction process due to its efficiency in handling time series data.
3. The Balanced Spider Monkey Optimization Bi-directional Long Short-Term Memory (BSMO-BILSTM) model has higher performance than existing methods in air quality prediction. It effectively handles the time series data and delivers less erroneous values.

This paper is organized as follows: an introduction about the air quality prediction is described in Section 1. The review of recent models of air quality prediction is given in Section 2, whereas the proposed BSMO-BILSTM model is explained in Section 3. The simulation setup is given in Section 4, while obtained results are demonstrated in Section 5. Finally, the conclusions and future scope of this study are stated in Section 6.

2. Literature Review

Air quality prediction is important for interdisciplinary air quality research, human health, and sustainable growth, not to mention Quality of Life (QoL), etc. Some of the recent air quality prediction models have been reviewed in this section.

Ragab et al. [19] applied Exponential Adaptive Gradients (EAG) optimization and 1D-CNN model for prediction process. The 1D-CNN model handled gradients of past model to improve learning rate and convergence. Three years of hourly air pollution were used for training the model. Parameter optimization and model evaluation processes

were conducted with a grid search technique. The irrelevant features were selected in the network, and this affected the model performance.

Hashim et al. [20] performed air quality prediction using weather parameters and hourly air pollution data. Six prediction models, such as Principal Component Regression (PCR), Radial Basis Function (RBFANN), Feed-Forward Neural Network (FFANN), Multiple Linear Regression (MLR), PCA-RBFANN, and PCA-FFANN, were investigated. The developed models had limitations of the overfitting problem due to irrelevant feature selection.

Sun and Liu [21] applied an ARMA-LSTM model for air quality prediction and a decomposition technique was applied for the data utilization. The vanishing gradient problem and overfitting problem affected the performance of the model once again.

Saufie et al. [22] applied six wrapper methods, such as genetic algorithm, weight-guided, brute force, stepwise, backward elimination, and forward selection for air quality prediction. Here, Wrapper method has been used for the feature selection. The Multiple Linear Regression (MLR) and Artificial Neural Network (ANN) predictive models were used for the classification process. The model showed that brute-force was the dominant wrapper method and MLR technique provided higher efficiency in the classification. The feature selection technique had local optima and overfitting problems in the classification.

Mao et al. [23] applied deep learning LSTM-based model for air quality prediction. The multi-layer Bi-LSTM model was applied with optimal time lag to realize sliding prediction based on temporal, meteorological, and $PM_{2.5}$ concentrations. The overfitting problem in the classifier affected the model performance.

Zou et al. [24] applied LSTM based model with spatio-temporal attention mechanism for air quality prediction. A temporal attention technique was used in a decoder to capture air quality dependence. The overfitting problem affected the developed model efficiency in the classification.

Seng et al. [25] developed a comprehensive prediction technique with multi-index and multi-output based on LSTM. The gaseous pollutant, meteorological, and nearest neighbor stations data were used for prediction of particle concentration. The LSTM based model had considerable performance in air quality prediction on a given dataset. The vanishing gradient and overfitting problem affected the model performance.

Ge et al. [26] applied a Multi-scale Spatio-Temporal Graph Convolution Network (MST-GCN) for air quality prediction. The MST-GCN model consisted of several spatial-temporal blocks, a fusion block, and a multi-scale block. The extracted features were separated into several groups based on two graphs of spatial correlations and domain categories. The irrelevant features from the selected features caused an overfitting problem in the prediction as well.

Janarthanan et al. [27] applied LSTM and SVR based model for prediction of air quality in Chennai city. The Grey Level Co-occurrence Matrix (GLCM) was used for extraction of the mean and mean square error, and standard deviation for feature extraction. The DL-based model provided efficient performance in the air quality prediction.

Asgari et al. [28] developed a parallel air quality prediction system with spatio-temporal data partition techniques, spark efficient environment, ML, resource manager, and Hadoop distributed platform. A distributed random forest technique was used for the evaluation process.

Dun et al. [29] combined a spatio-temporal correlation and fully connected CNN model for prediction of air quality. Grey relation analysis, $PM_{2.5}$ concentrations, and a new calculation method for measuring distance were used during the process. The CNN-based model had an overfitting problem that affected the model performance.

Ma et al. [30] applied a DL-based model of Graph CNN (GCN) for prediction of air quality. The coordinates of the monitoring points of spatial correlations were considered and Radial Basics Function (RBF) based fusion was conducted. The CNN model had an overfitting problem as well.

According to discussed models, air quality is a major issue that must be resolved in order to prevent or lessen the effects of pollution. For feature selection, Modified Grey

Wolf Optimization (MGWO) and Particle Swarm Optimization (PSO) have been used. The state of air will prompt us to take precautions, it may encourage people to carry out their regular activities in less polluted regions. However, it is still difficult to analyze the data and offer better results. Prediction of air pollution is not exempt from the sectors where deep learning technologies bring a significant increase in influence and penetration. In order to effectively predict the air quality, the authors employ sophisticated and advanced approaches. It is crucial to consider external aspects, such as weather conditions, spatial characteristics, and temporal features.

3. Proposed Method

The data were collected from the Central Pollution Control Board (CPCB) for four cities in India. The data were gathered in Bangalore, Chennai, Hyderabad, and Cochin, during a 5 year period between 2016–2022 [31]. Overall, the process of monitoring pollutants was performed two times a week, during a period of 24 h, which yields 104 observations per year. The normalization was performed using the Min-Max Normalization method that filled the missing values in the dataset. The CNN model was applied to extract the features from the input dataset and provide the hidden representation of features.

The BSMO technique was applied to select the factors from the extracted features for prediction purposes. The BSMO technique applied balancing factors to stabilize the exploration and exploitation in the feature selection which helps to escape from the local optima trap. The overview of the BSMO technique in air quality prediction is shown in Figure 1.
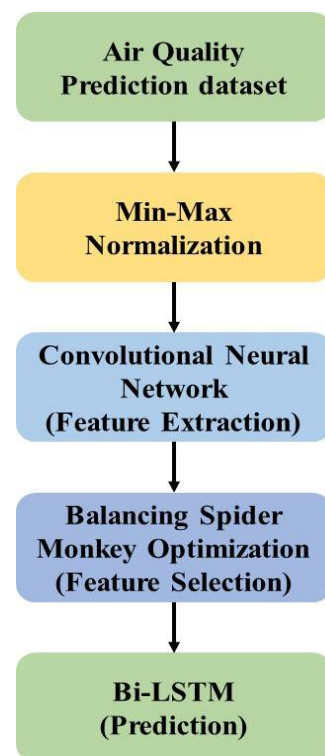

Figure 1. Overview of the proposed BSMO in air quality prediction technique.

3.1. Normalization

To maintain the original shape of the image, min-max normalization provided relationships among the original data values. This constrained range came at the expense of smaller standard deviations that supported to reduce the impact of outliers. The Min-Max

Normalization technique was applied to reduce the difference between respective values in the dataset. The formula for Min−Max Normalization is given in Equations (1) and (2):

$$X_{std} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{1}$$

$$X_{scaled} = X_{std} \times (max - min) + min \tag{2}$$

where the minimum feature range of input data X is min, and the maximum feature range is max.

3.2. CNN Based Feature Extraction

The CNN models consist of neurons organized in a layer to learn hierarchical representations similar to the typical neural network type [32–34]. Weights and biases are used to connect neurons. The input layer takes the initial data and processes them into a final layer that provides the prediction of the model. Input feature space transforms hidden layers to match its output. Generally, CNNs are applied with a minimum of one convolutional layer to exploit patterns. The architecture of the CNN is shown in Figure 2.
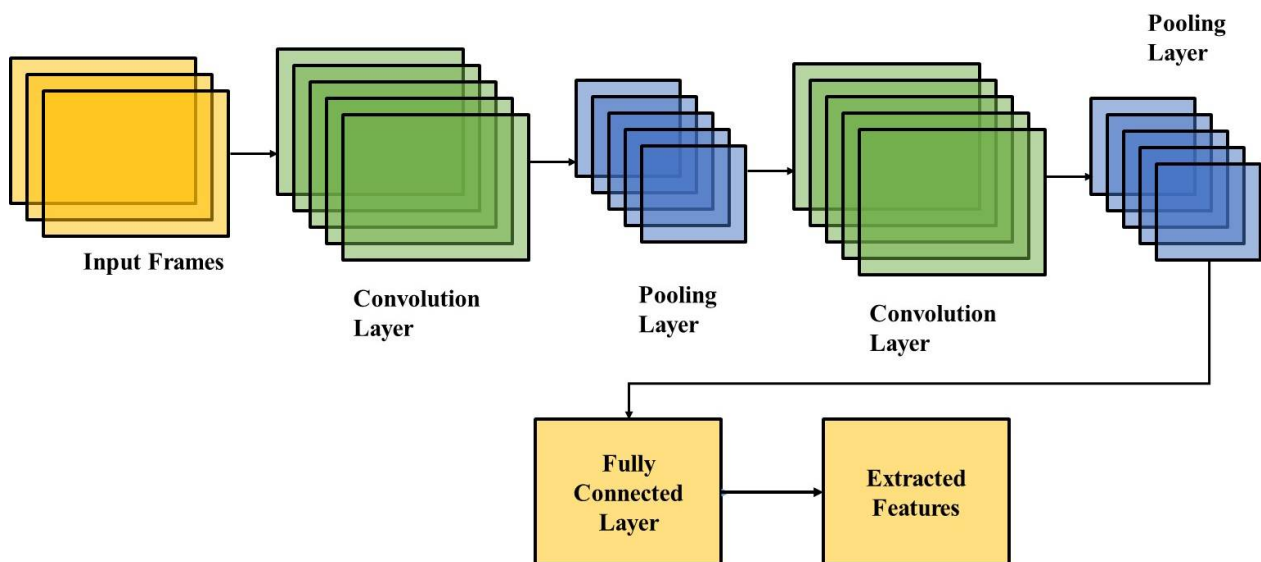

Figure 2. CNN for feature extraction process.

CNN are widely used in medical research, natural language processing, and other fields. Researchers found that CNN models have unique advantages for processing of input data and provide hidden representation. This study applies 1D time series data as input and air quality index of the next movement as an output to train a 1D-CNN model for prediction of air quality at the next step.

The input time series data is denoted as $x_{in} \in R_n$ with a length of n. The air quality index of the corresponding feature of output is denoted as $y \in R_{10}$ with a vector length of 10. The 1D-CNN learning goal is a non-linear mapping between the input factor and air quality index output for next time period, as described in Equation (3):

$$y_{out} = F(x_{in}) \tag{3}$$

where a complex non-linear mapping of input and output is denoted as F. For better representation of complex mapping relationships, the time series prediction problem

is converted to MSE. The MSE finds the minimum set to reduce differences between relationship fitted to mapping function F and real relationship, as provided in Equation (4):

$$L = \underset{F}{argmin}||y_{out}(x) - F(x)||_2^2 \tag{4}$$

Calibrate output and output differences are reduced using this process. Neural network processing is used to adjust the parameters.

The multi-layer CNN model is used in this study and consists of a total number of I layers. The input a_{i-1} is the upper layer output for ith convolutional layer and ith layer output is given in Equation (5):

$$a_i = f_i(a_{i-1}) = \sigma(W_i a_{i-1} + b_i) \tag{5}$$

where convolution kernel weight is denoted as W_i and non-linear activation function is denoted as σ. After the convolution process, offset b_i is added for a better non-linear fitting. The Rectified Linear Unit (ReLU) function is applied as an activation function for the ith layer of a convolutional layer. The activation function is used as a layer of sigmoid function for the fully connected layer of the ith layer. The non-linear relationship of input and output is denoted in Equation (6):

$$F(x_{in}) = f_l(f_{l-1}(\ldots f_1(x_{in}))) \tag{6}$$

The last layer provides the output of the CNN that consists of many features and back propagation is performed to reduce the error between calibration output and output. This helps to adjust the optimization parameters to reduce differences based on mean square error for minimization. The input data are convolved, pooled, fully connected, and multi-classified for a 1D-CNN, and, finally, features are extracted for the prediction process. The fully connected layer is shown in Figure 3.
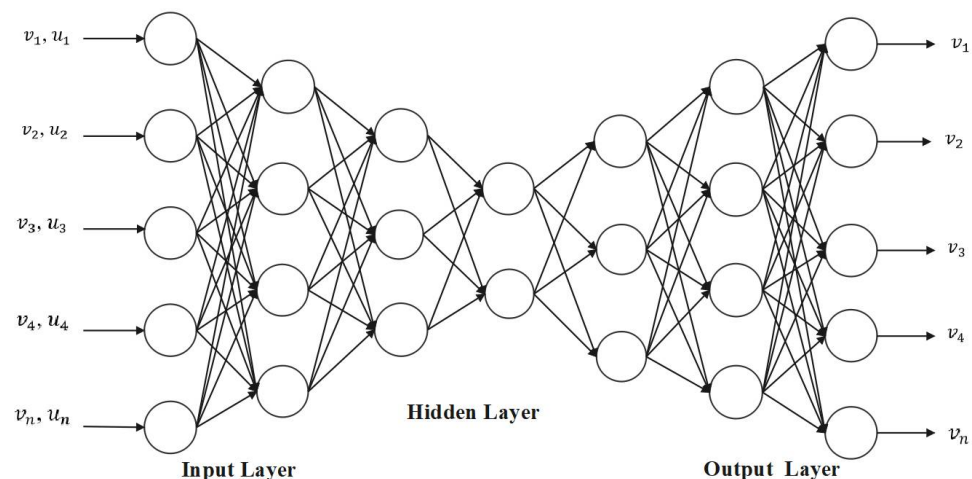

Figure 3. Fully connected layer in the CNN.

It consists of four parts: a fully connected layer, 1D pooling layer, 1D convolutional layer, and 1D-input layer. The pooled layer and convolution kernels are different from a two-dimensional CNN structure that also have a one-dimensional pooled layer structure.

1D-CNN convolution process: In a 1D CNN, the first layer of convolution is considered as an operational relationship between input data $x_{in} \in R_n$ and weight vector $W \in R_m$. The size of a weight vector is m and the convolution kernel size is also m. Specifically, each element of air quality concentration for that instance and air quality index period of output vector is denoted as x_{in}. Each m sequence of convolution kernel size m with length of input vector is used to provide first layer output. The air quality concentration value is denoted

as $m \ll n$ at input of each moment, which is included in the convolution operation. The step size is set as 1, and the convolution formula is given in Equation (7):

$$a_i = x_{i:i+m-1}W^T \tag{7}$$

1D-Convolution Layer: The one-dimensional vector is applied for 1D convolution layer; therefore, convolution kernel is set as one-dimensional. A 1D convolution process is used to provide the convolution process, more specifically, for an input length of 7, a kernel size of 5, and a convolution step of one.

1D-Convolution pooling layer: The 1D-CNN fully exerts the neural network due to existence of a pooling layer for feature extraction. The 1D-CNN training speed is inherently higher than other neural networks.

3.3. Balanced Spider Monkey Optimization

The spider monkey's intelligent behavior is inspired to develop SMO mathematical models that follow the FFSS [35–37]. Monkeys distribute from larger to smaller groups and smaller to larger for foraging based on FFSS. The BSMO steps are given as follows:

1. Initially, spider monkeys are present as 40–50 individuals in a group. Every group has a leader that makes decision for exploration of food, and this is a global leader of that group.
2. If food quality is insignificant, smaller sub-groups are created by the global leader. Each sub-group contains 3 to 8 members independently foraging, and each subgroup is led by a local leader.
3. For each sub-group, food search decision is selected by a local leader.
4. Communication of defensive boundaries and social bonds maintained by group members is conducted, and a unique sound is used by other members in that group.

The SMO foraging behavior of mathematical models is used for optimization problems and consist of six different phases. The SMO applies a 1D vector to denote a spider monkey and N spider monkey populations are randomly generated by SMO. Consider X_{ij}, which provides jth with the dimension of ith, the individual. Each X_{ij} of SMO is used to initialize as described in Equation (8):

$$X_{ij} = X_{minj} + U(0,1) \times (X_{maxj} - X_{minj}) \tag{8}$$

where lower and upper bounds of X_i in jth direction are X_{maxj} and X_{minj}, random number in the range of [0, 1] is denoted as $U(0, 1)$. The six phases of SMO are discussed as follows.

3.3.1. Local Leader Phase (LLP)

A new position individual is attained in this phase on basic knowledge of a local leader and group individuals, as described in Equation (9). The fitness value is used to decide a particular solution regarding quality. The highest fitness solution is selected for the next iteration:

$$Xnew_{ij} = X_{ij} + U(0,1) \times \left(LL_{kj} - X_{ij} \right) + U(-1,1) \times \left(X_{rj} - X_{ij} \right) \tag{9}$$

where jth is the direction of a local group leader position denoted as X_{kj} and X_{rj}, and rth spider monkey is selected from kth group, respectively. In order to maintain perturbation of present location, perturbation rate or probability p_r is used.

3.3.2. Global Leader Phase (GLP)

Global leader information is used to update individual spider monkey positions and group members, as denoted in Equation (10):

$$Xnew_{ij} = X_{ij} + U(0,1) \times (GL_j - X_{ij}) + U(-1,1) \times (X_{rj} - X_{ij}) \tag{10}$$

where global leader of *jth* direction is denoted as GL_j. The probability $prob_i$ is used to select particular dimension for X_i update and fitness value is calculated for each individual, according to Equation (11):

$$prob_i = \frac{fitness_i}{\sum_{i=1}^{N} fitness} \tag{11}$$

The better solution similar to LLP are newly generated position and spider monkey's old position is used for further processing.

3.3.3. Global Leader Learning (GLL) Phase

Global leader overall best fitness is provided in this phase and position change of the global leader is noted.

3.3.4. Local Leader Learning (LLL) Phase

Local leader is the best fitness position in the group. If the previous position of a local leader remains the same, then, similar to the GLL phase, local limit counter is updated with one.

3.3.5. Local Leader Decision (LLD) Phase

If a local limit counter of a local leader reaches to threshold count, then group members are reinitialized according to Equation (12):

$$Xnew_{ij} = X_{ij} + U(0,1) \times (GL_j - X_{ij}) + U(-1,1) \times (X_{rj} - X_{ij}) \tag{12}$$

3.3.6. Global Leader Decision (GLD) Phase

If a leader position was not updated for the applied iterations, then the global leader creates small sub-groups. The GLD of each group's local leaders are selected using the LLL phase. If the position is not updated for a time threshold value, then the global leader merges smaller groups to a single group. The SMO process of FFS structure is processed in this phase.

The balanced factor is proposed to provide a new global weighting coefficient α to control past classes importance using Equation (13):

$$q_k(x) = \frac{\lambda_k e^{z_k(x)}}{\sum_{j=1}^{N_t} \lambda_j e^{z_j(x)}} \ with \ \lambda_i = \begin{cases} \alpha n_i, & if \ i \in P \\ n_i, & otherwise \end{cases} \tag{13}$$

where weighting coefficient α is a real number, and the set of previous encountered classes denoted as P, usually between 0 and 1. The coefficient λ_i expression is not changed for new classes, and the training set n_i number of images for old classes is multiplied using weighting coefficient α. The flow chart of BSMO method is shown in Figure 4.

3.4. Bi-LSTM for Classification

The RNN based models are effective for time series applications, and this model has capacity to learn temporal dependencies. An RNN-based model has the possibility to learn current data based on previous data. RNN learn long term dependencies to solve one of RNN variants. The LSTM model, shown in Figure 5, is an improved version of the RNN and LSTM, and consists of memory cell or hidden layers to learn the long-term dependencies. Three gates, such as forget, input, and output, were used in memory cells to store temporal state of a network [38–40]. Network rest output and input control of a control memory cell is based on the output and input gate. Additionally, the forget gate in a network helps to pass high weight information to next neurons and discard the remaining ones. The activation function is used to determine the weight values for information and the high weight value of information is forwarded to the following neurons.
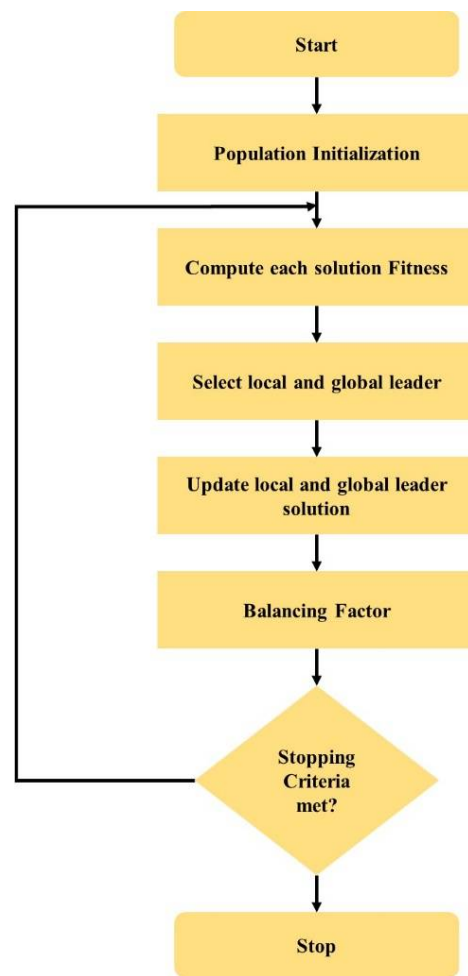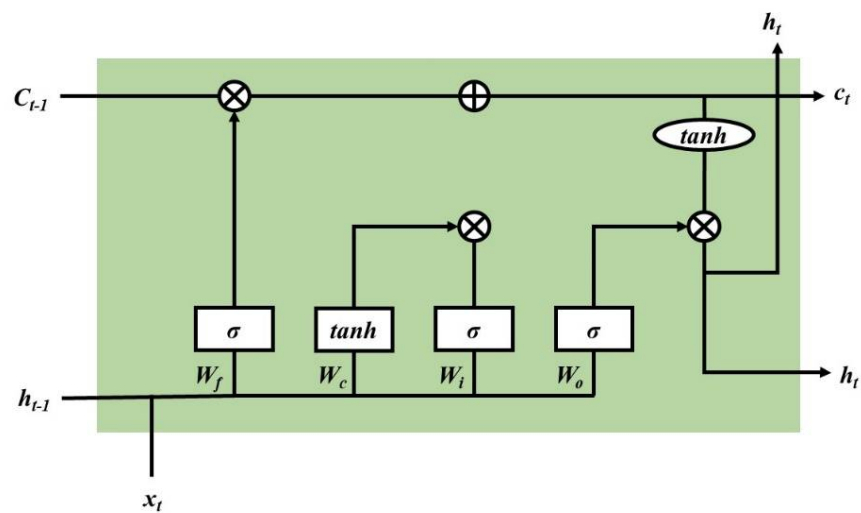
Figure 4. Flow chart of the BSMO method.

Figure 5. LSTM unit cell.

The input and output sequence are mapped using a LSTM network, i.e., $X = (X_1, X_2, \ldots, X_n)$ *and* $y = (y_1, y_2, \ldots, y_n)$, calculated according to Equations (14)–(17):

$$forgetgate = sigmoid\left(W_{fg}X_t + W_{hfg}h_{t-1} + b_{fg}\right) \tag{14}$$

$$inputgate = sigmoid\left(W_{ig}X_t + W_{hig}h_{t-1} + b_{ig}\right) \tag{15}$$

$$outputgate = sigmoid\left(W_{og}X_t + W_{hog}h_{t-1} + b_{og}\right) \tag{16}$$

$$(C)_t = (C)_{t-1} \otimes (forgetgate)_t + (inputgate)_t \\ \otimes(\tanh(W_C X_t + W_{hC}h_{t-1} + b_C)) \tag{17}$$

where weights and biases are represented using W_{ig}, W_{og}, W_{hC}, and W_{fg}, and b_{fg}, b_{ig}, b_{og}, and b_C for memory cells and three gates, respectively. The prior hidden layer unit is used to represent h_{t-1} and three gates' weights is added element wise. After processing described in Equation (13), a current memory cell unit is updated using $(C)_t$. The previous cell unit element wise is multiplied, and the output of the hidden unit is processed as described in Equation (14). Three gates are added with non-linearity in form of *tanh* and sigmoid activation function, as in Equations (14)–(17). The current time step is denoted as t and previous time step is denoted as $t-1$.

LSTM cell can work on previous content and not on future ones. The bi-directional recurrent neural network is shown in Figure 6. The input sequence is denoted as $X = (X_1, X_2, \ldots, X_n)$, Bi-LSTM forward direction is denoted as $\overrightarrow{h}_t = \left(\overrightarrow{h}_1, \overrightarrow{h}_2, \ldots, \overrightarrow{h}_n\right)$, and backward direction as $\overleftarrow{h}_t = \left(\overleftarrow{h}_1, \overleftarrow{h}_2, \ldots, \overleftarrow{h}_n\right)$. The final cell output y_t is formed using $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$, the output final sequence is denoted as $y = (y_1, y_2, \ldots y_t \ldots, y_n)$.
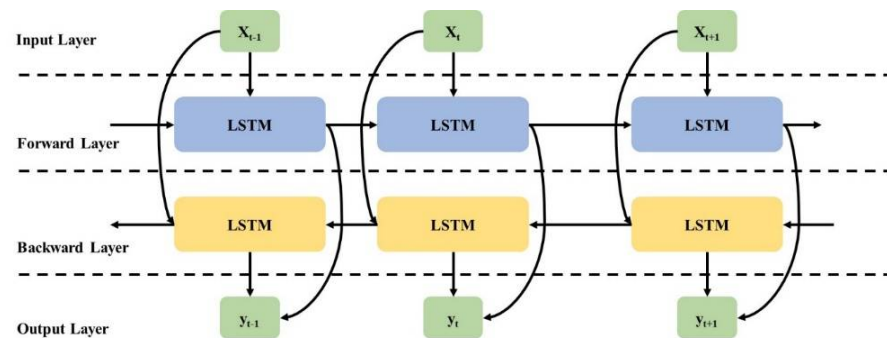

Figure 6. Bi-LSTM architecture for the air quality prediction.

4. Simulation Setup

Dataset: The majority of Indian cities continue to fall short of both national and global PM_{10} air quality targets. Pollution from respirable particulate matter continues to be a major problem for India. Some cities demonstrated far more improvement than others, despite the overall non-attainment, establishing the execution of a comprehensive plan for the prevention, control, and reduction of air pollution. The SO_2, NO_x, and PM_{10} are the major air pollutants that India's Central Pollution Control Board currently regularly monitors. Currently, these actions are realized regularly at 308 operating stations in 115 cities and towns throughout 25 states and 4 Indian union territories. Along with the monitoring of air quality, meteorological parameters, such as relative humidity, temperature, relative wind speed, location, and direction, are also included. The monitoring of these pollutants is done twice a week during a 24 h period, yielding 104 observations per year. The parameters considered for this research were: SO_2, NO_x, and $PM_{2.5}$. The data were collected in four cities in India, namely Bangalore, Chennai, Hyderabad, and Cochin, between 2016–2022 from the CPCB.

Metrics: As this model is a time series prediction, the error values such as MSE, RMSE, and MAE, have been calculated and compared with existing technique. The metrics equations are given as follows in Equations (18)–(20):

$$MSE = \frac{1}{n} \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i \right)^2 \tag{18}$$

$$RMSE = \sqrt{MSE} \tag{19}$$

where n denotes the number of data points, Y_i represents observed values, and $\hat{Y}_i$ relates to predicted values.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{20}$$

where y_i denotes the prediction value, x_i represents the true value, and n relates to the total number of data points.

System Requirement: The BSMO technique was tested on a desktop PC stand with an Intel i9 processor, 128 GB of RAM, a 22 GB graphics card, and Windows 10 as the serving OS. The Python 3.7 tool was used to implement the proposed model.

Parameter settings: For the CNN model, the learning rate was set to 0.001, and number of epochs was equal to 30. For the BSMO technique, the population was denoted as 50, whereas the number of iterations was fixed to 50, and the balancing factor of threshold value was set to 0.5. Bi-LSTM consists of 3 input/output layers and two hidden layers with similar output of opposite direction, which has 32 neurons. The output layer is exploited with previous and future information with this architecture. The Bi-LSTM model was applied with 30 epochs, with early stopping, 0.01 learning rate, and 0.1 dropout rate.

5. Results

In the result analysis, the proposed air quality prediction model helps to solve overfitting issues by removing irrelevant features in extracted features. Figure 7 displays the Mean Standard Error (MSE) value of the Bi-LSTM model as it predicted the air quality index [19,27] for several epochs.
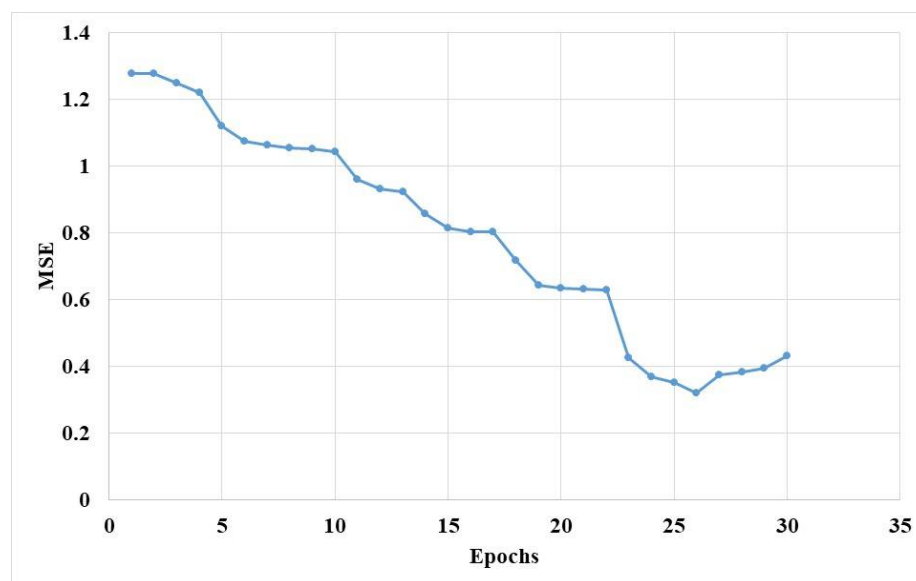

Figure 7. MSE values of various epochs of BILSTM.

The MSE value of the Bi-LSTM model was reduced up to 26 epochs and started to increase due to the overfitting problem. The early stopping technique stops the model at the 26th epoch and prevents overfitting in the model. Various DL techniques were applied for the feature extraction process and performance measurements. Their results are shown in Table 1.

Table 1. Feature extraction techniques in air quality prediction.

Methods	MSE	RMSE	MAE
Without Feature Extraction	2.344	1.531	1.236
AlexNet	1.24	1.114	0.73
VGG19	0.592	0.769	0.688
ResNet	0.348	0.590	0.625
CNN	0.318	0.564	0.224

The deep learning techniques provide the hidden representation of features in the air quality prediction. DL provides features for the classifier for better representation of input data. The Bi-LSTM and BSMO model are commonly applied for feature extraction techniques for fair comparison. The CNN model has lower error value compared to other DL models, because other models have a higher number of convolution, and pooling layers that cause the overfitting problem in the classification. The CNN feature extraction with the BSMO and BILSTM obtained 0.318 MSE, 0.564 RMSE, and 0.224 MAE, whereas the existing ResNet reached 0.348 MSE, 0.590 RMSE, and 0.625 MAE, respectively. The BSMO technique has been compared with existing feature selection techniques in terms of error measure. Their results are shown in Table 2.

Table 2. Feature selection techniques in air quality prediction.

Methods	MSE	RMSE	MAE
PSO	1.181	1.087	0.29
GO	1.169	1.081	0.321
WOA	0.783	0.885	0.189
SMO	0.485	0.696	0.227
BSMO-BILSTM	0.318	0.564	0.225

The BSMO technique achieves higher performance than existing feature selection methods. The BSMO technique has a balancing factor that maintains a tradeoff between exploration and exploitation. This process helps to overcome the local optima trap and reduces overfitting in the classification process. The existing feature selection techniques have limitations of the local optima trap and lower convergences in the feature selection. The BSMO-BILSTM technique obtained 0.318 MSE, 0.564 RMSE, and 0.224 MAE, whereas the WOA method reached 0.783 MSE, 0.885 RMSE, and 0.189 MAE, respectively, in air quality prediction. The BSMO-BILSTM model has been compared with various classifiers for air quality prediction. Their results are shown in Table 3. The classifiers, such as Support Vector Machine (SVM), Random Forest (RF), and K Nearest Neighbors (KNN), were compared with the proposed BSMO-BILSTM.

Table 3. Classifiers in air quality prediction.

Methods	MSE	RMSE	MAE
SVM	1.453	1.205	1.098
RF	0.982	0.991	0.911
KNN	0.816	0.903	0.894
LSTM	0.682	0.826	0.842
BSMO-BILSTM	0.318	0.564	0.225

The KNN model is sensitive to the outlier, whereas the RF model has an overfitting problem and SVM has an imbalance data problem. The LSTM model has an overfitting and vanishing gradient problem that affects performance. The BSMO technique selects the features based on the balancing factor in order to avoid overfitting, and increase learning performance. The BSMO-BILSTM model has higher performance due to an effective feature selection process. The BSMO-BILSTM obtained 0.318 MSE, 0.564 RMSE, and 0.224 MAE, whereas the KNN reached 0.816 MSE, 0.903 RMSE, and 0.894 MAE, respectively. The BSMO-BILSTM model has been compared with existing methods in air quality prediction. Their results are shown in Table 4.

Table 4. Comparison in air quality prediction.

Methods	MSE	RMSE	MAE
EAG-CNN [19]	1.081	1.040	1.104
RBFANN [20]	0.776	0.881	1.049
ARMA-LSTM [21]	0.747	0.864	0.912
Bi-LSTM [23]	0.713	0.844	0.909
Attention LSTM [24]	0.699	0.836	0.892
BSMO-BILSTM	0.318	0.564	0.225

From Table 4, The BSMO-BILSTM model has a lower error value than the existing models in air quality prediction. The BSMO method offers an advantage of applying balancing factor to maintain the exploration–exploitation feature selection that helps to overcome the local optima trap and overfitting problem. The existing Attention LSTM [24] method suffers from the vanishing gradient problem, BI-LSTM [23] has lower efficiency feature representation, whereas ARIMA-LSTM [21] and EAG-CNN [19] have an overfitting problem. The BSMO-BILSTM model obtained 0.318 MSE, 0.564 RMSE, and 0.224 MAE, whereas the Attention LSTM [24] reached 0.699 MSE, 0.836 RMSE, and 0.892 MAE, respectively.

Discussion

For the air quality prediction procedure, a variety of models were used, however the majority of them had issues with overfitting. By deleting pointless features from extracted features, the efficient feature selection technique, known as Balance Spider Monkey Optimization (BSMO), helps to address overfitting in classifiers.

In this collected dataset (two times a week during a 24 h time period), these pollutants were monitored, giving 104 observations annually. The variables considered for this study were: SO_2, NO_x, and $PM_{2.5}$. The Central Pollution Control Board provided the statistics, which were gathered in four Indian cities between 2016 and 2022, namely Bangalore, Chennai, Hyderabad, and Cochin. The suggested BSMO-BILSTM was compared to classifiers including: Support Vector Machine (SVM), Random Forest (RF), and K Nearest Neighbors (KNN). According to our findings, the Attention LSTM model achieved 0.699 MSE, 0.836 RMSE, and 0.892 MAE, whereas the proposed BSMO-BILSTM model attained 0.318 MSE, 0.564 RMSE, and 0.224 MAE, respectively.

6. Conclusions

Currently, air quality predictions have become a crucial task, particularly in developing nations. The deep learning-based prediction technologies have been shown to be more effective than conventional methods for researching these contemporary threats. DL algorithms are capable of handling difficult analyses required to perform accurate and reliable predictions from such large environmental data. Yet, DL techniques applied in air quality prediction have an overfitting problem due to irrelevant feature selection. This study proposes the BMSO-BILSTM model to select relevant features based on the balancing factor in order to overcome this problem. The Balanced Spider Monkey Optimization (BMSO) method maintains exploration and exploitation to select relevant features for further classification. The BMSO-BILSTM model obtains higher performance in air quality prediction in

case of data from four cities analyzed in this paper. The CNN model was used to provide a hidden representation of the input dataset and the BILSTM model enables efficient classification. The BSMO-BILSTM model obtained 0.318 MSE, 0.564 RMSE, and 0.224 MAE, whereas the Attention LSTM reached 0.699 MSE, 0.836 RMSE, and 0.892 MAE, respectively.

Air quality prediction is extremely useful for governments to control pollution, including various human health and sustainability issues. The findings of this investigation can be further enhanced. In future studies, it could include, e.g., a hyper parameter optimization in the Bi-LSTM model to reduce the error of prediction. Results of this study may support various governmental and non-government institutions focused on maintaining a stable and healthy development of both the local community and provinces or states. It could surely help in sustaining a stable advancement and high quality of life of modern-day societies while keeping pollution at the lowest possible level. Still, the process of analyzing data and providing better solutions remains a challenge. In order to analyze large amounts of data more effectively and efficiently, make the invisible visible, and extract the hidden data which is crucial to use effective methodologies and procedures. Therefore, in the future, this research will be further extended by analyzing the air quality prediction in some other regions with different methodologies to improve the statistical measures.

References

1. Espinosa, R.; Palma, J.; Jiménez, F.; Kamińska, J.; Sciavicco, G.; Lucena-Sánchez, E. A time series forecasting based multi-criteria methodology for air quality prediction. *Appl. Soft Comput.* **2021**, *113*, 107850. [CrossRef]
2. Fan, S.; Hao, D.; Feng, Y.; Xia, K.; Yang, W. A hybrid model for air quality prediction based on data decomposition. *Information* **2021**, *12*, 210. [CrossRef]
3. Benhaddi, M.; Ouarzazi, J. Multivariate time series forecasting with dilated residual convolutional neural networks for urban air quality prediction. *Arab. J. Sci. Eng.* **2021**, *46*, 3423–3442. [CrossRef]
4. Zhang, K.; Zhang, X.; Song, H.; Pan, H.; Wang, B. Air quality prediction model based on spatiotemporal data analysis and metalearning. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 9627776. [CrossRef]
5. Yang, Y.; Mei, G.; Izzo, S. Revealing influence of meteorological conditions on air quality prediction using explainable deep learning. *IEEE Access* **2022**, *10*, 50755–50773. [CrossRef]
6. Srinivas, M.; Roy, D.; Mohan, C.K. Discriminative feature extraction from x-ray images using deep convolutional neural networks. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; IEEE: Shanghai, China, 2016; pp. 917–921.
7. Ijjina, E.P.; Mohan, C.K. Human action recognition based on recognition of linear patterns in action bank features using convolutional neural networks. In Proceedings of the 2014 13th International Conference on Machine Learning and Applications, Detroit, MI, USA, 3–6 December 2014; IEEE: Detroit, MI, USA, 2014; pp. 178–182.
8. Saini, R.; Jha, N.K.; Das, B.; Mittal, S.; Mohan, C.K. ULSAM: Ultra-lightweight subspace attention module for compact convolutional neural networks. In Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), Snowmass Village, CO, USA, 1–5 March 2020; IEEE: Snowmass Village, CO, USA, 2020; pp. 1616–1625.
9. Deepak, K.; Chandrakala, S.; Mohan, C.K. Residual spatiotemporal autoencoder for unsupervised video anomaly detection. *SIViP* **2021**, *15*, 215–222. [CrossRef]
10. Roy, D.; Murty, K.S.R.; Mohan, C.K. Unsupervised universal attribute modeling for action recognition. *IEEE Trans. Multimed.* **2019**, *21*, 1672–1680. [CrossRef]

11. Perveen, N.; Roy, D.; Mohan, C.K. Spontaneous expression recognition using universal attribute model. *IEEE Trans. Image Process.* **2018**, *27*, 5575–5584. [CrossRef]

12. Roy, D.; Ishizaka, T.; Mohan, C.K.; Fukuda, A. Vehicle trajectory prediction at intersections using interaction based generative adversarial networks. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; IEEE: Auckland, New Zealand, 2019; pp. 2318–2323.

13. Roy, D.; Mohana, C.K. Snatch theft detection in unconstrained surveillance videos using action attribute modelling. *Pattern Recognit. Lett.* **2018**, *108*, 56–61. [CrossRef]

14. Wang, J.; Xu, W.; Zhang, Y.; Dong, J. A novel air quality prediction and early warning system based on combined model of optimal feature extraction and intelligent optimization. *Chaos Solit. Fract.* **2022**, *158*, 112098. [CrossRef]

15. Huang, Y.; Yu, J.; Dai, X.; Huang, Z.; Li, Y. Air-quality prediction based on the EMD–IPSO–LSTM combination model. *Sustainability* **2022**, *14*, 4889. [CrossRef]

16. Kothandaraman, D.; Praveena, N.; Varadarajkumar, K.; Madhav Rao, B.; Dhabliya, D.; Satla, S.; Abera, W. Intelligent forecasting of air quality and pollution prediction using machine learning. *Adsorpt. Sci. Technol.* **2022**, *2022*, 5086622. [CrossRef]

17. Zhang, Z.; Zeng, Y.; Yan, K. A hybrid deep learning technology for PM2.5 air quality forecasting. *Environ. Sci. Pollut. Res.* **2021**, *28*, 39409–39422. [CrossRef] [PubMed]

18. Zhang, L.; Liu, P.; Zhao, L.; Wang, G.; Zhang, W.; Liu, J. Air quality predictions with a semi-supervised bidirectional LSTM neural network. *Atmos. Pollut. Res.* **2021**, *12*, 328–339. [CrossRef]

19. Ragab, M.G.; Abdulkadir, S.J.; Aziz, N.; Al-Tashi, Q.; Alyousifi, Y.; Alhussian, H.; Alqushaibi, A. A novel one-dimensional CNN with exponential adaptive gradients for air pollution index prediction. *Sustainability* **2020**, *12*, 10090. [CrossRef]

20. Hashim, N.M.; Noor, N.M.; Ul-Saufie, A.Z.; Sandu, A.V.; Vizureanu, P.; Deák, G.; Kheimi, M. Forecasting daytime ground-level ozone concentration in urbanized areas of Malaysia using predictive models. *Sustainability* **2022**, *14*, 7936. [CrossRef]

21. Sun, Y.; Liu, J. AQI prediction based on CEEMDAN-ARMA-LSTM. *Sustainability* **2022**, *14*, 12182. [CrossRef]

22. Ul-Saufie, A.Z.; Hamzan, N.H.; Zahari, Z.; Shaziayani, W.N.; Noor, N.M.; Zainol, M.R.R.M.A.; Sandu, A.V.; Deak, G.; Vizureanu, P. Improving air pollution prediction modelling using wrapper feature selection. *Sustainability* **2022**, *14*, 11403. [CrossRef]

23. Mao, W.; Wang, W.; Jiao, L.; Zhao, S.; Liu, A. Modeling air quality prediction using a deep learning approach: Method optimization and evaluation. *Sustain. Cities Soc.* **2021**, *65*, 102567. [CrossRef]

24. Zou, X.; Zhao, J.; Zhao, D.; Sun, B.; He, Y.; Fuentes, S. Air quality prediction based on a spatiotemporal attention mechanism. *Mob. Inf. Syst.* **2021**, *2021*, 6630944. [CrossRef]

25. Seng, D.; Zhang, Q.; Zhang, X.; Chen, G.; Chen, X. Spatiotemporal prediction of air quality based on LSTM neural network. *Alex. Eng. J.* **2021**, *60*, 2021–2032. [CrossRef]

26. Ge, L.; Wu, K.; Zeng, Y.; Chang, F.; Wang, Y.; Li, S. Multi-scale spatiotemporal graph convolution network for air quality prediction. *Appl. Intell.* **2021**, *51*, 3491–3505. [CrossRef]

27. Janarthanan, R.; Partheeban, P.; Somasundaram, K.; Navin Elamparithi, P. A deep learning approach for prediction of air quality index in a metropolitan city. *Sustain. Cities Soc.* **2021**, *67*, 102720. [CrossRef]

28. Asgari, M.; Yang, W.; Farnaghi, M. Spatiotemporal data partitioning for distributed random forest algorithm: Air quality prediction using imbalanced big spatiotemporal data on spark distributed framework. *Environ. Technol. Innov.* **2022**, *27*, 102776. [CrossRef]

29. Dun, A.; Yang, Y.; Lei, F. A novel hybrid model based on spatiotemporal correlation for air quality prediction. *Mob. Inf. Syst.* **2022**, *2022*, 9759988. [CrossRef]

30. Ma, Z.; Mei, G.; Cuomo, S.; Piccialli, F. Heterogeneous data fusion considering spatial correlations using graph convolutional networks and its application in air quality prediction. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 3433–3447. [CrossRef]

31. Central Pollution Control Board. Available online: https://cpcb.nic.in (accessed on 9 November 2022).

32. Zhang, Z.; Tian, J.; Huang, W.; Yin, L.; Zheng, W.; Liu, S. A haze prediction method based on one-dimensional convolutional neural network. *Atmosphere* **2021**, *12*, 1327. [CrossRef]

33. Yu, C.; Han, R.; Song, M.; Liu, C.; Chang, C.I. A simplified 2D-3D CNN architecture for hyperspectral image classification based on spatial–spectral fusion. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 2485–2501. [CrossRef]

34. Niu, M.; Lin, Y.; Zou, Q. SgRNACNN: Identifying SgRNA on-target activity in four crops using ensembles of convolutional neural networks. *Plant Mol. Biol.* **2021**, *105*, 483–495. [CrossRef]

35. Kumar, S.; Sharma, B.; Sharma, V.K.; Sharma, H.; Bansal, J.C. Plant leaf disease identification using exponential spider monkey optimization. *Sustain. Comput. Inform. Syst.* **2020**, *28*, 100283. [CrossRef]

36. Khare, N.; Devan, P.; Chowdhary, C.; Bhattacharya, S.; Singh, G.; Singh, S.; Yoon, B. SMO-DNN: Spider monkey optimization and deep neural network hybrid classifier model for intrusion detection. *Electronics* **2020**, *9*, 692. [CrossRef]

37. Akhand, M.A.H.; Ayon, S.I.; Shahriyar, S.A.; Siddique, N.; Adeli, H. Discrete spider monkey optimization for travelling salesman problem. *Appl. Soft Comput.* **2020**, *86*, 105887. [CrossRef]

38. Singla, P.; Duhan, M.; Saroha, S. An ensemble method to forecast 24-h ahead solar irradiance using wavelet decomposition and BiLSTM deep learning network. *Earth Sci. Inform.* **2022**, *15*, 291–306. [CrossRef] [PubMed]

39. Chen, T.; Xu, R.; He, Y.; Wang, X. Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Syst. Appl.* **2017**, *72*, 221–230. [CrossRef]
40. Aslan, M.F.; Unlersen, M.F.; Sabanci, K.; Durdu, A. CNN-based transfer learning–BiLSTM network: A novel approach for COVID-19 infection detection. *Appl. Soft Comput.* **2021**, *98*, 106912. [CrossRef] [PubMed]