



# AQIPred: A Hybrid Model for High Precision Time Specific Forecasting of Air Quality Index with Cluster Analysis

Farhana Yasmin<sup>1</sup> · Md. Mehedi Hassan<sup>2</sup> · Mahade Hasan<sup>1</sup> · Sadika Zaman<sup>3</sup> · Jarif Huda Angon<sup>3</sup> · Anupam Kumar Bairagi<sup>2</sup> · Yang Changchun<sup>1</sup>

Received: 23 March 2023 / Accepted: 31 July 2023 / Published online: 7 August 2023  
© The Author(s) 2023

## Abstract

The discipline of forecasting and prediction is witnessing a surge in the application of these techniques as a direct result of the strong empirical performance that approaches based on machine learning (ML) have shown over the past few years. Especially to predict wind direction, air and water quality, and flooding. In the context of doing this research, an MLP-LSTM Hybrid Model was developed to be able to generate predictions of this nature. An investigation into the Beijing Multi-Site Air-Quality Data Set was carried out in the context of an experiment. In this particular scenario, the model generated MSE values that came in at 0.00016, MAE values that came in at 0.00746, RMSE values that came in at 13.45, MAPE values that came in at 0.42, and  $R^2$  values that came in at 0.95. This is an indication that the model is functioning effectively. The conventional modeling techniques for forecasting, do not give the level of performance that is required. On the other hand, the results of this study will be useful for any type of time-specific forecasting prediction that requires a high level of accuracy.

**Keywords** Modeling · Prediction · Time series forecasting · Neural network

## 1 Introduction

At the turn of the twenty-first century, the advent of vast amounts of data, effective supercomputers with graphics processing units (GPU), and scholarly interest in fresh emerging methodologies turned out to be key factors in the development of machine learning [1]. Although many AI approaches date back to the 1960s and have been studied extensively since then, the last few years are considered the “golden era” of AI and machine learning. The term “golden age” wasn’t coined until the 1990s. In recent decades, the volume of data and the ability to compute have grown significantly, leading to the current scenario [2]. The integration of machine learning in health, agriculture, and environmental problem-solving empowers professionals in these fields

to leverage data-driven insights for better decision-making, improved resource allocation, and enhanced problem-solving capabilities [3–6]. As technology continues to advance, machine learning will continue to play a pivotal role in addressing complex challenges and driving innovation in these important domains.

Significant progress has been made in machine learning (ML), which has led to the development of innovative methods such as autoregressive neural networks. These developments have begun to have a positive impact on forecasting [7–9]. In past M4 and M5 tournaments, ML-based methods performed well. M4 machine learning competition winners were neural network-based techniques and gradient boosting. These strategies outperform benchmarks in estimating future uncertainty [10].

Artificial neural networks (ANN) [11]. The best SR ANN model uses backpropagation (BP) and meteorological data. Malaysian data is used. Levenberg-Marquardt, SCG, and Bayesian regularization train the model (BR). BP algorithm with weather data builds the most accurate ANN model. Wind temperature has a negligible impact on temperature, relative humidity, and SR. BR-trained ANN models outperformed others.

Md. Mehedi Hassan  
mehedihsan@ieee.org; mh\_ashiq@yahoo.com

<sup>1</sup> School of Computer Science and Artificial Intelligence, Changzhou University, Changzhou 213164, China

<sup>2</sup> Computer Science and Engineering Discipline, Khulna University, Khulna 9208, Bangladesh

<sup>3</sup> Computer Science and Engineering, North Western University, Khulna 9100, Bangladesh

Deep learning (DL) [12], this work created an LSTM-MVO hybrid intelligence model to forecast and assess CCPP air pollution. The suggested approach uses a long short-term memory model to predict CO<sub>2</sub> and SO<sub>2</sub> emissions. MVO improves LSTM accuracy. The model was tested using Iranian power plant data. May–September 2019 data include wind speed, air temperature, NO<sub>2</sub>, and SO<sub>2</sub>.

Zhao et al. [13] use deep learning to investigate how lockdowns affect AQI. Social and spatiotemporal implications are considered along with historical pollution concentrations and climate. SAC, which combines temporal and geographical autocorrelation, specifically accounts for previous data and nearby cities. Deep learning research revealed: Wuhan's closure cost 25.88, and Shanghai's 20.47. AQI projections can be improved by 47% for Wuhan and 67% for Shanghai by lowering prediction errors.

Integrated AQI forecasting models improve urban air pollution control, public health infrastructure, and resident travel planning. Integrated forecasting models predict frequency subsequences. ELM and WOA-LSTM predict different high-low and trend sequences. Beijing decomposition improves RMSE, MAPE, and PA by 8.55, 10.36, and 6.1% [14].

Azhari et al. [15], cover the most significant field advancements from 2011 through 2021. They examined 155 items after a thorough search of major scientific resources. Machine learning, assessment metrics, predictor factors, predicted values and geographic distribution characterize the studies. This satellite-based study [16], analyses India's COVID-19 lockdowns and AQI changes. Covered are cutting-edge statistical and deep learning methods for short-term AQI forecasting.

To prevent overfitting and the local optima trap, Aarthi et al. [17] suggest using balanced spider monkey optimization (BSMO) for feature selection. The Central Pollution Control Board (CPCB) provided air quality estimates for Bangalore, Chennai, Hyderabad, and Cochin, four Indian cities. Missing data are filled by min–max normalization. The input dataset is deepened using a CNN. Based on the balancing factor, BSMO chooses pertinent properties for the bi-directional long short-term memory (Bi-LSTM) model. Four city air quality time series are predicted using Bi-LSTM.

An IVLSTM-MCMR air quality prediction model is presented as a novel approach [18]. The model recommends MCMR and IVLSTM modules. The VLSTM inner structure is strengthened by the IVLSTM module, which lowers convergence acceleration parameters. A novel historical knowledge technique is introduced to guarantee consistency in training. The multichannel data input model (MC) with increased linear similarity dynamic temporal warping is used by the MCMR module to choose the IVLSTM input data. A multi-route output model (MR) outputs the results of several target stations

with different attributes via various routes to incorporate MC findings.

Deep learning and machine learning are well liked because they excel at gathering, calculating, and comprehending complex dependencies and multidimensional data better than domain expertise. Gated recurrent units and an attention temporal graph convolutional network are used [19]. Support vector machines and neural networks predict air quality index (AQI) contamination in many investigations in this study. Air pollution from CPCB. The suggested ML model predicts and compares Delhi Air Quality Index (AQI) data [20]. Duan et al. [21] investigate Chongqing weather. Then splits sample data into training and test sets using the wavelet transform to verify the Naive Bayes Model. Second, they compare Naive Bayes, SVM, XGBoost, bagging, and random forest. Finally, Chongqing's Naive Bayes Model reliably assesses city air quality.

However, there are very few good works that can predict the forecasting of data, especially for the AQI. Therefore, this study developed an MLP-LSTM hybrid model to contribute the AQI predictive forecasting. Moreover, the predictive ability of the model was investigated in this study, which was conducted within the context of artificial intelligence and involved the development of the model. The main challenge with these types of data is the data preprocessing implementation in terms of making the data trainable. The following is a list of the most significant contributions made by the article:

1. Preprocessed the dataset to speed up the model's training and testing processes.
2. Developed the MLP-LSTM Hybrid Model and built it.
3. Utilized the forecasting prediction model on the selected dataset.
4. Examined the efficiency of the proposed model and contrasted its findings with earlier research

In the section where we examine studies that are similar to this one, we have touched briefly on past research that is pertinent. Following that, we described the research approach that we used. In this section, we discussed the characteristics of the dataset as well as the model that was suggested. After that, our suggested model's outcomes were evaluated during the stage labeled Sect. 6. In the end, we compared the results of our model to those of older models that were still in circulation and found our results to be superior.

## 2 Related Work

Asghari et al. [22] a hybrid method was used to estimate PM<sub>10</sub> air pollution in Tehran, combining data from the Aghdasiyeh and Mehrabad weather stations. The method

employed an artificial neural network with back propagation (BP) and 11 inputs for daily PM10 prediction. Additionally, a hybrid approach called BP-GA, combining Genetic Algorithm (GA) with BP, was utilized. Comparing the method's performance with basic artificial neural networks using BP, the results showed that BP-GA achieved a higher  $R^2$  value of 0.55, indicating better accuracy. However, it should be noted that the model's reliability decreased for longer time frames due to the impact of long-term data fluctuations on network performance. Further improvements are necessary to address these limitations and enhance the model's effectiveness for longer-term predictions. Zhao et al. [23] established a model STCNN-LSTM, which is a spatiotemporal collaborative prediction model for regional air quality. It combines CNN and LSTM for improved accuracy. The model uses a Relevance Data Cube to analyze and visualize air quality dimensions. It achieves an  $R^2$  value of 0.70. However, there are potential limitations such as model complexity, challenges in data preprocessing, and the need for further validation with diverse datasets. Pang et al. [24] compared three machine learning techniques for PM2.5 concentration forecasting (MART, DFNN, and LSTM), and the LSTM model exhibited superior performance. It effectively captured temporal relationships in the data, resulting in an RMSE of  $8.91 \mu\text{g m}^{-3}$  and an MAE of  $6.21 \mu\text{g m}^{-3}$ . However, it is important to note that the model only predicted 75% of pollution levels and explained 80% of the variability ( $R^2=0.8$ ) in PM2.5 concentrations. While the LSTM approach shows promise, there may still be room for improvement in accurately forecasting and reducing air pollution using machine learning techniques.

Zhang et al. [25] compared linear regression, LSTM-FCN, and LightGBM to modify the GRAPES-3 km model's temperature forecast in Shaanxi, China. Evaluation measures included forecast accuracy, RMSE, MAE, and  $R^2$ . RMSE, MAE, and  $R^2$  were reduced by 33%, 32%, and 40%, respectively. All three machine learning techniques are over 78% accurate. LightGBM's accuracy is 84%. Kardhana et al. [26] using Sadewa data, LSTM-RNN estimates the Katulampa Barrage water level. 44 Sadewa covers Katu-lampa. 88, 1616, and 2828. The latter crosses Java north-to-south. Sadewa has 30.8, 0.05-degree cells. LSTM-RNN accurately predicts water level. The model estimated Katulampa's water level using 44 and 88 extents and recurring  $t$ -24 h data. 24 h of model predictions kept  $R^2$  above 0.80. Won et al. [27] model-driven methods were utilized to develop a physical model with interconnected inland river and flood control systems. The rainfall-runoff model was calibrated using gauging stations and pump stations in August 2020. Using model-driven rainfall scenarios, flood alerts were examined. Urban flood forecasting and warning systems use ANN, LSTM, Stack-LSTM, and Bidirectional LSTM. A 30-min

water level forecast using a bidirectional LSTM system had an  $R^2$  of 0.9.

Ouma et al. [28] three sub-basins signify three discharge stations. Each LSTM and WNN hidden layer comprises 30 neurons. LSTM and WNN accurately predicted basin runoff with 0.89 and 0.88  $R^2$  values. MAE for the expected monthly rainfall trend was 9–11 mm, and RMSE was 15–21 mm. Both models reached the lowest RMSE in around the same amount of epochs, with WNN taking somewhat longer. Tuerxun et al. [29] several benchmark functions were used to evaluate the proposed MBES-LSTM model for EMD-based wind power forecasting and to estimate the LSTM's parameters. Revised bald eagle-assisted LSTM hyper-parameter tuning. The updated bald eagle method optimized the LSTM's relevant parameters, resulting in 0.09, 0.07, and 0.01 changes in RMSE, MAE, and  $R^2$ . Shen et al. [30] a framework anticipated daily streamflow in China's Hanjiang River Basin. During the validation period, the integrated framework/DBN model had an NSE of 0.91 and  $R^2$  of 0.93. The integrated framework provides higher streamflow prediction accuracy than the single data-driven model. The combined framework peak flood error was 4.6% lower than the standalone DBN model.

Mani et al. [31] predicted AQI with linear regression and time series. MLR predicts AQI. Regression model using CPCB Chennai data. Sensors verify model parameters. Model validation used RMSE, COD, and MAE. K-cross-fold, 92% MLR. ARIMA forecasts AQI. In parameter evaluation, untimed data were utilized. 95% confident 15-day AQI forecast. 80% of tests passed. Liu et al. [32] an ST-CCN-IAQI model is created. Weather and multi-source air pollution were taken into account. They used dilated convolution and temporal attention stacking to get time-dependent network features. To fine-tune ST-CCN-hyper-parameters, Bayesian optimization was used. They used IAQI's Shanghai air monitoring baselines (AR, MA, ARMA, ANN, SVR, GRU, LSTM, and ST-GCN). For a single station, ST-CCN-RMSE IAQI and MAE decreased by 24.95% and 16.87%. RMSE and MAE for all nine stations were 9.84 and 7.52, whereas  $R^2$  was 0.90. Rahimpour et al. [33] HSD and HPTD predicted Orumiyeh's 1-day AQI. CEEMDAN combined GRNN and ELM as HSD models using AQI data. CEEMDAN IMFs were divided into 9 VMs before forecasting IMF1. GRNN and ELM-HTPD predicted IMFs. CEEMDAN-ELM and CEEMDAN-GRNN predicted HSD's AQI.  $R^2=0.74$ , RMSE=5.45, MAE=3.87 for CEEMDAN-VMD-GRNN HTPD. HSD predicts AQI is worse than HTPD. Liu et al. [34] Zhangdian District air quality is projected using a random forest model and real-time meteorological emissions. MAE outperforms LSSVM, DT, and BP neural networks. Model inversion decreases waste gas emissions using weather and air quality. 2019 industrial waste gas emissions should be 5687.5 million cubic meters daily. This

methodology reduces air pollution risk by altering enterprise production capacity based on weather forecast inversion.

Fan et al. [35] address WRF dataset O3 and PM2.5 (PNW). Kennewick sends erroneous O3 warnings. Unified modeling reduces Low-O3 ML2  $R^2 = 0.79$  NMB by 7.6%. AQI and high-O3 capture. ML1&2 predict low-PM2. In fire and cold, ML2 showed increased high-PM2.5 and lower NMB. AIRPACT doubles PNW wildfire PM2.5 estimations. ML estimates NW station O3 and PM2.5. A comparative analysis [36] was conducted on deep learning models such as LSTM, GRU, and a statistical model to forecast air pollutants (NO<sub>2</sub>, O<sub>3</sub>, SO<sub>2</sub>, PM2.5, PM10) using a publicly available dataset from a monitoring station in Belfast, Northern Ireland. The deep learning models consistently outperformed the statistical model, achieving the lowest RMSE (0.59) and the highest  $R^2$  score (0.86).

In Table 1, a summarized version of the description of the related work is presented.

### 3 Methodology

In the section on data collection, this study went into considerable depth about the dataset instances as well as the dataset sources. After that, the stage of removing unnecessary data has been finished, which included the removal of data columns with unnecessary values and the deletion of any null values that were there. Examples of pair plots, density maps, and correlation analysis are provided in the section on data visualization so that the dataset can be understood better. The dataset was then prepared for the simulation in the next stage, which was the data preprocessing step. This step included the operations of integer encoding, floating conversion, train test split, min–max scaling, and data reshaping into 3D. Finally, a Hybrid MLP-LSTM technique was developed to accomplish the goals of estimating the air quality index and training the model. After that, conduct an ablation study so that our model can fit best. After this, the findings are assessed by making predictions and computing the MSE, MAE, RMSE, MAPE, and  $R^2$  scores. And made predictions for 6 important features and computed the MSE, MAE, RMSE, MAPE, and  $R^2$  scores as well. Figure 1 illustrates the full process of the proposed Hybrid MLP-LSTM model, with each phase represented by a block.

#### 3.1 Dataset Description

The Beijing Multi-Site Air-Quality (BMSAQ) Dataset [37] is accessible to the general public. The data was acquired from the UC Irvine Machine Learning Repository. The national government-managed 12 locations across the nation provided the hourly data for air contaminants that make up this data set. The Beijing Municipal Environmental Monitoring

Center is where the data on air quality was gathered. The meteorological data gathered at each site for monitoring air quality is compared to that of the weather station closest to the site operated by the China Meteorological Administration. The time's start date is March 1, 2013, and its end date is February 28, 2017. Six major air pollutants and six relevant climatic variables are taken into account in this hourly data collection at various places throughout Beijing. The summarized description of the dataset is presented in Table 2.

#### 3.2 Unnecessary Data Removal

To improve the accuracy of the air quality index prediction algorithm, certain columns were removed from the dataset. The wind direction and station columns were excluded based on their limited impact on the prediction, as determined by domain knowledge and previous research findings. Additionally, the date column was removed since there were already date integer columns available, and the AQI column was eliminated due to duplicate numeric AQI columns. These column removals helped streamline the dataset, eliminating redundancy and allowing the algorithm to focus on the most relevant features for accurate air quality index prediction.

Furthermore, any instances with null values were also removed to ensure the integrity and reliability of the data used for prediction. The dataset graph after the removal of unnecessary data is presented in Fig. 2, showcasing the density values of the remaining attributes in different colors. In which  $x$  represents the number of instances and  $y$  represents the values.

#### 3.3 Data Visualization

The instances and characteristics of the dataset have been shown, so that they may be better understood through the presentation of the dataset correlation, the pair plot of the data dataset, and the density of each instance. Whereas a correlation provided a clearer understanding of the linear relationship that existed between the continuous variables. It was beneficial to understand the proper collection of features to explain the relationship among the variables in the dataset by using pair plots. This was accomplished by understanding the suitable set of features. The use of density plots allowed for the examination of how a dataset's variables were distributed. Figure 3 shows the (a) dataset correlation, (b) dataset pair plot, and (c) data density.

To acquire insights and comprehend the relationships and distribution of variables inside a dataset, several data analyses and visualizations are utilized.

Dataset correlation, as shown in Fig. 3a, measures the statistical relationship between variables. It indicates how closely related two variables are and the direction of their

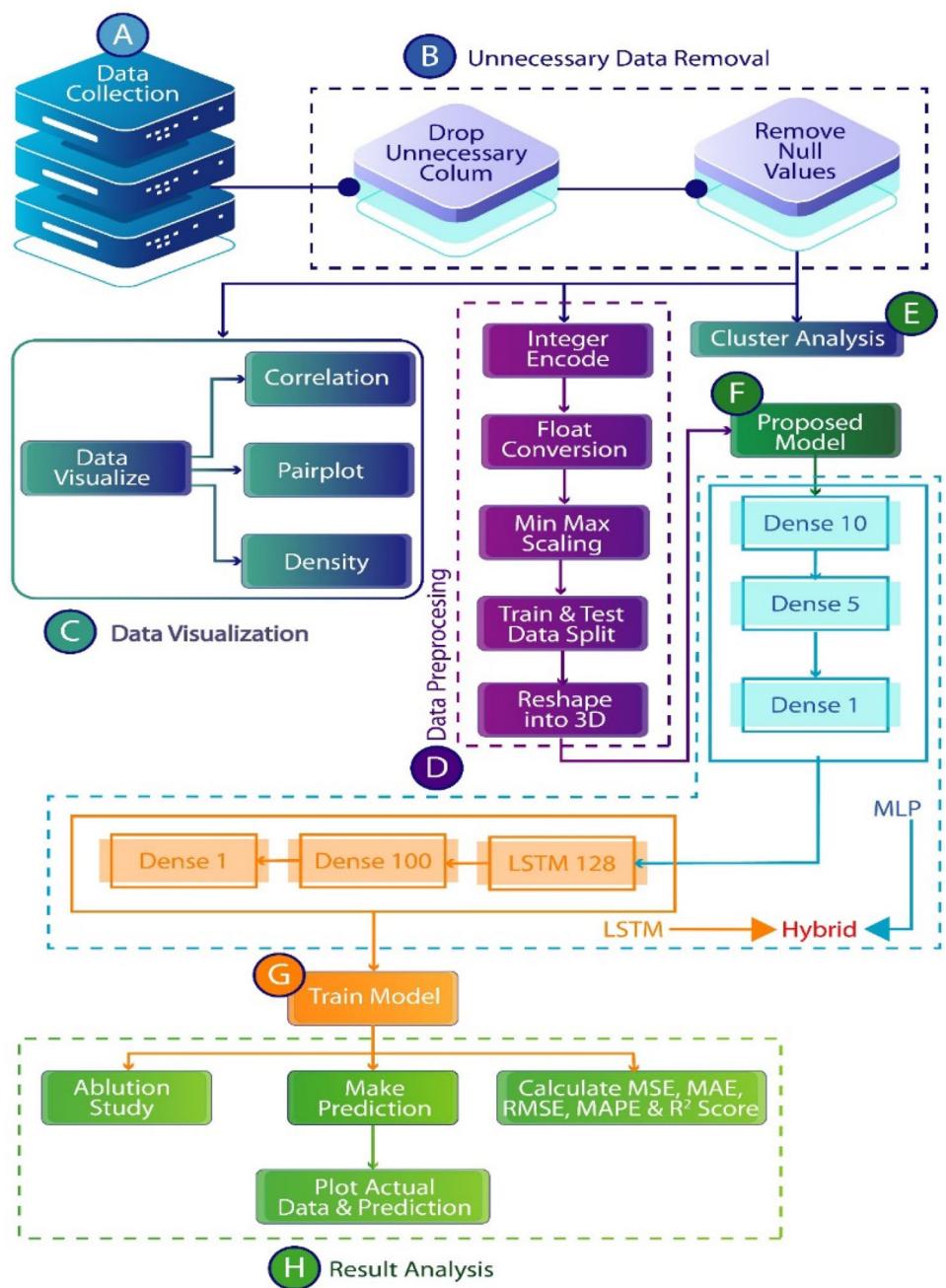
**Table 1** Summarized description of related work

Description	Method	Dataset	Findings and limitations
Asghari et al., (2016) [18]	BP-GA	Aghdasiyeh and Mehrabad weather stations	Reliability decreases for longer time frames due to the impact of long-term data fluctuations on network performance
Zhao et al., (2019) [19]	STCNN-LSTM	Air quality monitor stations in Beijing and Tianjin	The complex model with challenges in data preprocessing
Pang et al., (2019) [24]	LSTM	Tehran Air Quality Control Company	LSTM model shows promise but only predicts 75% of pollution levels and explains 80% of the variability in PM2.5 concentrations
Zhang et al., (2022) [25]	LightGBM	GRAPES-3 km forecasting data	Improvement in forecast accuracy, but limited discussion on model limitations or potential areas for improvement
Kardhana et al., (2022) [26]	LSTM-RNN	LAPAN's Sadewa data and Jakarta DWR data	Limited information on the LSTM-RNN model's limitations for water level estimation
Won et al., (2022) [27]	Bidirectional LSTM	Gauging stations d pump stations urban stream data	Model-driven methods show potential, but limitations related to data availability or uncertainties are not discussed
Ouma et al., (2022) [28]	LSTM WNN	Situ rainfall data and satellite-based meteorological data	Accurate prediction of basin runoff, but limitations regarding data quality and generalizability are not mentioned
Tucrxun et al. (2022) [29]	EMD-MBES-LSTM	SCADA data	Improved performance after hyper-parameter tuning, but limitations related to real-world applications are not discussed
Shen et al., (2022) [30]	DBN SOM-RF-DBN	Daily streamflow at the hydrological Xiantao station data	An integrated framework improves streamflow prediction accuracy, but limitations are not explicitly addressed
Mani et al., (2022) [31]	ARIMA	CPCB data of Chennai, India	MLR and ARIMA models show promise for AQI prediction, but potential limitations in different locations are not discussed
Liu et al., (2022) [32]	ST-CCN-IAQI	Shanghai monitoring air station data	ST-CCN-IAQI model improves prediction accuracy, but limitations related to data availability or model complexity are not mentioned
Rahimpour et al., (2021) [33]	HTPD	Iranian air pollution monitoring system data	CEEMDAN-GRNN HTPD model shows promise for AQI prediction, but limitations related to data availability or uncertainties are not discussed
Liu et al., (2022) [34]	Random Forest	Daily Emission Data of Zhangdian District, China	The random forest model outperforms other methods, but limitations related to data availability or uncertainties in predicting industrial waste gas emissions are not discussed
Fan et al., (2022) [35]	ML2	WRF Dataset	The unified modeling approach improves some aspects, but limitations or challenges are not explicitly mentioned
Naz et al., (2023) [36]	DL models	Monitoring station in Belfast, Northern Ireland	Deep learning models outperform statistical models, but the limitations of deep learning models are not discussed in detail

relationship. The correlation values range from  $-1$  to  $+1$ , where  $-1$  represents a strong negative correlation,  $+1$  represents a strong positive correlation, and  $0$  indicates no correlation. In the figure, it can be observed that Pm2.5 has the strongest positive correlation among all other attributes.

Pair plots, depicted in Fig. 3b, provide a visual representation of the relationship between pairs of variables. Each pair is represented as a scatter plot, with one variable's values on the  $x$ -axis and the other variable's values on the  $y$ -axis. This graphical representation helps identify patterns, trends, and correlations between variables.

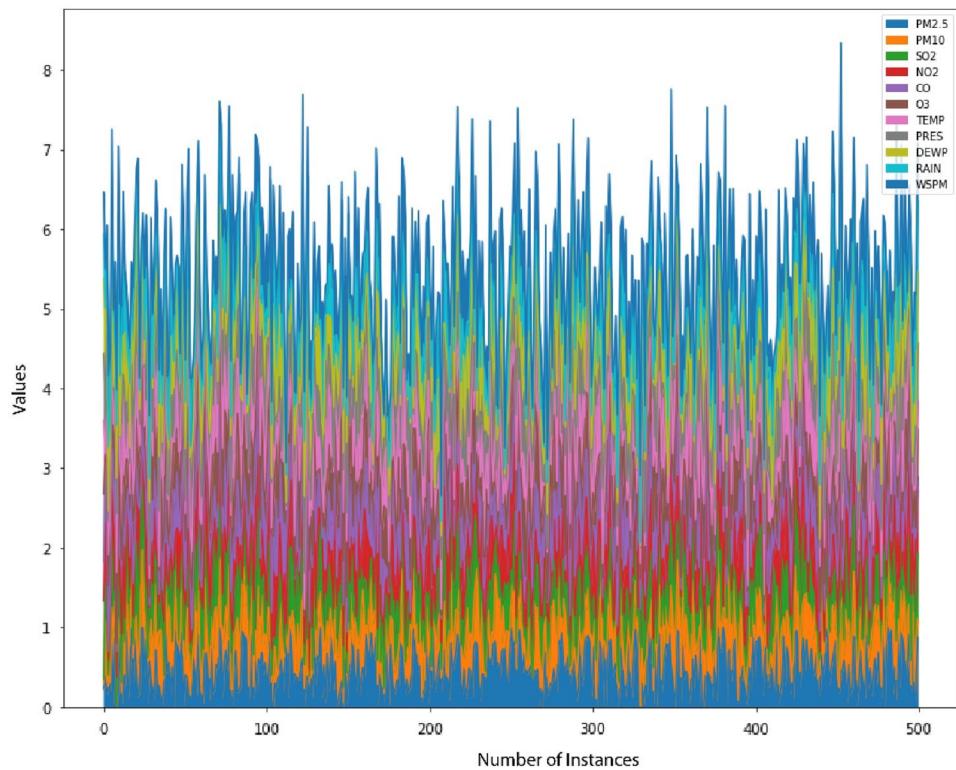
**Fig. 1** Illustrates the full process of the proposed Hybrid MLP-LSTM model, with each phase represented by a block. Block **A** Described dataset instances and data source **B** unnecessary data removal **C** data visualization **D** data preprocessing **E** cluster analysis **F** proposed model **G** train model **H** result analysis



**Table 2** Summarized description of six important features

	PM2.5	PM10	SO2	NO2	CO	O3
Count	418,946	418,946	418,946	418,946	418,946	418,946
Mean	79.19028	104.1894	15.65343	50.59347	1212.644	57.11511
STD	80.0203	91.08931	21.47491	34.60304	1132.051	55.84823
Min	2	2	0.2856	1.0265	100	0.2142
25%	21	36	3	24	500	12
50%	55	82	7	45	900	45
75%	109	144	19	70	1500	80
Max	999	999	500	290	10,000	1071

**Fig. 2** Dataset after unnecessary data



Dataset density, as illustrated in Fig. 3c, showcases the distribution of data values across the dataset. It reveals how the values are spread or concentrated within a certain range. Density plots, often depicted as smooth curves, visualize the shape, peaks, and tails of the distribution, providing insights into the central tendency and spread of the data.

### 3.4 Cluster Analysis

In this study, two different clustering algorithms, namely K-means and Agglomerative Clustering, were employed to determine the optimal number of clusters in the dataset. For K-means, the optimal number of clusters was identified as 4, with a corresponding silhouette score of 902,920,072,962.847. On the other hand, Agglomerative Clustering also yielded 4 as the optimal number of clusters, with a silhouette score of 1,124,108,699,292.519. These results suggest that both algorithms agree on the optimal number of clusters and provide relatively high silhouette scores, indicating well-separated and compact clusters. Figure 4 shows the (a) K-means clustering elbow method, (b) Agglomerative clustering elbow method, and (c) the optimal number of clusters.

## 3.5 Data Pre-processing

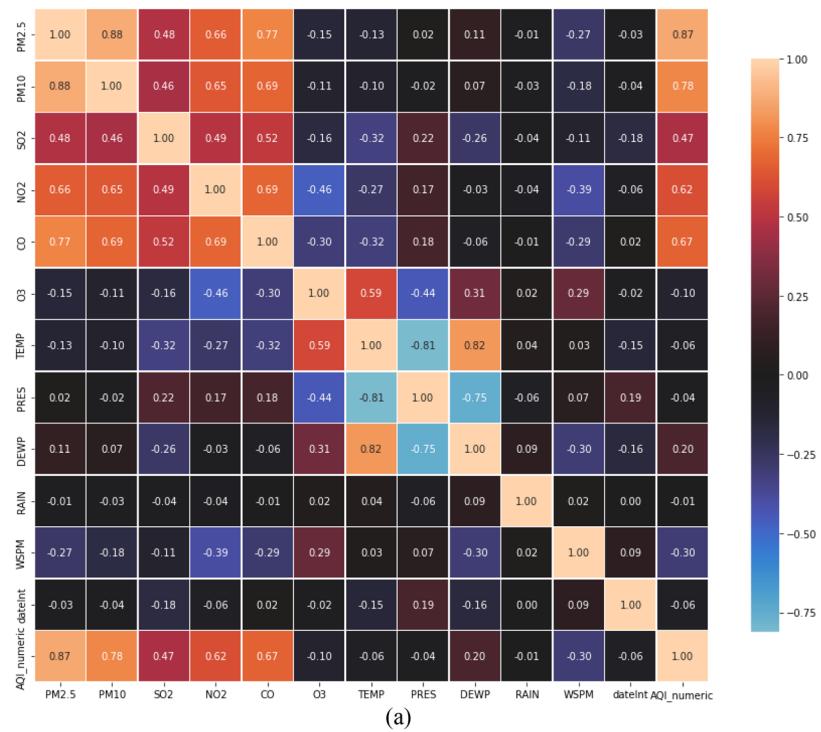
### 3.5.1 Integer Encoding

Numbers were substituted for the category variable that was originally included in the dataset by applying integer encoding. Where numbering is done on a purely arbitrary basis. This enables the models to be specified more expediently. Table 3 shows a short inspection of integer encoding output.

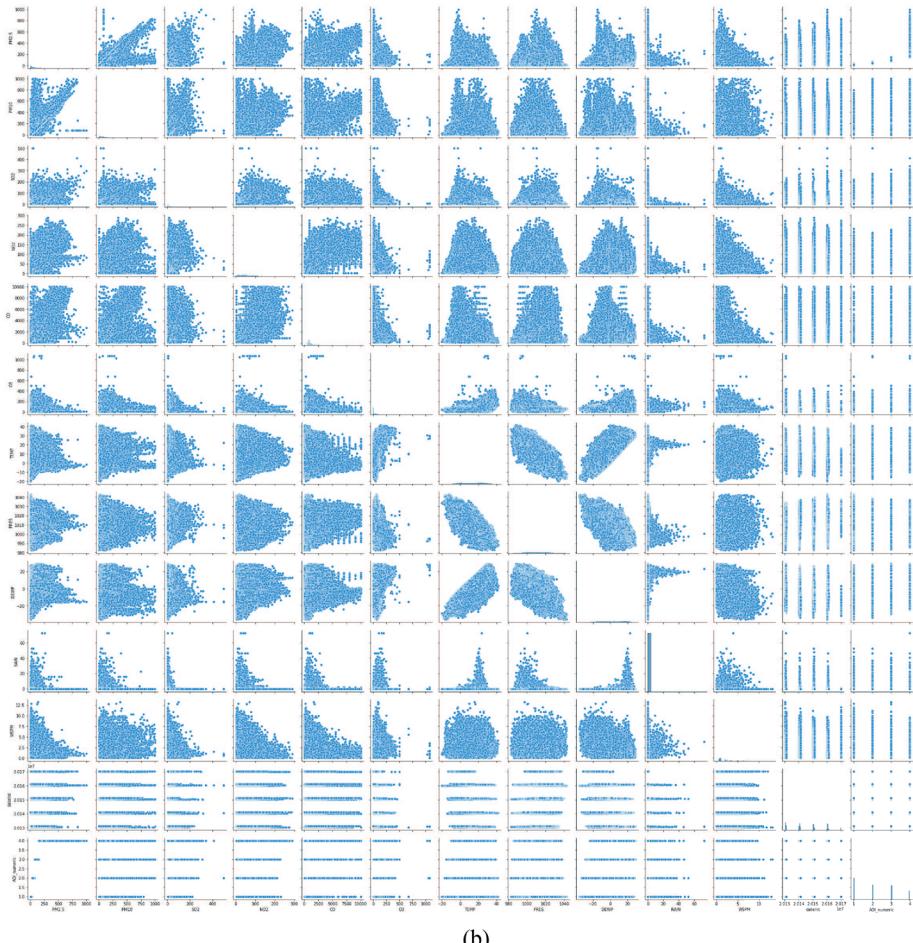
### 3.5.2 Floating Conversion

The processing times of the model are made more reasonable by utilizing floating-point conversion in the systems in our dataset that contain extremely small values and very large numbers. This is done in systems that contain both very large numbers and very small values. Table 4 shows a quick look at the floating conversion implementation output.

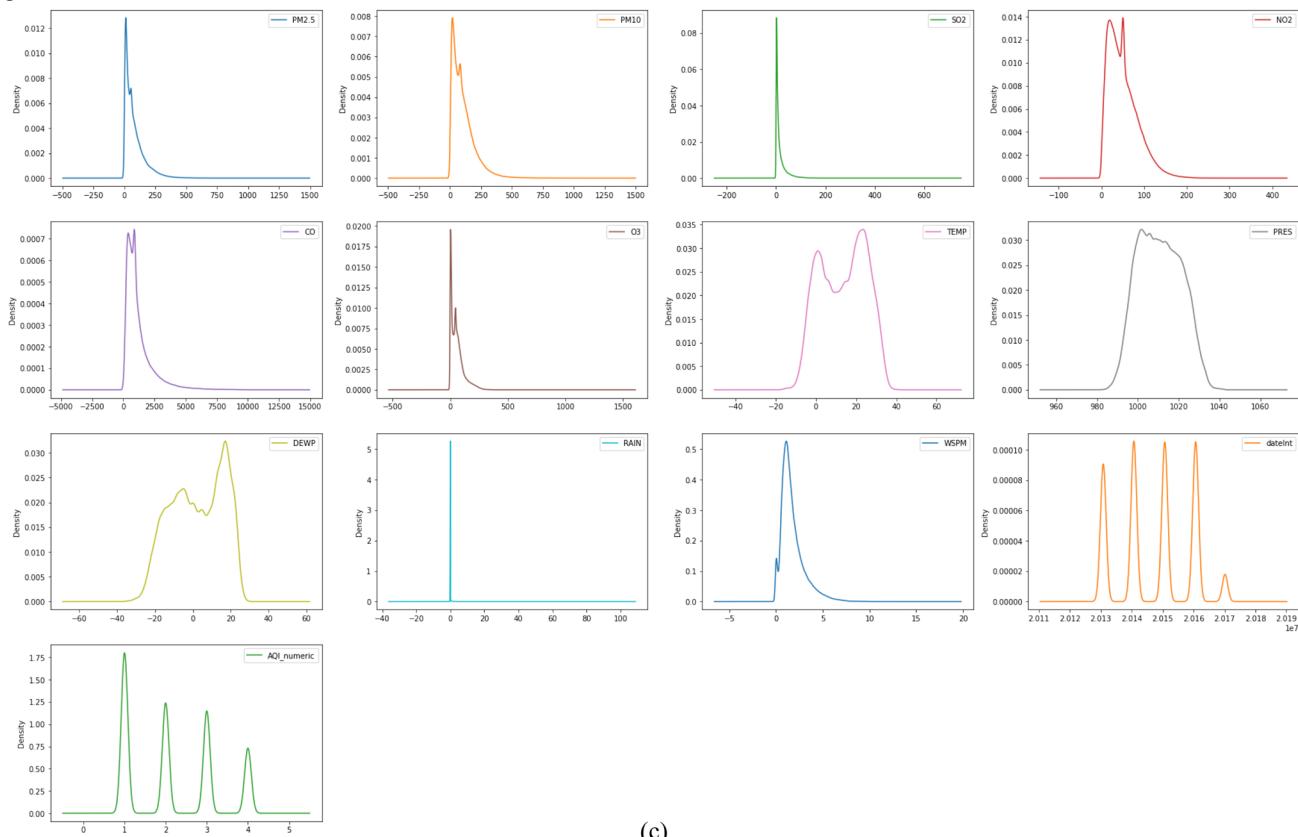
**Fig. 3** **a** Dataset correlation **b** dataset pair plot **c** data density



(a)



(b)

**Fig. 3** (continued)

### 3.5.3 Min–Max Scaling

To reduce the size of the dataset while staying within the parameters of the given range, the Min–Max Scaler was utilized. It modifies our data by adjusting the scale of the features to fit within the specified range. Which adjusts the values to fit within a particular value range without affecting the overall form of the distribution they were based on. Table 5 shows a quick scan of the min–max scaling output.

### 3.6 Train Test Split

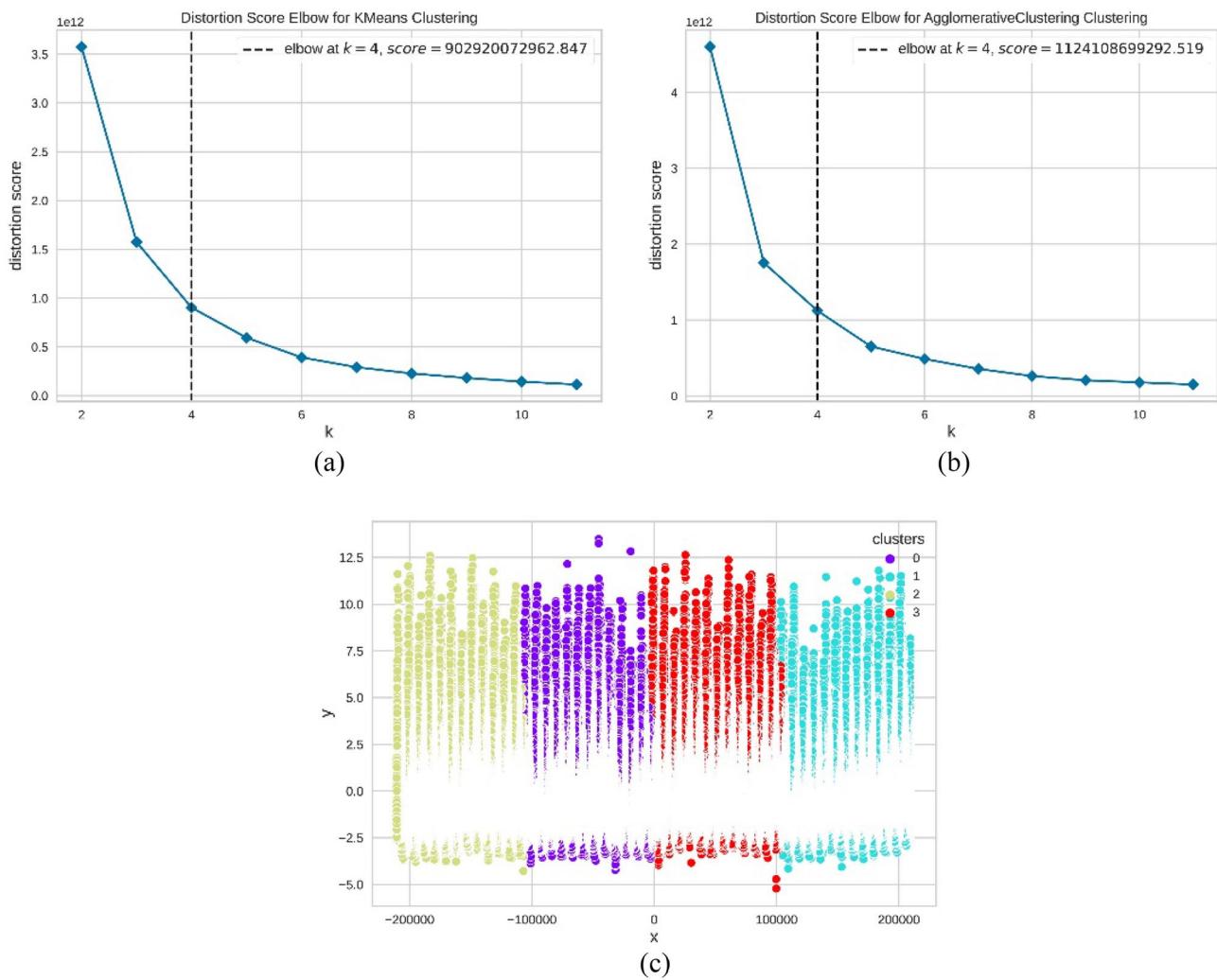
After the normalization process, two arrays are created to hold the test dataset and the training dataset separately. This is done to ensure compatibility with scikit-learn libraries, which typically expect input data in the form of arrays rather than lists. Converting the data into arrays enables effective utilization of these libraries. To train the model, 85% of the data is used for training purposes, while the remaining 15% is used for testing. This split helps evaluate the model's performance on unseen data and assess its generalization capabilities. The train-test split curve, as depicted in Fig. 5, visualizes the distribution of the data between the train and test sets, and shows the proportion of data allocated for each.

## 4 Evolution Metrics

This section gives an overview of the several performance criteria that have been applied in the proposed study to gauge how well-built prediction models function. Among other metrics, these include MSE, MAE, RMSE, MAPE, and  $R^2$  scores. Based on several metrics, the precision of power consumption estimations can be assessed. The accuracy of the forecasts will increase with decreasing error values. Because of this, accuracy is measured by examining the discrepancy between the output that was expected and the actual result. Each accuracy indicator will frequently produce varied results when testing and comparing various prediction models and strategies utilizing the same dataset. This will make the performance unpredictable.

### 4.1 MSE

Estimator's mean squared error measures the average squared difference between estimated and actual values (MSE). MSE calculates the average of error squares. It's a risk function whose value is similar to the expected squared error loss. It's usually always positive rather than zero, which may be due to randomness or the estimator failing to evaluate relevant information.



**Fig. 4** a K-means clustering elbow method, b agglomerative clustering elbow method, and c the optimal number of cluster

**Table 3** Integer encoding output

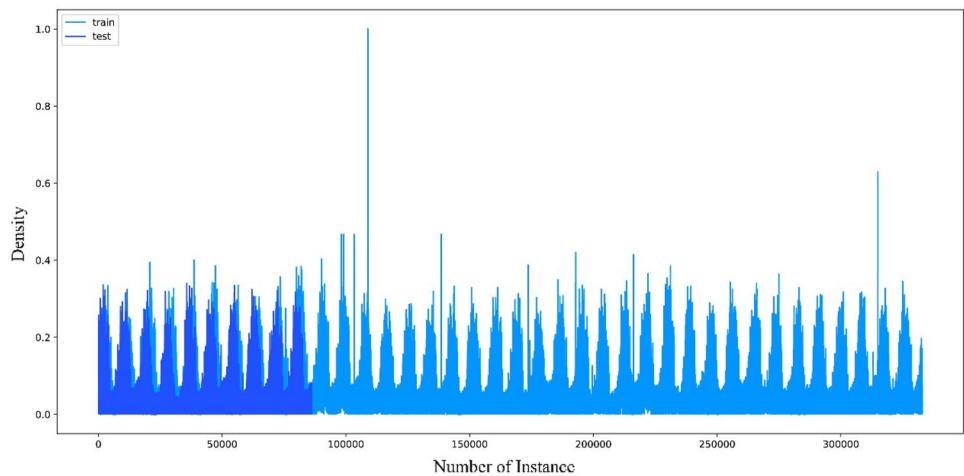
	0	1	2	3	4	5	6	7	8	9	10	11	12
0	4	4	4	7	3	77	-0.7	1023	-18.8	0	4.4	20,130,301	1
1	8	8	4	7	3	77	-1.1	1023.2	-18.2	0	4.7	20,130,301	1
2	7	7	5	10	3	73	-1.1	1023.5	-18.2	0	5.6	20,130,301	1
3	6	6	11	11	3	72	-1.4	1024.5	-19.4	0	3.1	20,130,301	1
4	3	3	12	12	3	72	-2	1025.2	-19.5	0	2	20,130,301	1

**Table 4** Floating conversion output

	0	1	2	3	4	5	6	7	8	9	10	11	12
0	4	4	4	7	3	77	-0.7	1023	-18.8	0	4.4	20,130,300	1
1	8	8	4	7	3	77	-1.1	1023.2	-18.2	0	4.7	20,130,300	1
2	7	7	5	10	3	73	-1.1	1023.5	-18.2	0	5.6	20,130,300	1
3	6	6	11	11	3	72	-1.4	1024.5	-19.4	0	3.1	20,130,300	1
4	3	3	12	12	3	72	-2	1025.2	-19.5	0	2	20,130,300	1

**Table 5** Min–max scaling output

	0	1	2	3	4	5	6	7	8
0	0.002006	0.002006	0.007433	0.020671	0.022901	0.07171	0.312195	0.672186	0.264209
1	0.006018	0.006018	0.007433	0.020671	0.022901	0.07171	0.305691	0.675497	0.273425
2	0.005015	0.005015	0.009434	0.031053	0.022901	0.067974	0.305691	0.680464	0.273425
3	0.004012	0.004012	0.021441	0.034514	0.022901	0.06704	0.300813	0.69702	0.254992
4	0.001003	0.001003	0.023442	0.037974	0.022901	0.06704	0.291057	0.708609	0.253456

**Fig. 5** Train and test split curve

## 4.2 MAE

Mean absolute error is used to evaluate models, especially regression models. The MAE of a model for the test set is obtained by averaging the absolute values of the individual prediction errors over all of the test set occurrences.

## 4.3 RMSE

The root mean square represents the residual standard deviation. Standard deviation quantifies dispersion. Residuals measure data point distance from the regression line. RMSE measures the difference between expected and actual values. It displays how closely the data fit the best-fit line.

## 4.4 MAPE

The mean absolute percentage error (MAPE) is a metric that evaluates how precise a forecasting system is. It subtracts actual values from real values, then divide by actual values. The report gives an accuracy percentage. MAPE is most often used to forecast errors. It's easier to read because the variable's units are in percentages.

## 4.5 $R^2$ Score

$R^2$  score evaluates a regression-based machine learning model. It's called the coefficient of determination. It computes the range of probable dataset outputs to function. It's the difference between actual dataset samples and model predictions.

To contrast the results with other methods, the performance evaluation metrics mean squared error (MSE), mean absolute error (MAE), mean absolute percentage error (MAPE),  $R^2$  score, and root mean square error (RMSE) were used as follows.

$$MSE = \frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2 \quad (1)$$

$$MAE = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t| \quad (2)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2} \quad (3)$$

$$MAPE = \frac{100\%}{n} \sum_{t=1}^n \left| \frac{y_t - \hat{y}_t}{y_t} \right| \quad (4)$$

$$R^2 = 1 - \frac{\sum_{t=1}^n (y_t - \hat{y}_t)^2}{\sum_{t=1}^n (y_t - \bar{y}_t)^2} \quad (5)$$

where  $n$  is the number of prediction points,  $y_t$  represents the actual load,  $\hat{y}_t$  the predicted load, and  $\bar{y}_t$  the average load [38, 39].

## 5 Applied Models

To compare the performance of the proposed hybrid MLP-LSTM models, we applied three different models to the dataset. MLP and LSTM models have also been used separately to further explain how well the hybrid MLP-LSTM performs to forecast the air quality index. Following is a description of the application and performance.

### 5.1 MLP Model

The MLP (Multilayer Perceptron) model with the given architecture consisting of three dense layers (with 10, 5, and 1 units) was trained for 10 epochs. The model utilized a tanh activation function for the intermediate layer. The training process resulted in a mean squared error (MSE) of 0.00157, indicating a relatively low level of prediction error. The mean absolute error (MAE) was found to be 0.02782, representing the average absolute difference between the predicted and actual values. The root mean squared error (RMSE) was calculated as 42.3931, which indicates the standard deviation of the prediction errors. The model achieved an  $R^2$  score of 0.41, suggesting that it explains

**Table 7** Summarized description of MSE, MAE, RMSE, MAPE, and  $R^2$  score

Description	Value
MSE	0.00018
MAE	0.00875
RMSE	14.5032
MAPE	46,053,752,832.00000
$R^2$ Score	0.92

41% of the variance in the target variable. While the model demonstrated promising performance, there is still room for improvement, particularly in reducing the mean absolute percentage error (MAPE) value, which indicates the percentage difference between the predicted and actual values. Table 6 shows the summarized description of the evolution metrics of the MLP model.

### 5.2 LSTM Model

We applied an LSTM model with 128 units on the dataset to predict the air quality index. The model achieved impressive results with an MSE of 0.00018 and an MAE of 0.00875. The MAPE score was 46,053,752,832.00000, indicating a relatively low percentage error. The root mean squared error was 14.5032, which is quite small, indicating accurate predictions. The  $R^2$  score of 0.92 suggests that the model explains 92% of the variance in the data, indicating a strong fit. Overall, the LSTM model demonstrated good performance in predicting the air quality index. Table 7 shows the summarized description of the evolution metrics of the LSTM model.

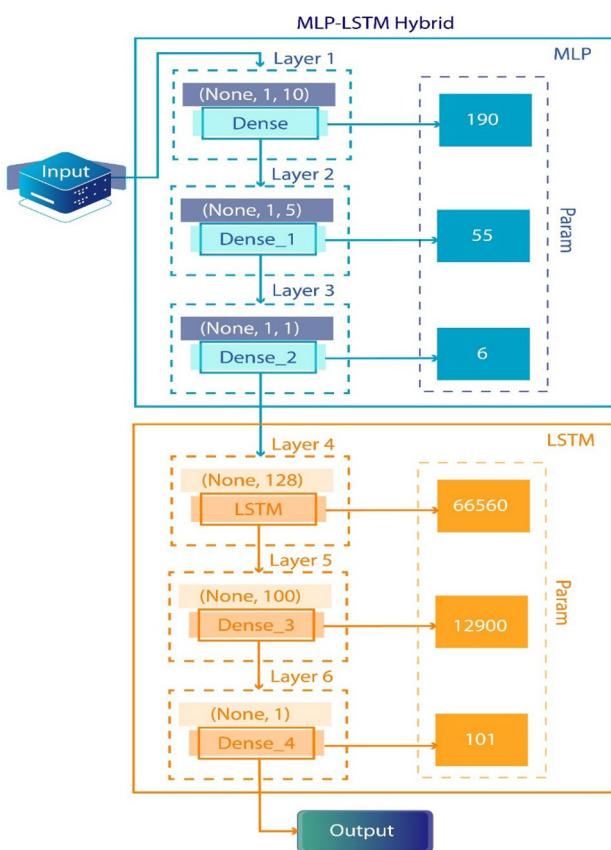
### 5.3 Proposed Model

An MLP-LSTM hybrid model, we have suggested here. Which has a total of six layers in its structure. Both the MLP block and the LSTM block are composed of three layers. The first layer of the MLP block is dense, its output shape was (None, 1, 10) and its parameter value was 190. Layer 2 is also dense. The output shape of Layer 2 was (None, 1, 5) and the parameter value was 55. Layer 3 is also dense, and its output shape is (None, 1, 1), while its parameter value is 6.

An LSTM layer with the output shape of (None, 128) and a parameter value of 66,560 is located in the fourth layer of the LSTM Block. Layer 5 is a dense layer, with the output shape set to (None, 100), and the parameter value set to 12,900. The output shape of Layer 6 was (None, 1) and the value of Parameters was 101. Layer 6 is also dense as shown in Fig. 6.

**Table 6** Summarized description of MSE, MAE, RMSE, MAPE, and  $R^2$  score

Description	Value
MSE	0.00157
MAE	0.02782
RMSE	42.3931
MAPE	112,912,818,176.00000
$R^2$ score	0.41



**Fig. 6** Architecture of Proposed Model. A total of 12 layers are divided into two sections, Where the MLP section is denoted by Blue color and the LSTM section is denoted by Orange color

### 5.3.1 Train Model

To train the MLP-LSTM hybrid model that was presented, the Adam optimizer was used. Where the amsgrad variable was set to False, the learning rate was 0.001, beta 1 was 0.9, and beta 2 was 0.9. The training of the model was done using a variety of different epoch sizes to get the optimum results. Figure 7 shows the Pseudo Code.

### 5.3.2 Hyperparametric Exploration

During the phase that focused on training, the following investigation on ablation was carried out. We have adjusted the values of hyperparameters, such as learning rate, beta values, decay, momentum, rho, and others, to find the best combination that maximizes the model's performance. Adam's optimization algorithm with a learning rate of 0.001, beta values of 0.9, beta 2 of 0.9, and

Amsgrad set to False achieved the highest  $R^2$  score of 0.95. The SGD optimization algorithm with a learning rate of 0.01, decay of 1e-5, momentum of 0.9, and Nesterov set to True achieved an  $R^2$  score of 0.92. Lastly, the SGD optimization algorithm with a learning rate of 0.01 and rho of 0.9 achieved an  $R^2$  score of 0.90. Table 8 shows the description of hyperparameter fine-tuning.

Throughout the experimentation process, different epoch sizes were tested to assess their impact on model performance. A maximum  $R^2$  score of 0.95 was achieved with 10 epochs, indicating a strong correlation between predicted and actual values. The lowest root mean square error (RMSE) of 11.95 was obtained with 30 epochs, demonstrating reduced average deviation. Mean square error (MSE) remained constant at 0.00016 across different epochs, indicating consistent accuracy. Additionally, the best mean absolute error (MAE) and mean absolute percentage error (MAPE) values of 0.00745 and 0.33, respectively, were achieved with 300 epochs, highlighting improved accuracy in absolute and percentage differences. The hyper-parametric exploration is shown in Table 9.

### 5.3.3 Prediction

After carrying out the ablation study, we discovered that the optimal number of epochs for the model is 10. After that, we made a prediction based on the test set and computed the following matrices, which included MSE, MAE, RMSE, MAPE, and  $R^2$  Score. The facts as well as the prediction are presented in Fig. 8.

### 5.3.4 Proposed Model's Performance

The obtained results indicate a good fit of the model for the analysis conducted. The mean squared error (MSE) value of 0.00016 suggests a favorable fit, while the mean absolute error (MAE) value of 0.00746 is deemed appropriate for the analysis, signifying good accuracy. The root mean square error (RMSE) value of 13.45 indicates an overall good fit, and the mean absolute percentage error (MAPE) of 0.42 is considered favorable. Finally, the coefficient of determination  $R^2$  value of 0.95 is fairly positive, reflecting a strong correlation between predicted and actual values. Collectively, these results signify the effectiveness of the model and its ability to accurately represent the analyzed data. The overall performance was best with the 10 epochs described earlier considering the execution time as well. The MSE, MAE, RMSE, MAPE, and  $R^2$  scores have been condensed and presented in Table 10, and Fig. 9 shows the evolution metrics on the x-axis and values on the y-axis as follows.

**Fig. 7** Pseudo code of proposed model

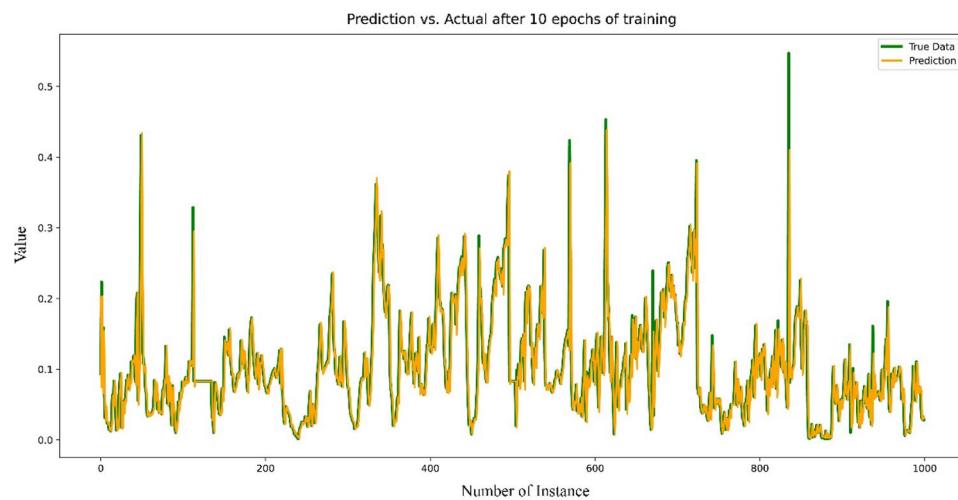
Pseudo Code
<b># Define the layers of the model</b> → dense_layer_1 = Dense(units=10, input_shape=[1, 18]) → dense_layer_2 = Dense(units=5, input_shape=[1]) → dense_layer_3 = Dense(units=1, input_shape=[1]) → lstm_layer = LSTM(128, input_shape=(train_X.shape[1], train_X.shape[2])) → dense_layer_4 = Dense(100, activation='tanh') → dense_layer_5 = Dense(1)
<b># Create a sequential model</b> → model = Sequential()
<b># Add the layers to the model</b> → model.add(dense_layer_1) → model.add(dense_layer_2) → model.add(dense_layer_3) → model.add(lstm_layer) → model.add(dense_layer_4) → model.add(dense_layer_5)
<b># Compile the model with the given parameters</b> → optimizer = Adam(learning_rate=0.001, beta_1=0.9, beta_2=0.9, amsgrad=False) → model.compile(loss='mae', optimizer=optimizer, metrics=['accuracy']) start = time.time()
<b># Train the model on the training data and validate it on the test data</b> → num_epochs = 300 → history = model.fit(train_X, train_y, epochs=num_epochs, validation_data=(test_X, test_y), verbose=1, shuffle=False)
<b># Make predictions on the test set</b> → yhat = model.predict(test_X)
<b># Reshape the test set to match the input shape of the model</b> → test_X = test_X.reshape((test_X.shape[0], test_X.shape[2]))
<b># Combine the predicted values with the remaining test set features</b> → inv_yhat = concatenate((yhat, test_X[:, 1:]), axis=1)
<b># Invert scaling on the predicted values</b> → inv_yhat = scaler.inverse_transform(inv_yhat)
<b># Extract only the first column (the predicted values)</b> → inv_yhat = inv_yhat[:, 0]
<b># Reshape the actual test set values to match the predicted values</b> → test_y = test_y.reshape((len(test_y), 1))
<b># Combine the actual values with the remaining test set features</b> → inv_y = concatenate((test_y, test_X[:, 1:]), axis=1)
<b># Invert scaling on the actual values</b> → inv_y = scaler.inverse_transform(inv_y)
<b># Extract only the first column (the actual values)</b> → inv_y = inv_y[:, 0] end = time.time()

**Table 8** Description of hyperparameter fine-tuning for modeling

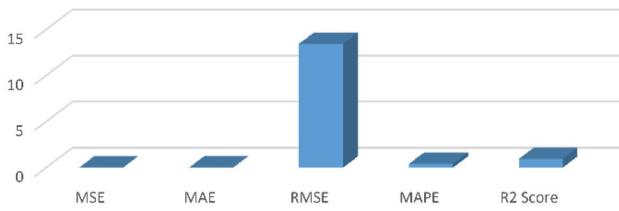
Optimizer	Learning rate	Beta 1	Beta 2	Decay	Momentum	Rho	Amsgrad	Nesterov	$R^2$ score
Adam	0.001	0.9	0.9	—	—	—	False	—	0.95
SGD	0.01	—	—	1e-5	0.9	—	—	True	0.92
RMSprop	0.01	—	—	—	—	0.9	—	—	0.90

**Table 9** Description of hyperparametric exploration

Epochs	Time (sec)	MSE	MAE	RMSE	MAPE	$R^2$ score
10	505.94	0.00016	0.00746	13.45	0.42	0.95
30	1362.34	0.00016	0.00748	11.95	0.39	0.94
100	4465.99	0.00016	0.00748	12.07	0.36	0.94
300	14,846.83	0.00016	0.00745	12.02	0.33	0.94

**Fig. 8** Prediction and actual data**Table 10** Summarized description of MSE, MAE, RMSE, MAPE, and  $R^2$  score

Description	Value
MSE	0.00016
MAE	0.00746
RMSE	13.45
MAPE	0.42
$R^2$ Score	0.95

**Fig. 9** Model performance according to the evolution metrics

### 5.3.5 Feature-Wise Performance

We have obtained the following six key characteristics: PM2.5, PM10, SO2, NO2, CO, and O3. And then provide the best guess for each of the features. The actual and predicted values for the following characteristics are displayed in Fig. 10 (a) PM2.5, (b) PM10, (c) SO2, (d) NO2, (e) CO, and (f) O3.

## 6 Result Analysis

### 6.1 Discussion on Feature-Wise Performance

Table 8 shows that our model has a higher  $R^2$  score of 0.93 for the feature PM2.5. There is no difference in the  $R^2$  values of the NO2 and CO features; both are equal to 0.91. This is accurate concerning both of these gases. On the other hand, a value of 0.88 was found for the  $R^2$  coefficient of PM10 and SO2 together. The O3 gain  $R^2$  value of a feature comes in at 0.85, making it the feature's concluding component. The SO2 achieved the best MAE value of 0.00538. And the NO2 obtained the best MAPE value of 0.17. The performance of our model was greater in terms of the overall forecasting prediction as opposed to the prediction of individual variables when compared to Tables 8 and 11.

### 6.2 Applied Model's Performance Comparison

We compared the efficacy of three distinct models: MLP, LSTM, and MLP-LSTM hybrid. The MLP model, which consisted of dense layers, obtained MSE values of 0.00157, MAE values of 0.02782, RMSE values of 42.3931, MAPE values of 112,912,818,176.00000, and an  $R^2$  value of 0.41. Using long short-term memory units, the LSTM model obtained an MSE of 0.00018, an MAE of 0.00875, an RMSE

**Fig. 10** The actual and prediction values of **a** PM2.5, **b** PM10, **c** SO2, **d** NO2, **e** CO, and **f** O<sub>3</sub>

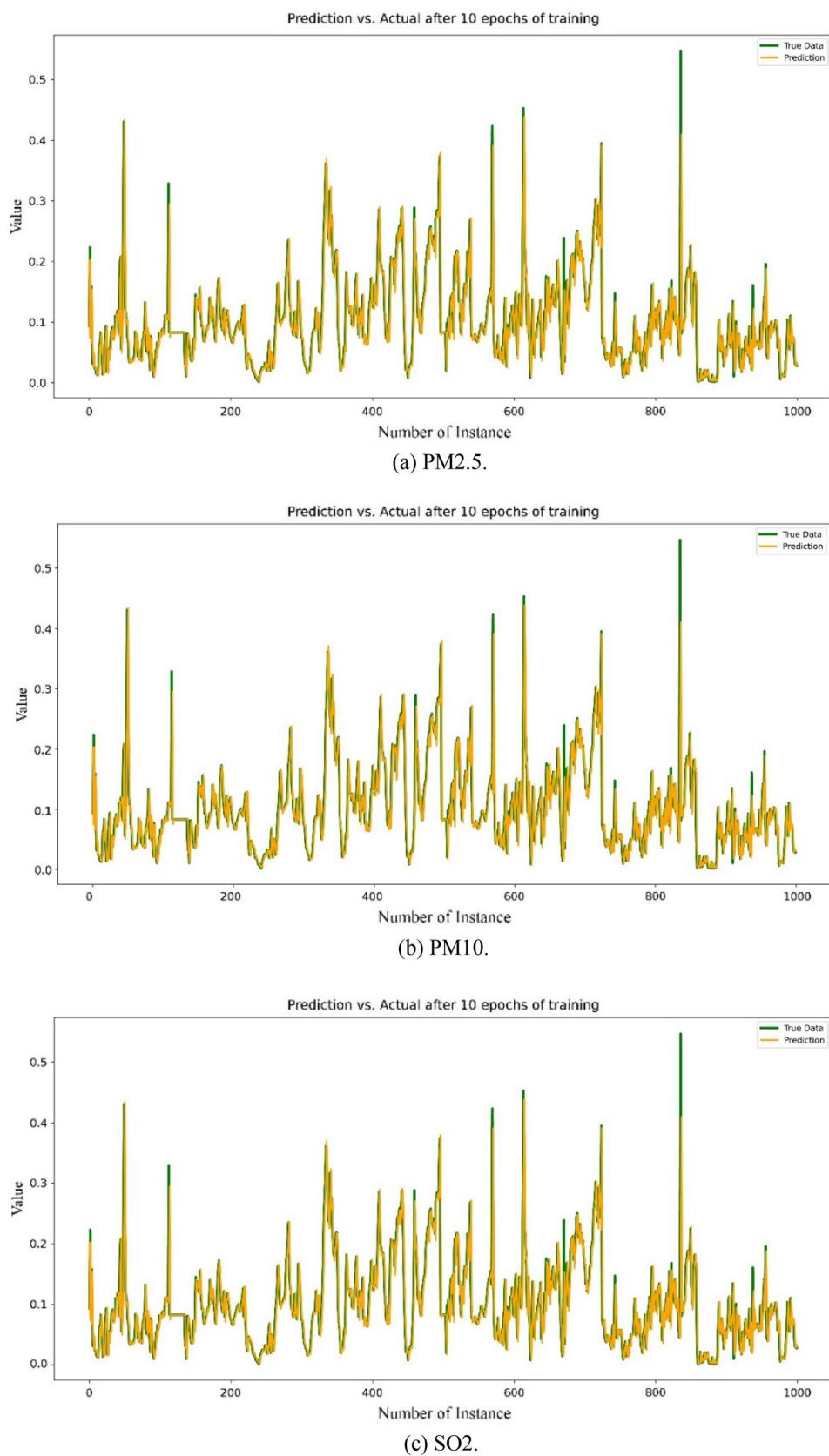
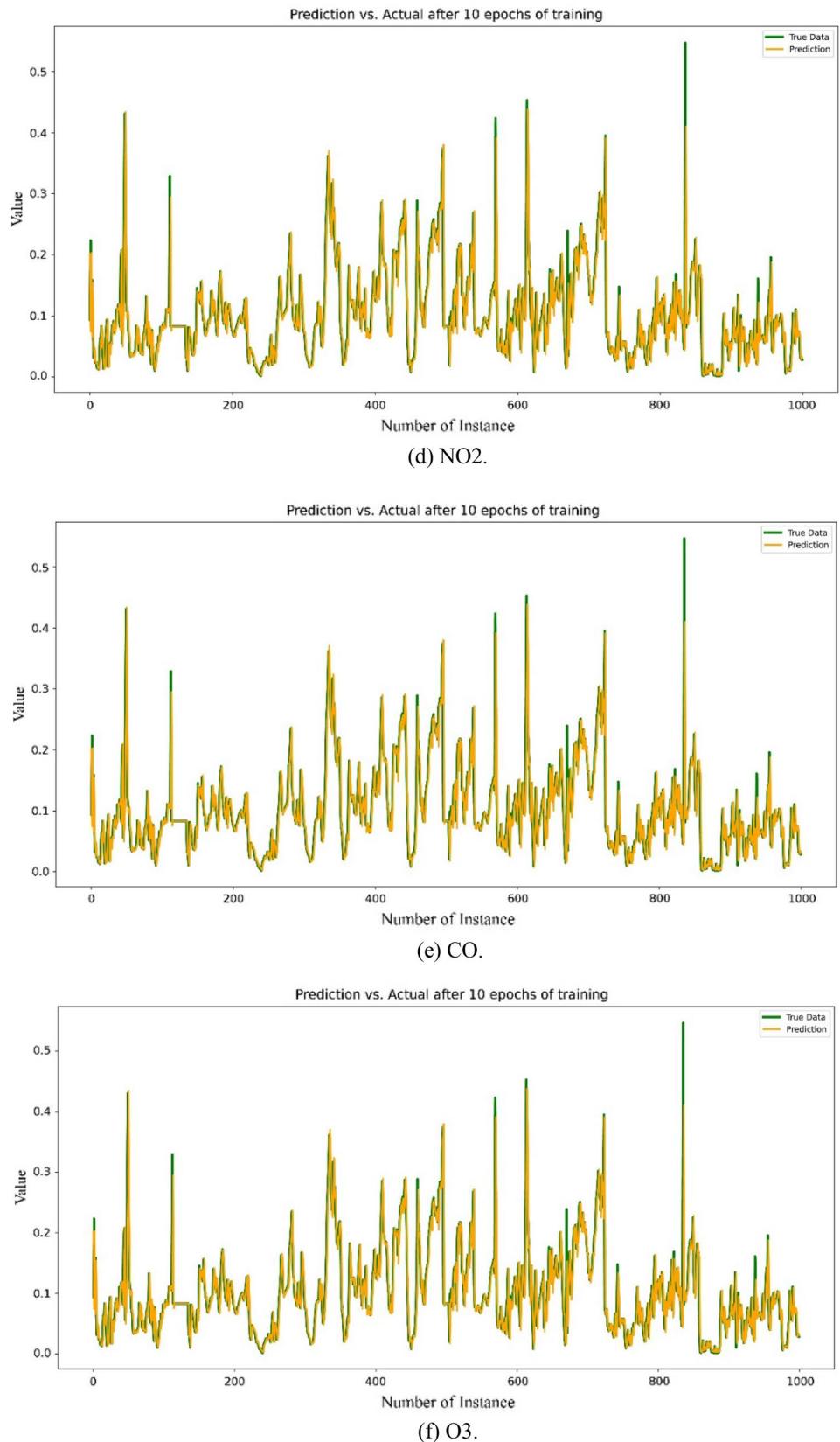


Fig. 10 (continued)



**Table 11** Summarized description of MSE, MAE, RMSE, MAPE, and  $R^2$  score of the 6 important features

Features	MSE	MAE	RMSE	MAPE	$R^2$
PM2.5	0.00049	0.01178	0.0005	0.51	0.93
PM10	0.00116	0.01821	0.0012	0.33	0.88
SO2	0.00021	0.00538	0.0002	0.24	0.88
NO2	0.00186	0.02740	0.0019	0.17	0.91
CO	0.00146	0.01741	0.0015	0.18	0.91
O3	0.00406	0.03213	0.0041	0.48	0.85

of 14.5032, a MAPE of 46,053,752,832.00000, and an  $R^2$  value of 0. However, the Hybrid MLP-LSTM model exhibited the greatest performance. This model incorporated the advantages of MLP and LSTM architectures. It was composed of dense layers and an LSTM layer. The Hybrid MLP-LSTM model outperformed the other two models across all performance metrics, with an MSE of 0.00016, MAE of 0.00746, RMSE of 13.45, MAPE of 0.42, and an impressive  $R^2$  score of 0.95. The capacity of the Hybrid MLP-LSTM model to capture both local and temporal dependencies in the data contributes to its superior performance. The MLP layers contribute by learning nonlinear relationships and feature representations, while the LSTM layer models the sequential nature of the data effectively. This combination improves the accuracy and reliability of predictions. The Hybrid MLP-LSTM model provides the greatest performance among the three evaluated models. It proves to be a potent and efficient method for predicting the target variable in the provided dataset, with enhanced accuracy and the ability to capture significant temporal patterns.

For multiple factors, the Hybrid MLP-LSTM model outperformed the other models in terms of predictive performance. First, the MLP component of the model is well suited for capturing non-linear relationships and learning complex feature representations due to its dense layers. It can effectively extract pertinent features from input data, enabling the model to capture intricate patterns and correlations within the dataset. The model's capacity to capture temporal patterns and long-term dependencies is enhanced by the LSTM component's ability to model sequential dependencies. LSTM units are equipped with a memory mechanism that enables them to retain

and effectively utilize past information. This is especially crucial for time series data, in which the order and sequence of observations play a crucial role. The Hybrid MLP-LSTM model leverages the MLP's capability for non-linear feature extraction and the LSTM's ability to capture temporal dependencies by combining the assets of the MLP and LSTM architectures. This combination of advantages enables the model to effectively learn from both the data's local patterns and temporal dynamics. The superior performance of the Hybrid MLP-LSTM model is a result of its enhanced capacity to capture intricate relationships, both within individual observations (local patterns) and across multiple observations over time (temporal patterns). This adaptability enables the model to make more accurate predictions by taking into account both the immediate and historical contexts of the data.

Overall, the ability of the Hybrid MLP-LSTM model to combine non-linear feature extraction with sequential modeling provides a comprehensive and effective method for capturing the intricate patterns and dependencies present in the dataset, resulting in its superior performance compared to the other models. Table 12 shows the summarized description of the applied model's performance.

## 7 Performance Comparison with Existing Works

The MLP-LSTM model stands out in comparison to other models due to its unique combination of multi-layer perceptron (MLP) and long short-term memory (LSTM) neural network architectures. Unlike traditional models, the MLP-LSTM model excels in capturing both nonlinear and temporal dependencies in the data, allowing it to effectively model complex spatiotemporal relationships. This makes it well-suited for forecasting tasks related to air quality.

Compared to the hybrid methods used by Asghari et al. [18] and Zhao et al. [19], the MLP-LSTM model offers a simpler and more streamlined approach by integrating MLP and LSTM architectures within a single model. This avoids the need for separate hybridization techniques such as Back Propagation with Genetic Algorithm (BP-GA) or combining multiple models like CNN and LSTM. Furthermore, the MLP-LSTM model demonstrates superior performance in

**Table 12** Summarized description of applied model's performance

Model	MSE	MAE	RMSE	MAPE	$R^2$
MLP	0.00157	0.02782	42.3931	112,912,818,176.00000	0.41
LSTM	0.00018	0.00875	14.5032	46,053,752,832.00000	0.92
Hybrid MLP-LSTM (proposed)	0.00016	0.00746	13.45	0.42	0.95

**Table 13** Summarized performance comparison with existing works

Description	Method	Dataset	Performance of $R^2$ score
Asghari et al., (2016) [18]	BP-GA	Aghdasiyeh and Mehrabad weather stations	0.55
Zhao et al., (2019) [19]	STCNN-LSTM	Air quality monitor stations in Beijing and Tianjin	0.70
Pang et al., (2019) [24]	LSTM	Tehran Air Quality Control Company	0.80
Zhang et al., (2022) [25]	LightGBM	GRAPES-3 km forecasting data	0.90
Kardhana et al., (2022) [26]	LSTM-RNN	LAPAN's Sadewa data and Jakarta DWR data	0.82
Won et al., (2022) [27]	Bidirectional LSTM	Gauging stations d pump stations urban stream data	0.90
Ouma et al., (2022) [28]	LSTM	Situ rainfall data and satellite-based meteorological data	0.89
	WNN		0.88
Tucrxun et al. (2022) [29]	EMD-MBES-LSTM	SCADA data	0.93
Shen et al., (2022) [30]	DBN	Daily streamflow at the hydrological Xiantao station data	0.89
	SOM-RF-DBN		0.93
Mani et al., (2022) [31]	ARIMA	CPCB data of Chennai, India	0.90
Liu et al., (2022) [32]	ST-CCN-IAQI	Shanghai monitoring air station data	0.90
Rahimpour et al., (2021) [33]	HTPD	Iranian air pollution monitoring system data	0.74
Liu et al., (2022) [34]	Random forest	Daily Emission Data of Zhangdian District, China	0.90
Fan et al., (2022) [35]	ML2	WRF Dataset	0.79
Naz et al., (2023) [36]	DL model	Monitoring station in Belfast, Northern Ireland	0.86
Proposed	Hybrid MLP-LSTM model	Beijing Multi-Site Air-Quality Data Set	0.95

comparison to other machine learning techniques, as seen in the study conducted by Pang et al. [20]. It effectively captures the temporal relationships in the data, resulting in lower root mean square error (RMSE) and mean absolute error (MAE) values for PM2.5 concentration forecasting. When compared to models like linear regression, LSTM-FCN, and LightGBM, as evaluated by Zhang et al. [21], the MLP-LSTM model showcases improved accuracy, as evidenced by reduced RMSE, MAE, and improved  $R^2$  values. It achieves high forecasting accuracy with an advantage in interpretability and the ability to handle complex temporal patterns.

Traditional models often rely on simplistic linear or statistical methods that struggle to capture complex nonlinear patterns and interactions present in real-world data. They require manual feature engineering, which can be time-consuming and subjective, potentially missing important features. These models may also struggle to handle temporal dependencies and capture long-term patterns effectively, lacking the ability to retain and utilize historical information. Additionally, they face challenges in integrating diverse data types, handling uncertainties, and scaling to large datasets. Moreover, computational constraints can hinder their performance, especially when dealing with complex techniques or real-time forecasting. These limitations collectively contribute to their inability to achieve the required level of performance in forecasting tasks that demand accuracy and adaptability in dynamic and complex environments.

While each model has its strengths and limitations, the MLP-LSTM model's ability to effectively capture nonlinear and temporal dependencies, its simplicity compared to hybrid methods, and its superior performance compared to other machine learning techniques make it a promising choice for accurate and reliable forecasting in environmental and related domains. Table 13 shows the summarized performance comparison of existing work.

## 8 Conclusion

In this study, a hybrid MLP-LSTM model was constructed to create forecasting predictions. This model was used to predict the forecasting. The research included a study of the Beijing Multi-Site Air-Quality Data Set, which was carried out as a part of the investigation. Although the dataset contains multi-site information the target value defines whether the air quality is safe, moderate, or unhealthy. Therefore, the model can target multi-site according to the air quality. According to what the model found to be appropriate for this specific set of circumstances, the MSE value was 0.00016, the MAE value was 0.00746, the RMSE value was 13.45, the MAPE value was 0.42, and the  $R^2$  value was 0.95. This demonstrates that the model is functioning as expected, which can be seen from the above. On the other hand, the

traditional modeling approaches for forecasting fall short of achieving the requisite level of performance. In this context, future research will focus on the selection of features and the creation of acceptable strategies for altering hyper-parameters to achieve the desired results. Moreover, automatically reshaped, added, and removed various layers to bring about an evolution in the neural network architecture. In addition, the MLP-LSTM model will be implemented on some more air and water quality datasets. Therefore, this model could be compatible with any kind of forecasting prediction.

**Acknowledgements** We would like to thank all of the authors for what they did for this research, and on behalf of all of the authors, we would like to thank Farhana Yasmin and Md. Mehedi Hassan for this study and for making a big contribution to it.

**Author contributions** FY contributed substantial conceptual and design contributions to the study. MdMH supported MdMH in assessing the data and appropriately drafting this report, as well as in supervising the full project and research. Other writers contributed to the development of the manuscript's preliminary draft. All authors reviewed the results and approved the final version of the paper.

**Funding** This research received no external funding.

**Data Availability** The dataset can be accessed from this link: <https://archive.ics.uci.edu/ml/datasets/Beijing+Multi-Site+Air-Quality+Data>

## Declarations

**Conflict of Interest** The authors declare that they have no competing interests.

**Ethical Approval and Consent to Participate** Not Applicable.

**Consent for Publication** Not Applicable.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Fradkov AL. Early history of machine learning. IFAC-PapersOn-Line. 2020;53(2):1385–90. <https://doi.org/10.1016/j.ifacol.2020.12.1888>.
2. McKendrick J. Artificial intelligence enters its golden age. Forbes. <https://www.forbes.com/sites/joemckendrick/2019/10/23/artificial-intelligence-enters-its-golden-age/> accessed 15 Aug 2022.
3. Wang Y, Huang L, Huang C, Hu J, Wang M. High-resolution modeling for criteria air pollutants and the associated air quality index in a metropolitan city. Environ Int. 2023;172:107752. <https://doi.org/10.1016/j.envint.2023.107752>.
4. Mehedi-Hassan M, Mollick S, Yasmin F. An unsupervised cluster-based feature grouping model for early diabetes detection. Healthc Anal. 2022;2:100112. <https://doi.org/10.1016/j.health.2022.100112>.
5. Rosser FJ, Rothenberger SD, Han Y-Y, Forno E, Celedón JC. Air quality index and childhood asthma: a pilot randomized clinical trial intervention. Am J Prev Med. 2023;64(6):893–7. <https://doi.org/10.1016/j.amepre.2022.12.010>.
6. Hassan MM, et al. A comparative assessment of machine learning algorithms with the least absolute shrinkage and selection operator for breast cancer detection and prediction. Decis Anal J. 2023;7:100245. <https://doi.org/10.1016/j.dajour.2023.100245>.
7. Benidis K, et al. Deep learning for time series forecasting: tutorial and literature survey. ACM Comput Surv. 2022. <https://doi.org/10.1145/3533382>.
8. Deng X, et al. Disease specific air quality health index (AQHI) for spatiotemporal health risk assessment of multi-air pollutants. Environ Res. 2023;231:115943. <https://doi.org/10.1016/j.enres.2023.115943>.
9. Hassan MM, et al. Efficient prediction of water quality index (WQI) using machine learning algorithms. Hum-Centric Intell Syst. 2021;1:1–12. <https://doi.org/10.2991/hcis.k.211203.001>.
10. Makridakis S, Spiliotis E, Assimakopoulos V. The M4 competition: 100,000 time series and 61 forecasting methods. Int J Forecast. 2020;36(1):54–74. <https://doi.org/10.1016/j.ijforecast.2019.04.014>.
11. Heng SY, et al. Artificial neural network model with different backpropagation algorithms and meteorological data for solar radiation prediction. Sci Rep. 2022;12(1):1. <https://doi.org/10.1038/s41598-022-13532-3>.
12. Heydari A, Majidi-Nezhad M, Astiaso-Garcia D, Keynia F, De Santoli L. Air pollution forecasting application based on deep learning model and optimization algorithm. Clean Techn Environ Policy. 2022;24(2):607–21. <https://doi.org/10.1007/s10098-021-02080-5>.
13. Zhao Z, Wu J, Cai F, Zhang S, Wang Y-G. A hybrid deep learning framework for air quality prediction with spatial autocorrelation during the COVID-19 pandemic. Sci Rep. 2023;13(1):1. <https://doi.org/10.1038/s41598-023-28287-8>.
14. Rodríguez-García MI, Carrasco-García MG, González-Enrique J, Ruiz-Aguilar JJ, Turias IJ. Long short-term memory approach for short-term air quality forecasting in the Bay of Algeciras (Spain). Sustainability. 2023;15(6):6. <https://doi.org/10.3390/su15065089>.
15. Azhari F, Sennersten CC, Lindley CA, Sellers E. Deep learning implementations in mining applications: a compact critical review. Artif Intell Rev. 2023. <https://doi.org/10.1007/s10462-023-10500-9>.
16. Wang W, Tang Q. Combined model of air quality index forecasting based on the combination of complementary empirical mode decomposition and sequence reconstruction. Environ Pollut. 2023;316:120628. <https://doi.org/10.1016/j.envpol.2022.120628>.
17. Aarthi C, Ramya VJ, Falkowski-Gilski P, Divakarachari PB. Balanced spider monkey optimization with Bi-LSTM for sustainable air quality prediction. Sustainability. 2023;15(2):2. <https://doi.org/10.3390/su15021637>.
18. Fang W, Zhu R, Lin JC-W. An air quality prediction model based on improved Vanilla LSTM with multichannel input and multiroute output. Expert Syst Appl. 2023;211:118422. <https://doi.org/10.1016/j.eswa.2022.118422>.
19. Iskandaryan D, Ramos F, Trilles S. Graph neural network for air quality prediction: a case study in Madrid. IEEE Access. 2023;11:2729–42. <https://doi.org/10.1109/ACCESS.2023.3234214>.

20. ArunaKumari P, Vijayalata Y, Susmitha Valli G, Lakshmi Prasanna Y. Air contamination prediction and comparison using machine learning algorithms. In: Kumar A, Ghinea G, Merugu S, Hashimoto T (Eds) Proceedings of the International Conference on Cognitive and Intelligent Computing, in Cognitive Science and Technology. Singapore: Springer Nature, 2023; 661–677. [https://doi.org/10.1007/978-981-19-2358-6\\_60](https://doi.org/10.1007/978-981-19-2358-6_60).
21. Duan J, Ren Q. Air quality prediction based on wavelet analysis and machine learning. *Strateg Plan Energy Environ.* 2023. <https://doi.org/10.13052/spee1048-5236.4217>.
22. Asghari M, Nematzadeh H. Predicting air pollution in Tehran: genetic algorithm and back propagation neural network. *J AI Data Min.* 2016;4(1):49–54. <https://doi.org/10.5829/idosi.JAIDM.2016.04.01.06>.
23. Zhao G, Huang G, He H, He H, Ren J. Regional spatiotemporal collaborative prediction model for air quality. *IEEE Access.* 2019;7:134903–19. <https://doi.org/10.1109/ACCESS.2019.2941732>.
24. Pang A, Chang MWL, Chen Y. Evaluation of random forests (RF) for regional and local-scale wheat yield prediction in Southeast Australia. *Sensors.* 2022;22(3):3. <https://doi.org/10.3390/s22030717>.
25. Zhang H, Wang Y, Chen D, Feng D, You X, Wu W. Temperature forecasting correction based on operational GRAPES-3km model using machine learning methods. *Atmosphere.* 2022;13(2):2. <https://doi.org/10.3390/atmos13020362>.
26. Kardhana H, Valerian JR, Rohmat FIW, Kusuma MSB. Improving Jakarta's Katulampa barrage extreme water level prediction using satellite-based long short-term memory (LSTM) neural networks. *Water.* 2022;14(9):9. <https://doi.org/10.3390/w14091469>.
27. Won Y-M, Lee J-H, Moon H-T, Moon Y-I. Development and application of an urban flood forecasting and warning process to reduce urban flood damage: a case study of Dorim River Basin, Seoul. *Water.* 2022;14(2):2. <https://doi.org/10.3390/w14020187>.
28. Ouma YO, Cheruyot R, Wachera AN. Rainfall and runoff time-series trend analysis using LSTM recurrent neural network and wavelet neural network with satellite-based meteorological data: case study of Nzoia hydrologic basin. *Complex Intell Syst.* 2022;8(1):213–36. <https://doi.org/10.1007/s40747-021-00365-2>.
29. Tuexun W, Xu C, Guo H, Guo L, Zeng N, Gao Y. A wind power forecasting model using LSTM optimized by the modified bald eagle search algorithm. *Energies.* 2022;15(6):6. <https://doi.org/10.3390/en15062031>.
30. Shen J, Zou L, Dong Y, Xiao S, Zhao Y, Liu C. Improving daily streamflow forecasting using deep belief net-work based on flow regime recognition. *Water.* 2022;14(14):14. <https://doi.org/10.3390/w14142241>.
31. Mani G, Viswanadhapalli JK, Stonie AA. Prediction and forecasting of air quality index in Chennai using regression and ARIMA time series models. *J Eng Res.* 2022;10(2A):2A. <https://doi.org/10.36909/jer.10253>.
32. Liu X, et al. Fine-grained individual air quality index (IAQI) prediction based on spatial-temporal causal convolution network: a case study of Shanghai. *Atmosphere.* 2022;13(6):6. <https://doi.org/10.3390/atmos13060959>.
33. Rahimpour A, Amanollahi J, Tzanis CG. Air quality data series estimation based on machine learning approaches for urban environments. *Air Qual Atmos Health.* 2021;14(2):191–201. <https://doi.org/10.1007/s11869-020-00925-4>.
34. Liu Y, Wang P, Li Y, Wen L, Deng X. Air quality prediction models based on meteorological factors and real-time data of industrial waste gas. *Sci Rep.* 2022;12(1):1. <https://doi.org/10.1038/s41598-022-13579-2>.
35. Fan K, Dhammapala R, Harrington K, Lamb BK, Lee Y. Development of a machine learning approach for local-scale ozone and PM<sub>2.5</sub> forecasting: application to multiple AQs sites in the Pacific Northwest,” May 2022. <https://eartharxiv.org/repository/view/3338/>. Accessed: 18 Aug 2022.
36. Naz F, et al. Comparative analysis of deep learning and statistical models for air pollutants prediction in urban areas. *IEEE Access.* 2023;11:64016–25. <https://doi.org/10.1109/ACCESS.2023.3289153>.
37. UCI machine learning repository: Beijing Multi-site air-quality data data set. <https://archive.ics.uci.edu/ml/datasets/Beijing+Multi-Site+Air-Quality+Data> accessed 16 Aug 2022.
38. Yin M, Iannelli A, Smith RS. Data-driven prediction with stochastic data: confidence regions and minimum mean-squared error estimates. In: 2022 European Control Conference (ECC), 2022, pp. 853–858. <https://doi.org/10.23919/ECC55457.2022.9838046>.
39. Zhou M, Zhu Z, Hu F, Bian K, Lai W, Hu T. Short-term commercial load forecasting based on peak-valley features with the TSA-ELM model. *Energy Sci Eng.* 2022;10(8):2622–36. <https://doi.org/10.1002/ese3.1203>.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.