scientific reports

Check for updates

OPEN

Implementing heuristic-based multiscale depth-wise separable adaptive temporal convolutional network for ambient air quality prediction using real time data

Raj Anand Sundaramoorthy[1], Antony Dennis Ananth[1], Koteeswaran Seerangan[2], Malarvizhi Nandagopal[3], Balamurugan Balusamy[4] & Shitharth Selvarajan[5,6]✉

In many emerging nations, rapid industrialization and urbanization have led to heightened levels of air pollution. This sudden rise in air pollution, which affects global sustainability and human health, has become a significant concern for citizens and governments. While most current methods for predicting air quality rely on shallow models and often yield unsatisfactory results, our study explores a deep architectural model for forecasting air quality. We employ a sophisticated deep learning structure to develop an advanced system for ambient air quality prediction. We utilize three publicly available databases and real-world data to obtain accurate air quality measurements. These four datasets undergo a data cleaning to yield a consolidated, cleaned dataset. Subsequently, the Fused Eurasian Oystercatcher-Pathfinder Algorithm (FEO-PFA)—a dual optimization method combining the Eurasian Oystercatcher Optimizer (EOO) and Pathfinder Algorithm (PFA)—is applied. This method aids in selecting weighted features, optimizing weights, and choosing the most relevant attributes for optimal results. These optimal features are then incorporated into the Multiscale Depth-wise Separable Adaptive Temporal Convolutional Network (MDS-ATCN) for the ambient Air Quality Prediction (AQP) process. The variables within MDS-ATCN are further refined using the proposed FEO-PFA to enhance predictive accuracy. An empirical analysis is performed to compare the efficacy of our proposed model with traditional methods, underscoring the superior effectiveness of our approach. The average cost function is reduced to 5.5%, the MAE to 28%, and the RMSE to 14% by the suggested method, according to the performance research conducted with regard to all datasets.

Keywords Contamination of air, Air Quality Index, Ambient air quality prediction, Fused Eurasian Oystercatcher-Pathfinder Algorithm, Multiscale depth-wise separable adaptive temporal convolutional network

Environmental issues have been brought about by the ongoing expansion of globalized industrialization and urbanization. The deterioration of air quality brought on by the growth of industrialization and urbanization constitutes one of the major environmental issues[1]. Power stations, manufacturing industries, and automotive exhaust emissions caused by transportation-related requirements have eventually contributed to the worsening of the worldwide quality of air[2]. Air pollution poses a severe threat to people's health and well-being because it can result in cancer and other respiratory disorders[3]. Recurrent air pollution occurrences not only have an adverse effect on people's health but also result in significant financial losses and other societal issues[4]. According

[1]School of Computing, SASTRA (Deemed to be University), Tirumalaisamudram, Thanjavur, Tamil Nadu 613401, India. [2]Department of CSE (AI&ML), S.A. Engineering College (Autonomous), Chennai, Tamil Nadu 600077, India. [3]Department of CSE, School of Computing, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, Tamil Nadu 600062, India. [4]Shiv Nadar (Institution of Eminence Deemed to Be University), Greater Noida, Uttar Pradesh 201314, India. [5]Department of Computer Science, Kebri Dehar University, 250 Kebri Dehar, Ethiopia. [6]School of Built Environment, Engineering and Computing, Leeds Beckett University, LS6 3QS Leeds, UK. ✉email: ShitharthS@kdu.edu.et

to the features of air pollutants, rapid scientific study, precise forecasting of the quality of the air, and efficient prevention and purification can aid in the early implementation of preventative measures. People's everyday lives are significantly impacted by the quality of air surrounding them[5]. Reliable air quality forecasting has become increasingly vital as a tool for reducing pollutants and enhancing the quality of the air[6]. Effective performance appraisal systems and forecasting of fluctuations in air quality are beneficial in the prevention and control of air pollution, which in turn enable the public's health and the environment to be kept safe[7]. This is made possible by a deeper understanding of the factors involved and the evolutions of air pollutants. Regulations for preventing and controlling air pollution can also be developed in accordance with particular circumstances[8].

Information on air quality has caused great worry throughout the world. In order to predict air quality, various statistical techniques, like Auto-Regressive Integrated Moving Average (ARIMA), are extensively utilized. If the pattern is irregular or non-linear, these linear models might not produce an accurate result[9]. Support Vector Regression (SVR)[10] has been used in nonlinear regression forecasting in past years. Nevertheless, due to the large and complicated amount of data used by air quality predictors, which could lead to overfitting[11], SVR with underlying kernel translation, such as RBF kernel, could fail to provide an AQP model with satisfactory performance. Unfortunately, the complicated drift in air quality, such as PM2.5 concentrations, cannot be accurately captured by the AQP systems now in use[12]. Deep learning-based classifiers can extract features from the information regarding air quality and can increase forecast reliability[13]. Several techniques simultaneously mimic the spatial and time interdependence of air quality data. Yet, commonly used machine learning techniques frequently exhibit considerable performance variability under various conditions[14].

Neural networks are better suited for predicting air quality than the methods mentioned above because they have high non-linearity, huge parallelization, and strong learning capacity[15]. Many variables, including climate, wind speed, and geographical arrangement, have an impact on the quality of air[16]. Because of this consequence, it is challenging to produce definite and precise prediction results using the popular single-model prediction system. One kind of approach is to combine many predictive models for air quality. In comparison to current methods, the integrated approach can greatly increase prediction accuracy[17]. However, there is still much to learn about how to combine the benefits of several models depending on the features of the data collection. The data-driven technique for modeling air quality has been taken into consideration for designing various AQP systems because of the utilization of artificial intelligence and big data. Artificial Neural Networks (ANN) has emerged as the trending technique for air pollution modeling among data-driven techniques. The neural-network approaches have been shown to be useful in a variety of fields. Complicated meteorological phenomena can be effectively modeled using the deep-learning strategy. Nevertheless, the majority of the present deep learning-based approaches fell short of modeling both the PM2.5 time series and the meteorological data at the same time in a unified predictive framework.

The following is a description of the paper's main accomplishments in the forecasting model.

- To create an enhanced AQP model employing an adaptive-based deep learning framework and a meta-heuristic algorithm to mitigate the harmful impacts brought on by ambient polluted air.
- To put forth the FEO-PFA, by which the parameters like kernel size, filter size, epochs, and random state in the TCN as well as features and weights, are optimized in order to obtain the best-weighted feature selection and better prediction rate.
- To accurately forecast the quality of the air employing the MDS-ATCN framework, where TCN is incorporated to provide a better prediction result.
- The National Air Quality Index, Open Government Data (OGD) Platform India, and Air Quality Index (AQI) across stations and cities in India from 2015 to 2020 are some of the open source datasets used in this work for system validation and assessment. Given that they are the largest, most current, and emerging datasets, they are also the most well-liked. As a result, the proposed work uses these datasets to assess how well the methodology could be used to forecast the air quality based on various attribute information types.
- Using various performance metrics, the proposed MDS-ATCN's performance results have been verified and compared against the results of other existing AQ prediction techniques.

By applying the MSD-ATCN approach, the proposed system can effectively predict various contaminants. With appropriate data handling processes, it estimates the air quality based on the provided datasets. Compared to conventional statistical models, the framework can estimate air pollution levels more accurately by utilizing the proposed methods. Because of the increased precision, early warning systems and more informed decision-making are made possible, empowering stakeholders to take preventative action to lessen the negative effects of air pollution. Since air pollution levels can be predicted throughout time and temporal dependencies could be effectively modelled using the MDS-ATCN. Through the utilization of these functionalities, the framework can pinpoint areas of high pollution, monitor patterns in pollution, and make more accurate projections about future pollution levels. Furthermore, it is highly proficient at automatically collecting pertinent features and acquiring informative representations from the provided data. This is especially helpful for predicting air quality because the relationship between the input variables (like meteorological data and pollutant concentrations) and the output (like the AQI) can be rather complex and nonlinear. It also has the ability to learn hierarchical representations of air quality data, which allows it to find hidden patterns that conventional feature engineering techniques would miss.

The further topics that are discussed in this paper are as follows. In section "Materials and methods", the existing works related to AQP are surveyed. In section "Development of multiscale depthwise separable adaptive deep learning network for ambient air quality prediction using real time data", the development of an MDS deep learning network for ambient AQP using real-time data is discussed. In section "Modified heuristic-based

weighted feature selection model for real time data-aided ambient air quality prediction", a modified heuristic-based weighted feature selection model for real-time data-aided ambient AQP is discussed. In section "Multi-scale depthwise separable adaptive temporal convolutional network for real time data-aided ambient air quality prediction", MDS-ATCN for real-time data-aided ambient AQP is provided and discussed in detail. In section "Result and discussion", the simulation carried out, and the comparison done for evaluation of the developed AQP model are provided. In section "Conclusion", the summary of the developed AQP model is given.

Materials and methods

Attention, Autoencoder, GRNN, LSTM, and seq2seq model show better performance in predicting the air quality. This method provides accurate results, and also this model is quick to train. This model also provides warnings prior to sudden pollution strikes. Yet, this method provides accurate prediction results only in temporal attention, and hence the efficiency in spatial attention is still under consideration. There is lagging in the functioning of this model because of its loss function. LSTM, GRU, and SHAP are efficient in identifying only the essential features for predicting air quality and the effects of these features on the quality of the air. Yet, the computational complexity is more, the memory requirement is more, and the overall efficiency of the computation is not as expected in this method. Attention LSTM, Seq2Seq, and XGBoosting tree has higher accuracy in predicting the quality of air. But, this model faces some issues while dealing with outlier values. PSO and BP Unlike the conventional methods, this method attains good searching ability and does not allow the system to fall into a local minimum, but it has lower convergence speed. CNN, LSTM, and ST-DNN prediction performance is higher in the initial hours. However, this method does not consider the delay in a long time which results in reduced system performance. IoT, NARX[6] method is highly suitable for real-time prediction of the quality of air because of the utilization of various sensors by this method. Yet, this model is highly complex, requires more memory, requires more power, and has issues regarding the security of data. SVM memory requirement is low. However, this model is highly affected by noised data. Autoencoder and LSTM are efficient in forecasting the PM2.5 time series in a wider area. But, deterministic factors like gas emission and economy are not considered by this method while predicting the quality of air. This results in the development of an advanced deep learning-based air quality prediction model.

Related works

Liu et al.[18] have proposed the Attention-based Air Quality Predictor (AAQP), a Sequence-to-Sequence (Seq2Seq) model aimed at protecting people from air pollution. It used past air quality data as well as weather information to forecast the next AQI. To increase the training speed of Seq2Seq, the original Recurrent Neural Network (RNN) in the encoder was replaced with a fully connected encoder. Position embedding was also implemented to help the fully linked encoder determine the sequential links between the source sequences. The n-step recurrent prediction was used to solve the error accumulation problem. As compared to the original Seq2Seq attention model, the training time was much shortened, yet the error accumulation was still decreased. The n-step recurrent prediction in the AAQP outperformed other approaches, as the experimental results confirmed. Yang et al.[19] have combined data from air pollution, temperature, moisture content, and atmospheric pressure databases to create the Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) ambient air forecast models in four different scenarios. After this, the interpretability of the AQP was evaluated using the SHapley Additive exPlanation (SHAP) technique. Merely considering meteorological conditions did not improve prediction accuracy. When atmospheric parameters and numerous other air pollutants were also taken into consideration, the model's precision increased. In addition, the pressure gradient had the greatest influence on air quality forecasts, with temperature and humidity following closely behind. The different forecast accuracy rates could have their origins in the interplay between different air pollutants and climatic circumstances. The investigational results reported in this study may help create an AQP that is more trustworthy and accurate.

Chen et al.[20] have proposed a forecasting approach that makes use of a dual LSTM mixed model. Using the Seq2Seq method, a single forecasting model was first constructed. The expected quantity of every element in the air quality data was independently ascertained using this procedure. Every facet of air quality was handled as a temporal data series in the prediction process. The attention LSTM model carried out the multi-factor forecasting. The model considered the factors that influence air quality, including data from neighboring stations and meteorological conditions. In the end, the XGBoosting (eXtreme Gradient Boosting) tree was used to combine two models. The total predicted outcomes were attained by summing the estimated values of the best sub-tree nodes. When five assessment methodologies were implemented to evaluate and analyse the suggested strategy, it fared better in terms of variance and design expressive power. When compared to other models, the model's forecast data accuracy had much enhanced.

Huang et al.[21] have put forward an enhanced method for optimizing Back Propagation (BP) neural networks for AQI prediction, based on the Particle Swarm Optimization (PSO) algorithm. The improved PSO algorithm improved the learning component and the weighting factor variant method, enabling fast convergence to the best path and guaranteeing early global searching capabilities. An adaptive mutation approach was developed to keep the elements from settling into the local minimum during the search phase. A BP neural network was optimized using the improved PSO method, and the results were compared and examined to generate an AQI forecast that was more accurate. Soh et al.[22] have determined spatial relations to forecast air quality for up to 48 h using a hybrid LSTM, artificial neural network, and CNN. The proposed forecasting algorithm used a range of meteorological data from the previous few hours along with topographical region data to assess how geography impacts air quality. The framework comprised trends from many locations that were inferred via time-domain correlations between neighboring locations and locations with similar attributes. Tests conducted on data sets

from Beijing and Taiwan showed that the proposed model performed amazingly well, outperforming the most recent state-of-the-art methods.

Sridhar et al.[23] have developed an Internet of Things (IoT) node equipped with sensors to measure temperature, humidity, and pollution. IoT nodes were placed throughout the research facility in order to gather data on air quality from interior sections for study and to validate the suggested concept. Using GSM/Wi-Fi equipment, the suggested system transmitted the observations of the air quality in real-time to a webpage and software applications. It also produced audible alerts if it detected deviations in the air quality. The technology was appropriate for Big Data analysis applications like traffic and weather forecasting since it captured sensor signals at a rate of one sample per second. The standardized air pollutant sensor values and sensor data had an optimal correlation. The goal of the endeavor was to advance the creation of inexpensive, decentralized sensor devices for environment intelligence applications. Chun et al.[24] have forecasted the air quality at an unclear location and time using the Support Vector Machine (SVM) approach. Features such as population, land use, pollution sources, topographical factors, and economy were retrieved from the Geographic Information System (GIS) and then combined into a time-series data based on geography. Temporal estimation was first performed in the base stations in order to spatially estimate the upcoming AQI of unknown locations. It was confirmed that short-term temporal forecasting has high precision. It has been discovered that a variety of climatic and meteorological elements influence seasonal variations. During the geographical assessment phase, the spatial variables that appeared to have an impact on the quality of the air were urban sprawl and city type. The AQI computation has been linked to characteristics such as transportation usage, forest utilization, agricultural utilization, economic factors, and terrain utilization. An SVM-based forecasting framework for northern Taipei was created by this study. Other locations could develop their own designs utilizing their local data to enhance decision-making.

Xu and Yoneda[25] have proposed forecasting PM2.5 time series across multiple metropolitan areas using the LSTM auto-encoder. The underlying relevance of the contaminants at different places was automatically and implicitly revealed by the framework. Furthermore, all of the meteorological data from the observation station was used, which improved the performance of the suggested framework. In instance, multilayer LSTM networks have been able to replicate the spatiotemporal characteristics of contaminant nanoparticles in urban areas. Furthermore, a stacked auto-encoder was used to encode the primary transformation pattern of urban meteorological networks, providing essential auxiliary data for PM2.5 time-series forecasting. Furthermore, multitask learning solved the problem of the traditional data-driven modelling technique's insufficient use of multisite data by automatically recognizing the dynamic connection between various key pollutant time series. The simulation results of Beijing's PM2.5 forecasts showed how effective the recommended technique was.

Problem statement

Many intelligence-based computational techniques are developed for air quality prediction and classification in the early literature publications. Recent reviews have revealed a number of obstacles and issues, which are shown below:

- Complex nonlinear correlations found in air quality data may be difficult for linear prediction algorithms to identify. In the event that the relationships are not linear, they may under fit the data, resulting in less than ideal performance.
- Even though decision trees and their ensemble variations are effective and easy to comprehend, they have a tendency to overfit the training set, particularly when the dataset contains a large number of features or intricate feature interactions. Inadequate adaptation to new data might result from overfitting.
- Given the class label, the Bayesian model assumes that characteristics are conditionally independent, which may not be the case for all datasets on air quality. A breach of this supposition may result in less than ideal performance.
- Even though deep learning models are capable of capturing intricate correlations in data, they frequently need a great amount of data to train, and they may overfit, particularly when working with small datasets. They also need careful hyperparameter adjustment and can be computationally expensive to train.
- Certain classifiers are challenging to interpret, especially the more sophisticated ones like neural networks and ensemble techniques. This makes it challenging to comprehend how the model makes its predictions. Interpretability is critical, particularly in fields where decision-making requires a comprehension of the elements impacting predictions, such as air quality prediction.
- The selection of features and the quality of the input data have a significant impact on classifier performance. A noisy, incomplete, or outlier-filled set of data might have a detrimental effect on classifier performance. Similar to this, feature engineering is essential for drawing out pertinent information from the data; yet, poor feature extraction or selection might produce less-than-ideal outcomes.

The purpose of the proposed work is to present a novel intelligent model of prediction for air quality, which shall overcome all the pitfalls involved in its predecessors. Most of the current models in the prediction of the quality of the air rely on shallow learning algorithms that lack the possibility of capturing complex patterns in the data and thereby lead to suboptimal performance. Our method represents a deep, sophisticated learning structure of multi scale depth-wise separable adaptive temporal convolutional networks in order to model air quality time–space data dynamics. We combine the Fused Eurasian Oystercatcher-Pathfinder Algorithm with dual optimization power derived from the Eurasian Oystercatcher Optimizer and Pathfinder Algorithm to enhance predictive accuracy. This is, therefore, an important optimization technique in the choice of weighted features, optimizing weights, and selecting relevant attributes for the consolidated and cleaned data set that is derived from the integration of four publicly available air quality databases. FEO-PFA integration will ensure

that the model comes out not only accurate but also efficient in handling big datasets having high dimensionality. To this end, by addressing these challenges, our proposed model significantly reduces the mean cost function, MAE, and RMSE and thus further exhibits superior performance outcomes compared to traditional methods on air quality prediction.

Development of multiscale depthwise separable adaptive deep learning network for ambient air quality prediction using real time data

Implemented ambient air quality prediction model

Modern human activities inevitably involve energy usage and its effects. The burning of coal, kerosene, and straw are only a few examples of anthropogenic sources of air pollution, along with emissions from cars, aerosol cans, and factories. Every day, a wide range of harmful pollutants, including NO_2, CO_2, CO, Particulate Matter (PM), O_3, SO_2, Pb, and NH_3, are discharged into our environment. The health of animals, people, and even plants is impacted by the chemicals and particles that make up air pollution. Humans are susceptible to a wide range of dangerous illnesses brought on by air pollution, including lung cancer, bronchitis, pneumonia, and heart disease. Aerosol production, global warming, smog, acid rain, impaired vision, early mortality, and other current environmental problems are all caused by poor air quality. A measurement metric known as the AQI has a direct connection to public health. A higher AQI value denotes a riskier degree of exposure for the general public. Consequently, the desire to accurately anticipate the AQI drove scientists to track and model air quality. With an increase in industrial and motorized activity, monitoring and forecasting AQI, particularly in metropolitan areas, has become an essential and difficult undertaking. A variety of models, including statistical, deterministic, physical, and machine learning models, has been used to predict AQI. The conventional methods based on statistics and probability are exceedingly intricate and ineffective. It has been demonstrated that machine learning-based AQI prediction models are more dependable and consistent. Data collecting was made simple and accurate by modern technologies and sensors. Only machine learning algorithms are capable of handling the rigorous analysis needed to make accurate and trustworthy predictions from such vast environmental data. So an enhanced AQP model is developed by utilizing the deep learning technique. The pictorial illustration of the developed model for AQP is shown in Fig. 1. The health of animals, people, and even plants is impacted by

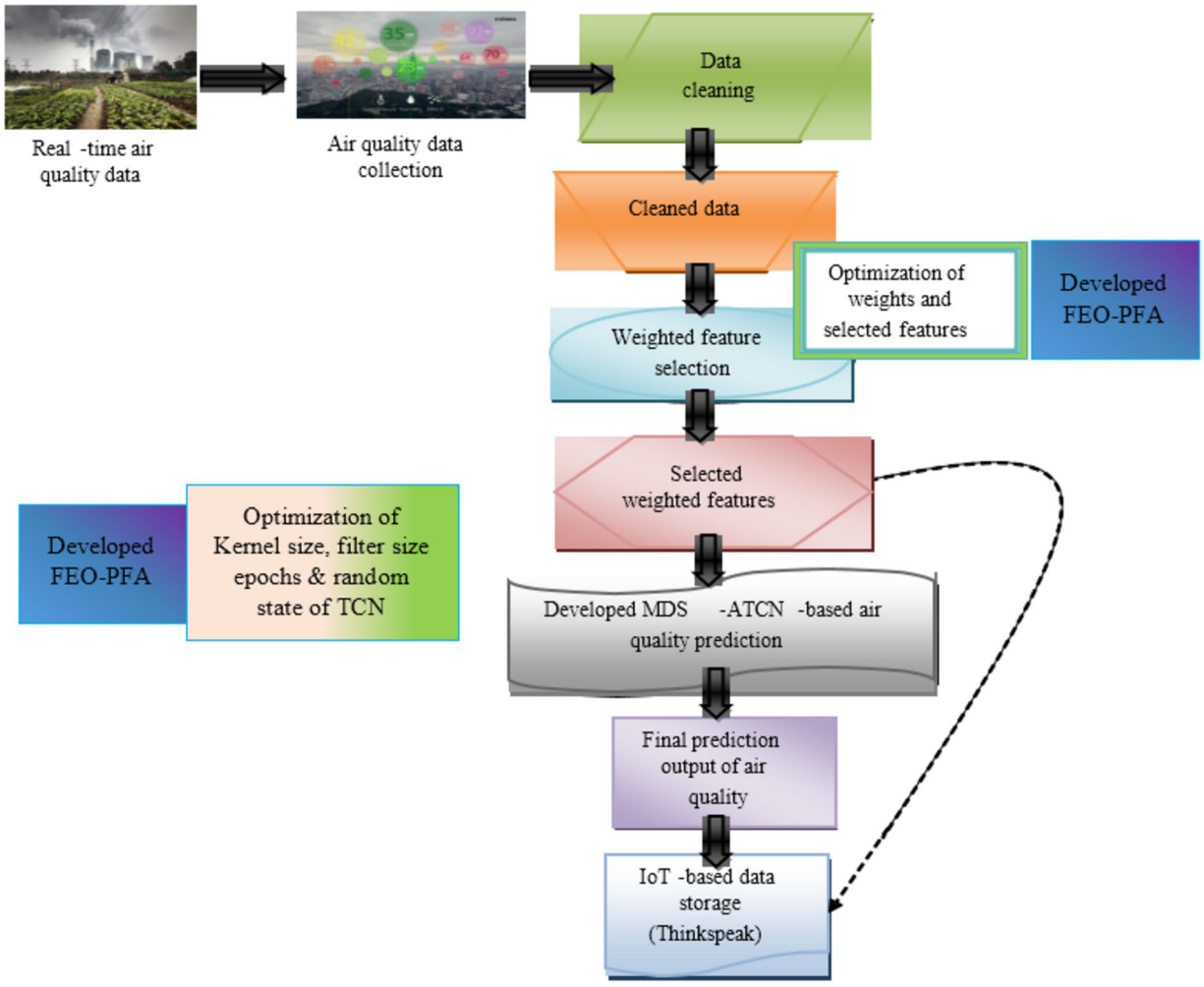

Figure 1. Architecture of the generated deep learning-based AQP framework.

the chemicals and particles that make up air pollution. Humans are susceptible to a wide range of dangerous illnesses brought on by air pollution, including lung cancer, bronchitis, pneumonia, and heart disease. Aerosol production, global warming, smog, acid rain, impaired vision, early mortality, and other current environmental problems are all caused by poor air quality. But the statistical and traditional method of AQP is inaccurate and ineffective. So, deep learning technique is adopted in forecasting the quality of the air. The pros and cons of the conventional air quality prediction model based on deep learning technique are depicted in Table 1.

The primary goal of the proposed work is to apply the following approaches to produce an accurate and efficient air quality prediction utilizing real-time and public datasets:

1. A standard data preprocessing with outlier removal and normalization operations is performed for avoiding error predictions.
2. A hybrid FEO-PFA methodology is applied to optimize the parameters like kernel size, filter size, epochs, and random state of classifier.
3. A hybrid and unique MDS-ATCN model is developed for the prediction of air quality with good performance outcomes.

The novel FEO-PFA algorithm for classifier parameter optimization is developed in the proposed study by combining two different and cutting-edge optimization techniques. By using this method, the suggested model's air quality prediction performance is significantly improved while system complexity is reduced. Additionally, a novel classification strategy for air quality prediction is developed by integrating the Temporal Convolutional Network (TCN) with the Multi-scale Depthwise Separable Adaptive Network (MDS). By using these two distinct and innovative hybrid approaches significantly enhances the suggested work's prediction performance.

The real-time dataset of ambient air is gathered in the first stage from numerous databases that are readily available. To remove redundant and irrelevant data from the real-time data acquired, the collected dataset is cleaned. The features are then selected optimally from the cleaned data by the recommended FEO-PSA. Finally, using the suggested FEO-PFA, the crucial features associated with the environmental data are chosen. The weights for each optimally selected feature must be optimized by the newly created FEO-PFA in order to determine the weighted feature selection in the best possible way. The MDS-ATCN model classifies the outdoor data once the best-weighted features have been gathered, which helps to forecast the air quality. The proposed FEO-PFA is used to tune the hyperparameters of the TCN classifier. Finally, the high predicted accuracy of the proposed MDS-ATCN methodology to lessen the effects of air pollution was confirmed by analyzing it with existing classifiers. The optimally selected features and the predicted AQI level from the developed FEO-PFA-MDS-ATCN model is uploaded into the IOT platform called "think speak" for future use.

Experimental air quality dataset

This research uses four different datasets to validate the results of the proposed air quality prediction system, three of which are publically available and one of which is a real-time dataset. The distinct training and validation procedures are used for prediction during classification. Table 2 contains a summary of every dataset that was used in this investigation.

Author [citation]	Methodology	Features	Challenges
Liu et al.[1]	Attention, Autoencoder, GRNN, LSTM, and seq2seq model	It shows better performance in predicting air quality This method provides accurate results, and also this model is quick to train This model also provides warnings prior to sudden pollution strikes	This method provides accurate prediction results only in temporal attention, and hence the efficiency in spatial attention is still under consideration There is lagging in the functioning of this model because of its loss function
Yang et al.[2]	LSTM, GRU, and SHAP	It is efficient in identifying only the essential features for predicting air quality and the effects of these features on the quality of the air	The computational complexity is more, the memory requirement is more, and the overall efficiency of the computation is not as much expected in this method
Chen et al.[3]	Attention LSTM, Seq2Seq, and XGBoosting tree	The accuracy of predicting the quality of air by this method is higher	This model faces some issues while dealing with outlier values
Huang et al.[4]	PSO, and BP	Unlike the conventional methods, this method attains good searching ability and does not allow the system to fall into a local minimum	Lower convergence rate
Soh et al.[5]	CNN, LSTM, and ST-DNN	The prediction performance of this method is higher in the initial hours	This method does not consider the delay of a long time which results in reduced system performance
Sridhar et al.[6]	IoT, NARX	This method is highly suitable for real-time prediction of the quality of air because of the utilization of various sensors by this method	This model is highly complex, requires more memory, requires more power, and has issues regarding the security of data
Chun et al.[7]	SVM	The memory requirement of this model is low	This model is highly affected by noised data
Xu and Yoneda[8]	Autoencoder, and LSTM	It is efficient in forecasting the PM2.5 time series in a wider area	The deterministic factors like gas emission and economy are not considered by this method while predicting the quality of air

Table 1. Features and challenges of existing AQP model based on deep learning approaches.

Sl. no.	Name of the air quality dataset	Air quality dataset link	Brief description of the dataset
1	Air Quality Index (AQI) across stations and cities in India from 2015 to 2020	"https://github.com/adityarc19/aqi-india access date: 2023-03-03"	The Kaggle website provided the dataset. It includes information on air quality and AQI from numerous stations located throughout various Indian cities from 2015 to 2020, both at hourly and daily levels. It consists of various bar graphs, pie charts, and various other analyses to represent the most polluted cities in India
2	Open Government Data (OGD) Platform India	"https://data.gov.in/resources/real-time-air-quality-index-various-locations access date: 2023-03-03"	The real-time AQI from various locations in India is provided in this link. The website shows the most recent National AQI readings obtained from numerous observation locations spread out over India. The analysis includes particulate matter (PM2.5 and PM10), carbon monoxide (CO), ozone (O3), nitrogen dioxide (NO2), sulphur dioxide (SO2), and other contaminants. Because it provides information hourly, the website updates its data hourly. These data are used in this website to analyze the effectiveness of various machine learning classifiers using a variety of performance indicators
3	Comparison-of-ML-models-for-predicting-AQI	"https://github.com/Anindya-Das02/Comparison-of-ML-models-for-predicting-AQI access date: 2023-03-03"	The Open Government Data (OGD) Portal India provided the data set. The website displays current National AQI values from several observation sites located throughout India. Nitrogen Dioxide (NO2), Sulfur Dioxide (SO2), Ozone (O3), Carbon Monoxide (CO), Particulate Matter (PM2.5 and PM10), and other pollutants are analyzed. The website updates its data hourly because it offers information on an hourly basis. The performance of various machine learning classifiers is evaluated with various performance metrics by using these data on this website
4	National Air Quality Index	"https://airquality.cpcb.gov.in/AQI_India/ access date: 2023-03-03"	This website provides real-time data about the level of contaminants in the air like PM2.5, PM10, NO2, NH3, CO, and ozone in various states of India. The severity of air contamination is also provided by a color indicator

Table 2. Description of the experimental air quality dataset. The gathered real-time air quality data is represented as AQ_{da}^{gath}.

Modified heuristic-based weighted feature selection model for real time data-aided ambient air quality prediction

Data cleaning

When preparing data for analysis or predictive models, cleaning and preprocessing are essential stages. As part of preprocessing, a procedure termed normalization is used to scale numerical attributes to a standard range, usually between 0 and 1 or − 1 and 1, so as to make it possible for accurate comparisons and prevent some features from overriding the evaluation. The data cleaning method is employed to find and eliminate mistakes and irregularities from the collected air quality dataset AQ_{da}^{gath}. There are certain types of distorted data, anomalies, undesirable qualities, and irrelevant information among the input data. When working with these kinds of inappropriate, corrupted, or anomalies in the data, it results in the generation of inaccurate outcomes and the evaluation is also in inconsistent form, and the computing time is considerably high. Here, the multivariate outlier removal is performed based on the estimation of Mahalanobis distance, which is the separation of a data point from the computed centroid of all other instances where the starting point is determined by a function of the average of each variable under consideration. Every point is identified as an A and B tandem, and multivariate outliers are separated from the other cases by a specific distance. When interpreting the distances, a p-value of less than 0.001 and the accompanying B2 value are used, where the corresponding degrees of independence indicate the total number of variables. Additional methods for identifying multivariate outliers include leverage, disparity, and significance. Although leverage is assessed on a separate scale and is therefore not subject to the B2 distribution, it is connected to Mahalanobis distance. Large scores indicate that the situation may still be on the same line even if it is farther away. The degree to which the case is consistent with the other instances is evaluated by discrepancy. Leverage and discrepancy are used to calculate power, which evaluates how the coefficients change when instances are excluded. Data cleansing is performed to eliminate the duplicate features from the input collected data in order to fix these errors, which increases performance precision while lowering computation time. The resulting information after data cleaning is thus provided as CD_{kc}^{clean}.

Heuristic weighted feature selection model

It is prompted to calculate the weighted feature selection in order to increase the prediction accuracy. The suggested FEO-PFA uses the cleaned air quality data CD_{kc}^{clean} as input to find the best possible solution for the surrounding air attributes. The benefit of feature selection is that it reduces the difficulty in training and evaluating the predictor while still delivering the most accurate results for the necessary framework. Despite some significant advantages, it is unable to produce the model's expected outcomes. Even though the redundancy level is removed, it may still result in overfitting problems and reduced accuracy. Weights are optimized by the created FEO-PFA and used with the optimum features selected to increase performance. By assigning weights to its related optimally selected features, the weighted feature selection is computed. The relative influence of

each feature can be calculated by the weighted features. The suggested framework is unable to examine the quality of air with certain distorted or inaccurate information due to the relevance of the feature. So, the suggested framework has a clear goal of improving forecasting ability with the weighted features selection process. The weighted feature selection is given by Eq. (1).

$$WeF_{fv}^{opt} = wh_{ig}^{opt} * SF_{js}^{opt} \tag{1}$$

In Eq. (1), the term WeF_{fv}^{opt} denotes the optimized weighted feature. This WeF_{fv}^{opt} is obtained by multiplying each optimally selected feature SF_{js}^{opt} with the optimized weights wh_{ig}^{opt}. The weighted feature selection is done to enhance the variance of the system. Equation (2) represents the fitness function $FF1$ of the weighted feature selection.

$$FF1 = \underset{\left\{SF_{js}^{opt}, wh_{ig}^{opt}\right\}}{\arg\min} \left(\frac{1}{vrn}\right) \tag{2}$$

In Eq. (2), the term vrn indicates the variance, SF_{js}^{opt} represents the optimally selected features, and wh_{ig}^{opt} represents the optimized weights by the FEO-PFA. The features are optimally selected in the range [1, 25]. For each optimally selected feature, 25 weights are optimized, respectively. The weights are within the limit (0.01, 0.99). The objective of feature selection is to pick a particular group of attributes from a larger collection that, while reducing noise and redundancy, accurately represents the underlying structure of the data. Variance is a popular criterion for selecting features, frequently used in conjunction with filtering techniques. The spread or dispersion of values within a feature is represented by variance. There are two reasons to use variance minimization as a feature selection criterion:

Relevance: Low variance features exhibit little fluctuation across distinct dataset instances, making them less informative. Stated differently, a feature is less likely to yield meaningful information for differentiating between classes or patterns in the data if its values don't fluctuate substantially. As a result, choosing traits that are more likely to be pertinent for modelling is made easier by minimizing variation.

Redundancy: Reducing variance also aids in lowering feature redundancy. Two or more characteristics may provide duplicate information if they are highly linked, indicating that they show similar patterns of variation. The effectiveness of the feature subset is increased by choosing features with low variance since they are less probable to be identical with other features.

In Eq. (2), $\left(\frac{1}{vrn}\right)$ element symbolizes the reciprocal of the feature variance. Since smaller variance values provide bigger reciprocal values, minimizing the variance essentially entails maximizing this term. Combining everything together, it appears that the equation defines FF1 as the best set of chosen features SF_{js}^{opt} and their weights wh_{ig}^{opt} that reduces the reciprocal of the features' variability. The rationale behind this formulation could be that features with low variance are less informative, and maximizing their reciprocal helps in selecting features that exhibit more variability across instances in the dataset, thus potentially providing more discriminative power for the given task. The variance is computed as in Eq. (3).

$$vrn = \frac{\sum \left(WeF_{fv}^{opt} - mn\left(WeF_{fv}^{opt}\right)\right)^2}{fv} \tag{3}$$

where, fv represents the feature vector, and WeF_{fv}^{opt} indicates the average of optimal weighted features, and the term $mn\left(WeF_{fv}^{opt}\right)$ denotes the mean value of optimal features. The process of optimal weighted feature selection by means of the suggested EFO-PFA is provided in Fig. 2.

Optimization model: FEO-PFA

The final prediction result of the developed AQP model can be enhanced by optimizing the elements in the TCN, such as kernel size, filter size, epochs, and the random state, as well as the features and weights for weighted feature selection. This parameter optimization is achieved by the recommended FEO-PFA. The EOO[26] algorithm is utilized in this work because of its balanced exploitation and exploration and the ability to eliminate local optimum. But, this algorithm is not able to solve complex real-time problems. So, we utilize the advantages of PFA[27] to overcome the above-mentioned drawback. Therefore, the EOO algorithm is fused together with the PFA so that the FEO-PFA is developed. In the implemented FEO-PFA, the final prediction output is obtained by implementing the adaptive averaging concept in which the final location is amended by averaging the optimal position obtained from the EOO algorithm and PFA, respectively.

EOO: The eating habits of Eurasian Oystercatchers (EO) while looking for mussels served as the inspiration for EOO. EOO is developed by observing how the Oystercatchers behave and how they eat. Finding the right solution depends on finding the right equilibrium between calories ingested, time lost, and energy generated. The search method and the relevant mussel chosen by EO are shown as follows. To equalize the calories and the energy from the mussels is the primary goal of EO. The size of the mussel, the energy spent to open the mussel, and the calories obtained from eating the mussel are tightly correlated. The number of calories consumed and the amount of time needed to open mussels both increase with the increase in the mussels' size. Hence, EO wastes significant energy in the process of opening the mussel from its shell. The behaviors of EO during the exploration phase are represented by Eq. (4) and Eq. (5). The energy of the EO is computed as in Eq. (4).

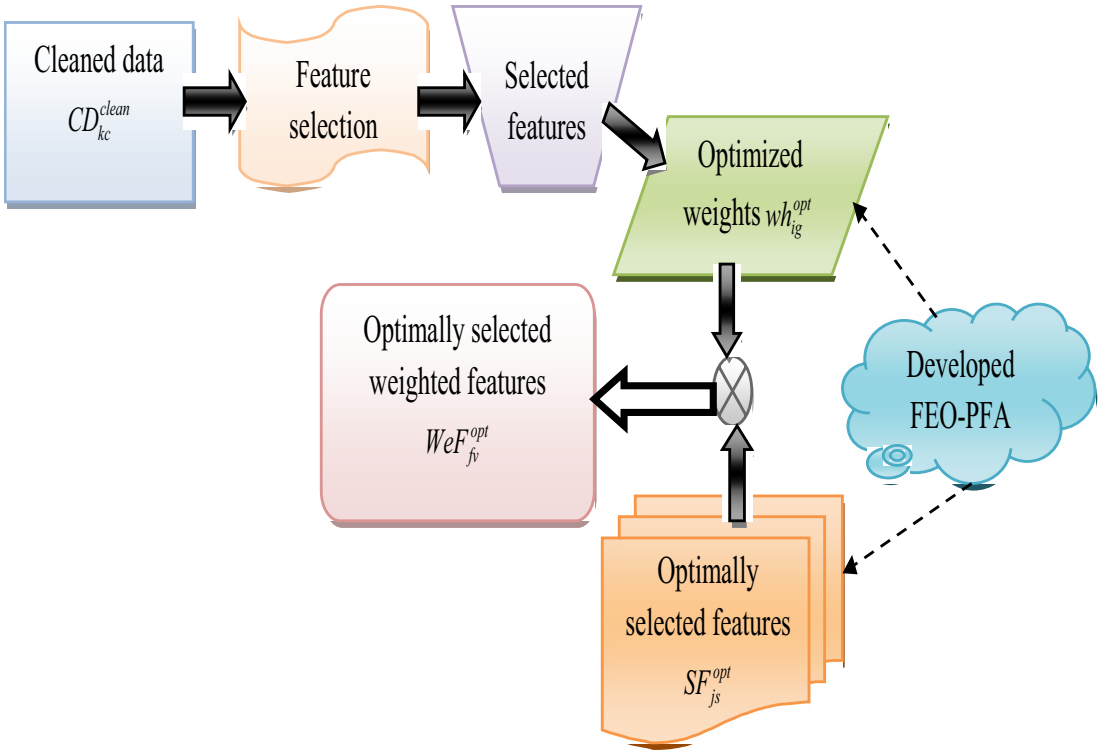

Figure 2. Pictorial illustration of the heuristic weighted feature selection model.

$$A = B + D + G * a * \left(M_b - M_{h-1}\right) \tag{4}$$

The term A in Eq. (4) is the energy of EO that is still available at last, which is inversely proportional to the iteration count,G corresponds to the size of the mussel, and is a random number between [3, 5], which indicates the limit of the ideal size of mussel and B is the amount of time needed to break the present mussel (solution), and its value depends on G, D represents the energy required at the present time, and a denotes a random number that is selected in order to increase unpredictability and uncover additional locations in the search region, which is in the range between [0, 1]. The location of the optimum mussel is given in Eq. (5).

$$M_h = M_{h-1} * I \tag{5}$$

In Eq. (5), the term M_h denotes the location of the potential mussel, and the term I denotes the caloric value that the EO gain based on the size of the mussel G. The time period B is determined in accordance with Eq. (6).

$$B = \left(\left(\frac{G-3}{5-3}\right) * 10\right) - 5 \tag{6}$$

The sizes of the ideal mussel G has been employed in Eq. (6) in order to obtain the time required to break the mussel. The value of the current energy available in the EO is determined using Eq. (7).

$$D = \left(\frac{h-1}{c-}\right) - 0.5; \quad h > 1 \tag{7}$$

The caloric amount I is found by Eq. (5), which relies on the size of the mussel G.

$$I = \left(\left(\frac{G-3}{5-3}\right) * 2\right) + 0.6 \tag{8}$$

From Eq. (8), it is obtained that the value of I is the range between [0.6, 0.8] the results from Eq. (6), which produces the result in the range [5, −5]. These parameters were chosen after extensive testing. It is crucial to note that if the time is negative, then the amount of time needed to open the mussels exceeds the bird's capacity, which is taken as a limitation. In Eq. (7), the value of D is acquired and linearly reduced from [0.5, −0.5], where h is the iteration value, which starts with H and ends at 1. The value of D is constant in the final two iterations. Hence, the terms D and B, which represent the energy and time necessary to break the potential mussel, thus, have negative values. Both the B value in Eq. (4) and the I value in Eq. (5) are dependent on a randomized number G that fluctuates constantly. This criterion makes it possible for EO to explore any location in the solution space and avoid an optimal local issue, encouraging exploration each time. The key aspects of EOO that aid in solving optimization problems are theoretically explained by the following points:

(1) The accuracy of choosing the mussel by using the duration required to break a mussel, which is computed using the bird's energy and the mussel's size as factors to estimate the anticipated location of the desired food.
(2) During each cycle, the random numbers entered during optimization aid in the exploration of new locations. As a result, avoid a local minimum issue.
(3) The random numbers utilized at every phase of optimization ensure investigation and utilization.

PFA: Inspired by the swarm algorithm, a unique meta-heuristic algorithm called PFA is built with a different mathematical model. It mimics the haphazard behavior and movement of the animal in swarms that follow their head to an adjacent site in search of prey or sustenance. Changes in a leader are possible if the search's objective is not attained. The motion of the competitor of the animal, which is gathered in groups in a community form, is what drives the algorithm process. The head of the swarm and its competitors works together to determine the best path to the destination. Based on the forces and direction in the multidimensional area, the path's direction is enhanced. At any point throughout the search, the contestant in the optimal location is regarded as the head of the swarm. This candidate is denoted as the pathfinder. In the current iteration, pathfinder and its position are seen as the optimal solution, and the other competitors adopt it. A vector indicating the competitors' movement location in many dimensions is used to organize the suggested solutions. To control how the rival behaves in the exploration area, four parameters have been modified. Each cycle simultaneously generates the competitors' vibration ν and oscillating frequency τ. The attraction factor α adjusts the random space of separation necessary to ensure the availability of the head of the swarm, and the communication factor ς maintains the movement with regard to the surrounding competitors. The major motive is to estimate the optimal position of the rival. Equation (9) is utilized to figure out the prey and to follow the pathfinder.

$$C(i+\Delta i) = C^0(i)\cdot d + E_f + K_J + \nu \tag{9}$$

In Eq. (9), the term C represents the vector for the position, K_J represents the force that is dependent on the pathfinder's position, d denotes the identity vector, E_f denotes the communication that takes place among two rivals C_f and C_k, and i denotes the period. The position of the pathfinder is updated using Eq. (10)

$$C_J(i+\Delta i) = C_J(i) + \Delta C + \tau \tag{10}$$

The term ΔC in Eq. (10) denotes the value that is estimated by considering the space between the two distinct positions of the pathfinder and C_J denotes the position vector of the pathfinder. The above two equations, Eqs. (9), and (10) are upgraded for solving the problem as follows.

$$\overrightarrow{C}_f^{o+1} = \overrightarrow{C}_f^{o} + \vec{Q}_1 * \left(\overrightarrow{C}_k^{o} - \overrightarrow{C}_f^{o}\right) + \vec{Q}_2 * \left(\overrightarrow{C}_J^{o} - \overrightarrow{C}_f^{o}\right) + \nu \tag{11}$$

In Eq. (11), the terms $\vec{Q}_1$ and $\vec{Q}_2$ denotes two trajectory vectors in random. The value of $\vec{Q}_1 = \alpha \cdot q_1$, and the value of $\vec{Q}_2 = \varsigma \cdot q_2$, where q_1 and q_2 denotes the random movement generated homogeneously. The value of q_1 and q_2 is in the range $[-1, 1]$. The terms $\overrightarrow{C}_k^{o}$ and $\overrightarrow{C}_f^{o}$ are the position vectors of the two rivals k and f at the current iteration o. The value of ν can is determined using Eq. (12).

$$\nu = \left[1 - \left({}^{o}/_{O}\right)\right] p_1 \cdot N_{os}; \quad N_{os} = \left\|C_f - C_k\right\| \tag{12}$$

The term O denotes the recommended maximum iteration count, o denotes the current iteration, and N_{os} represents the distance of separation between the two rivals. To regulate the mechanism of the algorithm, the attraction factor α, and communication factor ς values are changed. Each rival stops moving and loses track of the head of the swarm when α and ς equal to ∞. Each rival move independently and randomly throughout the area when α and ς equal to 0. When α and ς are either less than 1 or greater than 2, then the affiliate rival is unable to produce an optimal solution. Hence, it is vital that the values of α and ς should be between [1, 2]. The position is further upgraded as in Eq. (13).

$$\overrightarrow{C}_J^{o+1} = \overrightarrow{C}_J^{0} + 2q_3 \cdot \left(\overrightarrow{C}_J^{0} - \overrightarrow{C}_J^{0-1}\right) + \tau \tag{13}$$

The term q_3 in Eq. (13) is a random vector of the rival. In the event that the terms $\vec{Q}_1 * \left(\overrightarrow{C}_k^{o} - \overrightarrow{C}_f^{o}\right)$ and $\vec{Q}_2 * \left(\overrightarrow{C}_J^{o} - \overrightarrow{C}_f^{o}\right)$ in Eq. (11) or the term $2q_3 \cdot \left(\overrightarrow{C}_J^{0} - \overrightarrow{C}_J^{0-1}\right)$ in Eq. (13) became 0, then ν and τ can arbitrarily move every rival with suitable values across several paths. The oscillating frequency τ can be computed as in Eq. (14).

$$\tau = p_2 \cdot \exp^{\left(\frac{-2o}{O}\right)} \tag{14}$$

The term p_2 in Eq. (14) represents a random value in the limit $[-1, 1]$. The divergence and convergence of the PFA searching are influenced by the values ν and τ. It has the power to speed up or slow the algorithm. For the search to succeed without diverging between them in each iteration, values ν and τ should be between [1,

2]. The competitor can swiftly leave their positions without finding a solution if both p_1 and p_2 are beyond the range [−1, 1]. The algorithm of the recommended FEO-PFA is provided in Algorithm 1.

Load the population of EO and pathfinder.
Load the parameters in EOO and PFA
Compute the fitness of every search agent
For $(r \to 1 to R)$
For $(t \to 1 to T)$
While $(stopping\ \ criterion)$ do
Compute the location using the EOO algorithm
Determine B , D and I using Eq. (6), Eq. (7), and Eq. (8)
Amend the location using Eq. (4), and Eq. (5)
Determine the location using the EOO algorithm UL_{js}^{EOO}
Compute the location using PFA
Determine the value of $\vec{C}_f^{\,o+1}$ and $\vec{C}_J^{\,o+1}$ using Eq. (11), and Eq. (14)
Determine the location using PFA UL_{hd}^{PFA}
Obtain the final location by averaging UL_{js}^{EOO} and UL_{hd}^{PFA}
End
Figure out the best location
Amend the fitness of all solution
End
End

Algorithm 1: Recommended FEO-PFA.

The flowchart of the developed FEO-PFA is depicted in Fig. 3.

Multiscale depthwise separable adaptive temporal convolutional network for real time data-aided ambient air quality prediction

Temporal convolutional network

In consecutive modeling, the convolutional technique is typically more effective than recurrent models. This study used a TCN[28] for the AQP sequential model. The TCN framework is capable of handling both input and output sequences of varying sizes when using the time series forecasting technique. The TCN concept essentially rests on two key tenets. TCN utilizes temporal convolution; therefore, the results are independent of the inputs' future evolution. It only relies on the input received before c if it generates output at a period c. The TCN framework can take a series of any size and convert it to an output vector of the very same size; therefore, there won't be any data loss from the forward to the history. To accomplish this goal, a 1D Fully Connected (FC) layer is used. The output P at period c can only be affected by a small number of past inputs at once $(c-1,\ c-2, \ldots)$. Hence, using past data of greater scale necessitates a bigger receptive field. To acquire a broader receptive field, a greater size filter or denser connection is a necessity. The greater size of the filter is the primary cause of non-convergence problems, whereas complex deep connectivity is the cause of problems with training the model. The model's effectiveness is hampered by problems with non-convergence and training the model. Dilated convolution employing residual mapping is used to get a bigger past data volume to get over this problem. Dilated convolution doesn't need the pooling operation and expands the range of perception through a sequence of dilated convolution, which is utilized in TCN to solve the problem of data leakage. This technique eventually incorporates excellent information from lengthy tracking.

Dilated convolutions are crucial for time series datasets in which long-term correlations are important, such as in the forecasting of air quality. Equation (15) provides the dilated convolution function e on the sequence's element P.

$$e(l) = \left(P *_g e\right)(l) = \sum_{m=0}^{j-1} e(m) P_{l-g \cdot m} \tag{15}$$

In Eq. (15), j stands for filter size, $l - g \cdot m$ stands for past direction, and g stands for dilation factor. Residual connections are another important element of the TCN structure. TCN substitutes a general residual unit for convolution. Convolutional layers of TCN units are layered in the residual blocks, which are periodic frames. It integrates the most recent input with the output of the preceding convolution. As a result, the output of the temporal unit is approximated in relation to the input. The deep channel's learning difficulties are resolved by the node. Its branch includes results that are appended to the input after a sequence of transformations. As a result, TCN has a quick interface to perform the residual mapping from P to $e(P)$. The mapping function of TCN is determined as in Eq. (16).

$$n = \gamma(P + e(P)) \tag{16}$$

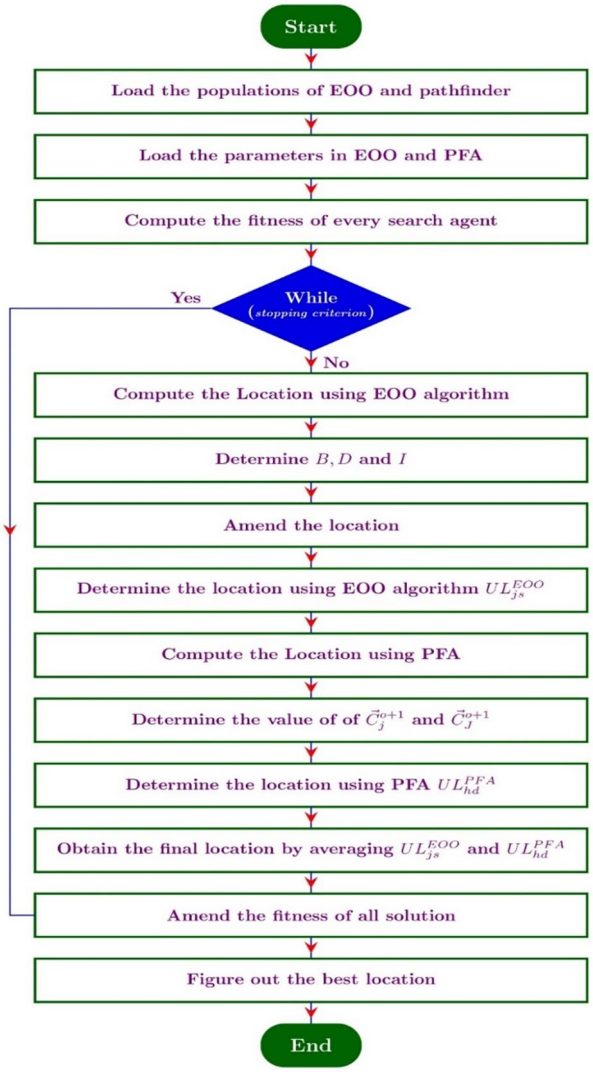

Figure 3. Flowchart of the recommended FEO-PFA.

The term γ in Eq. (16) is the activation function that is not linear in nature. There is additional activation, dropout layer, pooling, weight normalization, and 1D convolution in the residual block.

Developed MDS-ATCN for ambient air quality prediction

The weighted selected features WeF_{fv}^{opt} are given as input to the recommended MDS-ATCN framework. MDS-ATCN is a combination of both multiscale processing and depth-wise separable convolution. Multiscaling is nothing but the process of operating on each data at various scales in order to obtain local and global attributes from it. In depth wise separable convolution, the convolution operation is divided into point-wise and depth-wise convolutions. Depth-wise convolution operates individually on every channel whereas the point-wise convolution fuses the results from the depth-wise convolution. The temporal and spatial network are the two main parts of MDS-ATCN. To extract spatial characteristics, the spatial network individually processes every air quality data. The series of spatial information is subsequently processed by the temporal network to identify temporal dependencies. From the MDS-ATCN model, the final prediction result about the air quality is obtained. The elements in the TCN, like filter size, random state, kernel size, and epochs, are optimized by using the implemented FEO-PFA. The Fitness function of the proposed MDS-ATCN framework for AQP is given by Eq. (17).

$$FF2 = \underset{\left\{ks_{ls}^{TCN}, fs_{ta}^{TCN}, ec_{ix}^{TCN}, rs_{jw}^{TCN}\right\}}{\arg\min} \left(qs + fg_{\infty}\right) \tag{17}$$

In Eq. (17), the term fg_{∞} denotes the infinity norm, qs denotes the Root Mean Square Error (RMSE), ks_{ls}^{TCN} denotes the optimized kernel size of TCN, fs_{ta}^{TCN} denotes the optimized filter size, ec_{ix}^{TCN} denotes the optimized epochs in TCN, and rs_{jw}^{TCN} denotes the optimized random state of TCN. The ranges in which the epochs are

optimized are [50, 100]. The range in which the kernel size is optimized is [1, 7]. The limit in which the filter size is tuned is [0, 3] that the sizes are mentioned as [16, 32, 64, 128]. The boundary of the random state optimization is [0, 1]. The optimization of these attributes is done in order to minimize the RMSE and the Infinity Norm. RMSE *qs* is evaluated as in Eq. (18).

$$qs = \sqrt{\frac{\sum_{com=1}^{co} \left(rl_{com1} - prd_{com2}\right)^2}{co}} \tag{18}$$

The terms *prd* in Eq. (18) represent the predicted value, and the term *co* represents fitted points count, *com* denotes the computational value that has to be added to every fitted point, and *rl* represents the real value. The Infinity Norm fg_{∞} is calculated using Eq. (19).

$$fg_{\infty} = \max_{1 \le com \le co} \left| fg_{com} \right| \tag{19}$$

The term *fg* in Eq. (19) represents a matrix. The diagrammatic representation of the generated MDS-ATCN-based AQP framework is given in Fig. 4.

Our predictive model is trained based on a combined and cleaned dataset drawn from four public air quality databases, all of which are very complex. This is because preprocessing is time-consuming, expensive, and indispensable for data integrity and increasing predictive accuracy. On the other hand, Figs. 5 and 6 concern the Air Quality Index dataset and underline the key phases of the preprocessing pipeline. Figure 5a shows the AQI dataset before outlier removal. The plot indicates raw data with probable anomalies, which may bias results of this predictive model. Physically, Fig. 5b indicate a much cleaner dataset; that is, outlier removal will ensure a more reliable set of data for training the model. This will ensure that the extreme values are removed from the dataset and will not blur expected patterns in the data; hence, the dataset is safeguarded for analysis. Figure 6 shows the sample distribution of features for the preprocessed AQI dataset. The graph shows some variables' distributions after data cleaning. This plot clearly illustrates the improvement in structure and quality of the dataset, thus showing the impact of preprocessing steps involved, from outlier removal to normalization and feature scaling. Proper cleaning of data and its preprocessing are hence a very essential step for the optimization of MDS-ATCN's performance. As illustrated, it attains better predictive accuracy by refinement of these variables through the proposed FEO-PFA. All rigorous preprocessing guarantees that high-quality data is fed into the MDS-ATCN; no doubt, this has greatly contributed to the reduction of the Average Cost Function, Mean Absolute Error, and Root Mean Squared Error metrics, underlining the effectiveness and superiority of our approach over traditional methods.

The frequency of features for the input data is displayed in Fig. 5a,b before and after outlier removal procedures, where sample instances of features are provided for analysis. The results show that utilizing Mahalanobis distance to effectively remove outliers greatly aids in producing preprocessed data that is of higher quality.

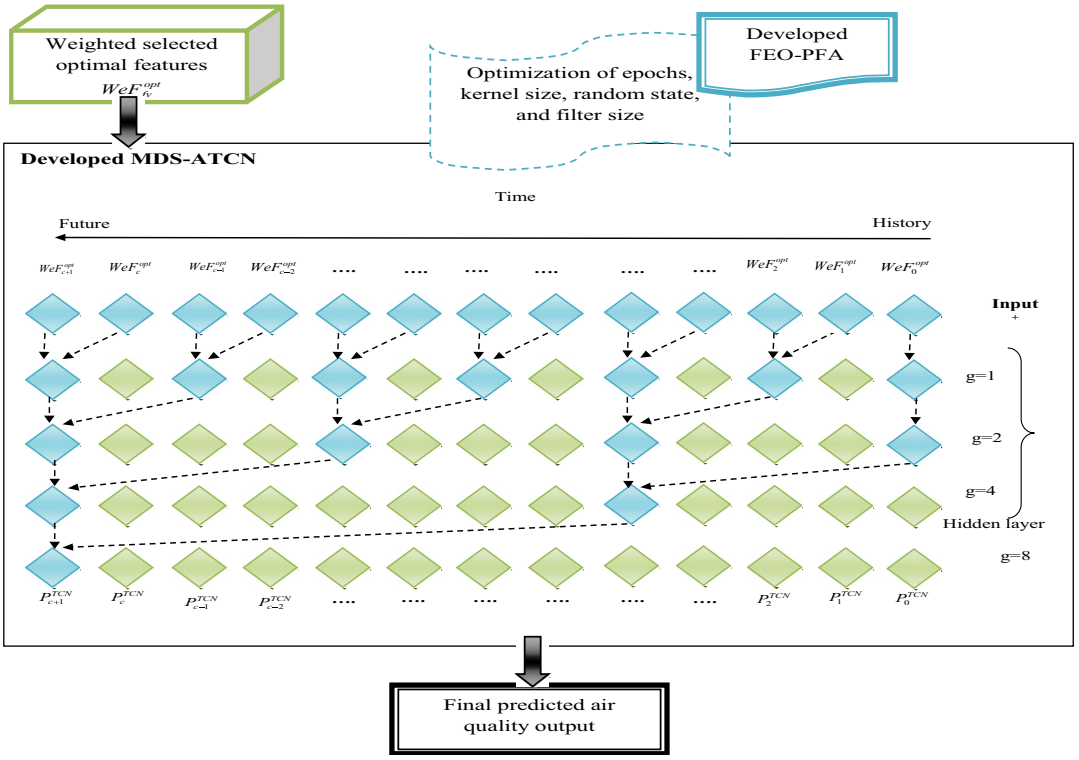

Figure 4. Diagrammatic representation of the generated MDS-ATCN framework for ambient AQP.

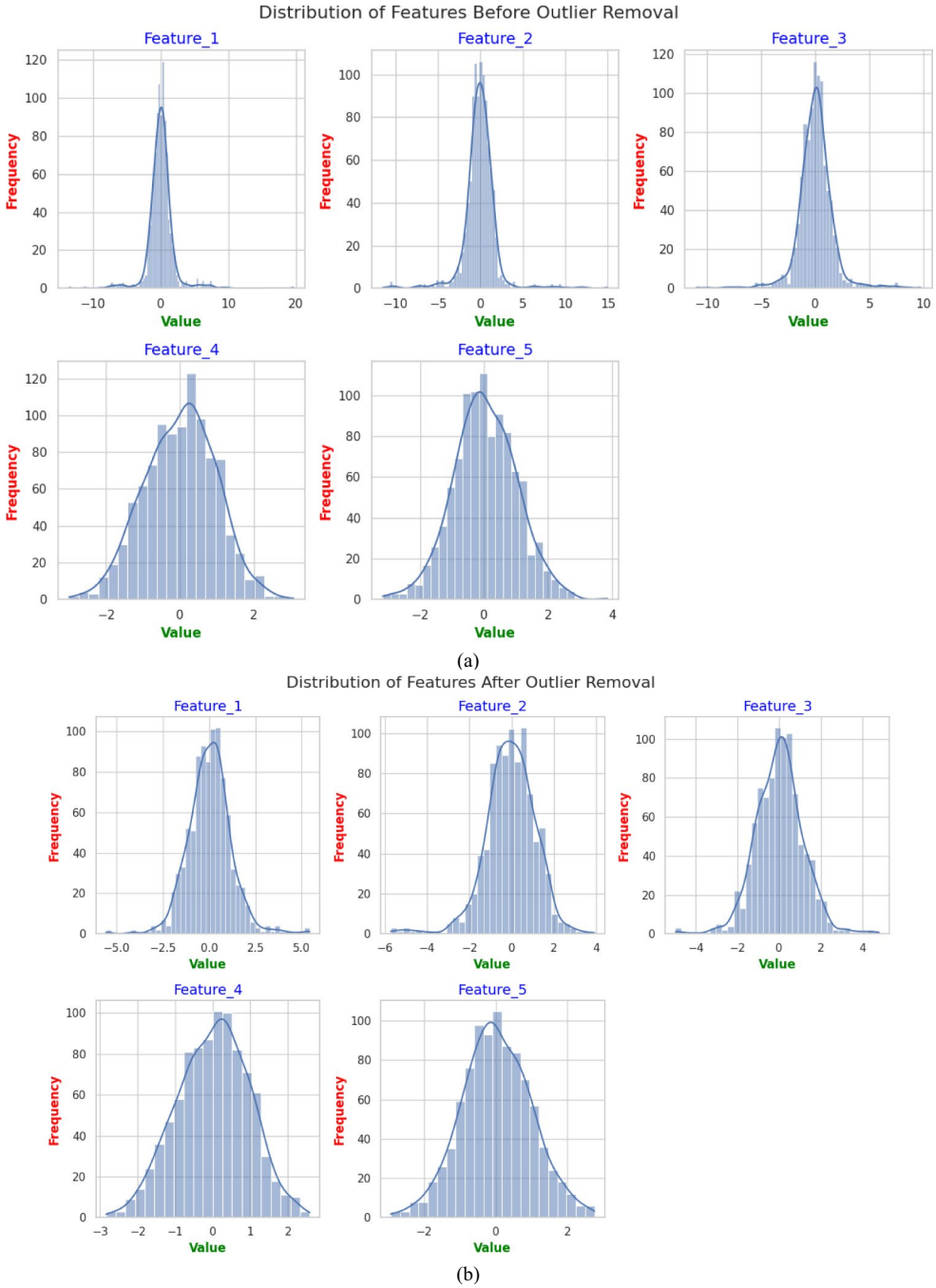

Figure 5. (**a**) Before outlier removal and (**b**) after outlier removal for Air Quality Index (AQI) dataset.

Consequently, Fig. 6 also provides a visual representation of the sample distribution of features for the pre-processed data. Figure 7 shows the relevance score for the features selected from the dataset in the best possible way. Additionally, Fig. 8 shows the training and testing data partitioning according to the sample index. The significance of feature selection, preprocessing, data splitting, and outlier removal for classification are examined in light of these stages' inferences.

Result and discussion

Experimental setup

The constructed AQP framework based on deep learning was implemented and assessed using the Python paradigm. The assessment outcomes were listed in the upcoming section. The AQP framework that operates on the principle of deep learning techniques was built with a maximum iteration count of 25 and a population size of 10, respectively. The AQP model based on MDS-ATCN was evaluated and contrasted with various predictors like Convolutional Neural Network (CNN)[29], Deep Neural Network (DNN)[30], Long Short Term Memory

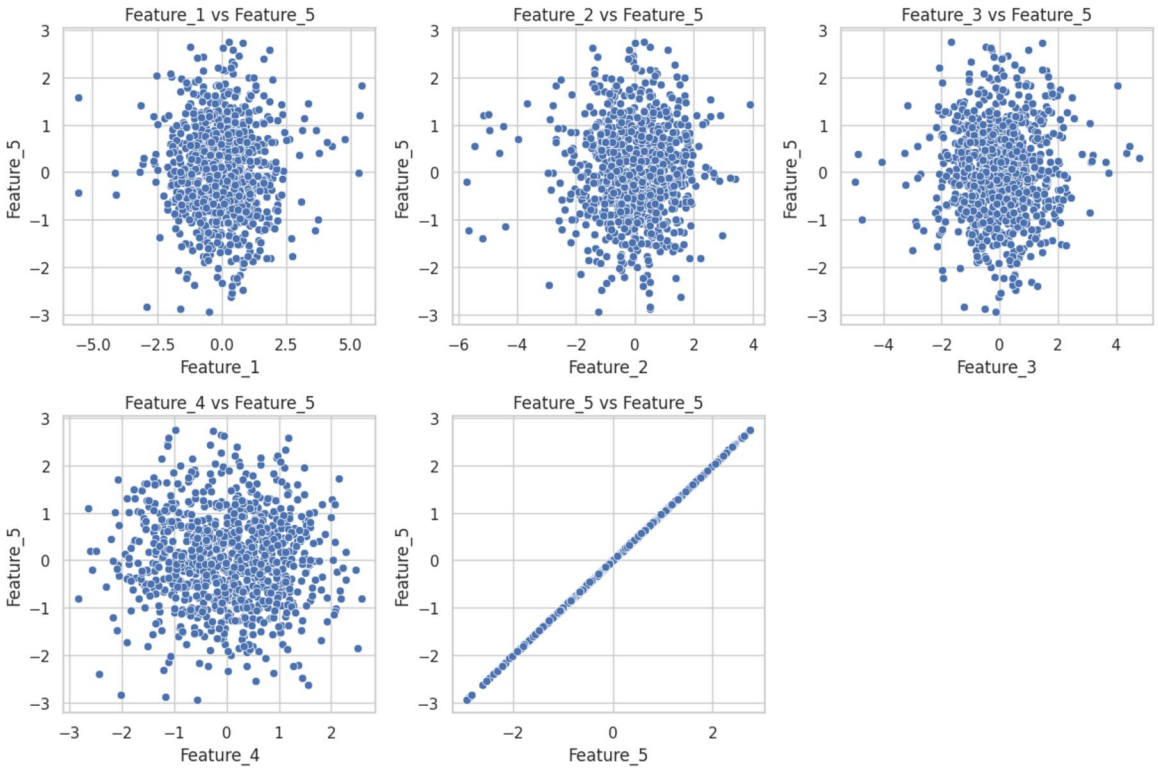

Figure 6. Sample distribution of features for the preprocessed Air Quality Index (AQI) dataset.

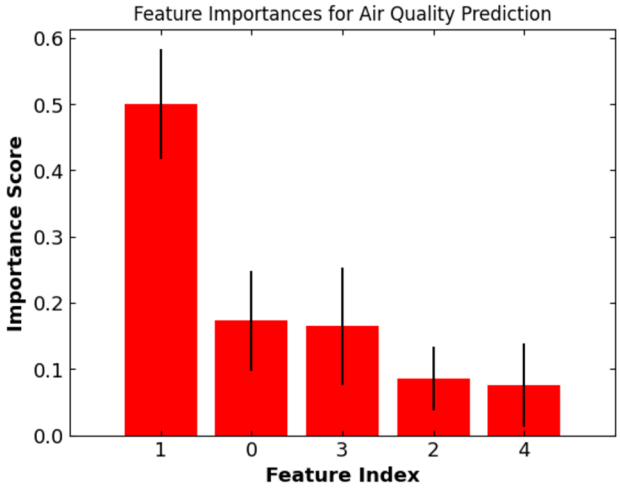

Figure 7. Feature importance for air quality prediction.

(LSTM)[31], and Temporal Convolution Network (TCN)[28], and compared with conventional heuristic algorithms such as Jaya Algorithm[32], Deer Hunting Optimization Algorithm (DHOA)[33], Salp Swarm Algorithm (SSA)[34], and EOO[26] for illustrating the precise prediction of the suggested AQP model.

We chose the following benchmarking models for our work: Convolutional Neural Network, Deep Neural Network, Long Short-Term Memory, and Temporal Convolutional Network. These models have been previously proven to be quite efficient in air quality prediction and time-series forecasting tasks. Thereby, every benchmarking model has its strengths that are targeted at different facets of the prediction challenge. One of the main characteristics that makes CNNs very efficient in capturing spatial dependencies is highly critical in environmental monitoring applications. The DNN will learn complex representations from large and high-dimensional datasets; this becomes particularly central to the multifaceted nature of air quality data. LSTMs have long been particularly relevant for modeling the temporal dependencies inherent in air quality datasets because, like a specially designed form of RNNs, they have a generic ability to capture long-term dependencies in sequential data. TCNs

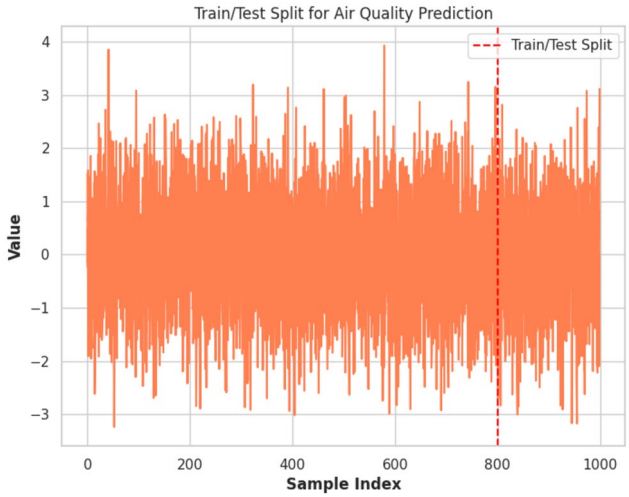

Figure 8. Training and testing data.

apply convolution in time and thus ensure a number of benefits in capturing time patterns without common limitations for recurrent networks—especially the vanishing gradients—which provides robust performance in time-series forecasting tasks.

These models are also used for benchmarking purposes, as they have been implemented in a lot of recent literature, proving their effectiveness and reliability for similar predictive tasks. For instance, CNNs have already found an application in numerous environmental monitoring scenarios with spatially distributed data, which proves their capacity to deal with this kind of data. DNNs are capable of modeling complex nonlinear relationships; therefore, they fit well for air quality prediction, which is indeed very complex. LSTMs have, therefore, gained much popularity due to the strength they possess in the treatment of sequential data, since they remain and use information for very extensive periods of time inherently, which is very vital for the prediction of correct air quality. TCNs, by processing sequences with convolutional layers, offer a compelling alternative to classical RNNs with better scalability and long-term sequence processing efficiency.

Validation metrics

The better operation of the recommended AQP framework is evaluated using the following error functions, as shown below.

(a) MEP *us* is determined using the formula provided in Eq. (20).

$$us = \frac{100\%}{co} \sum_{com=1}^{co} \frac{rl - prd}{rl} \tag{20}$$

(b) SMAPE *ol* is calculated using Eq. (21).

$$ol = \frac{100\%}{co} \sum_{com=1}^{co} \frac{|prd - rl|}{\frac{(|rl|+|prd|)}{2}} \tag{21}$$

(c) MAE *ia* is evaluated using the formula given in Eq. (22)

$$ia = \frac{\sum_{com=1}^{co} |prd - rl|}{co} \tag{22}$$

(d) MASE *ks* is computed using Eq. (23).

$$ks = mn \left(\frac{|prd|}{\frac{1}{co-1} \sum_{com=1}^{co} |rl_{com} - rl_{com-1}|} \right) \tag{23}$$

The term *mn* in Eq. (23) represents the mean value.
(e) L1-Norm fg_1 is determined using Eq. (24).

$$fg_1 = \sum_{com} |fg_{com}| \tag{24}$$

(f) L2-Norm fg_2 is evaluated using Eq. (25).

$$fg_2 = \left(\sum_{com=1}^{co} fg_{com} \right)^2 \tag{25}$$

(g) The RMSE is estimated using Eq. (26).

$$ia = \sqrt{\left(\frac{\sum_{com=1}^{co} |prd - rl|}{co} \right)^2} \tag{26}$$

Algorithmic analysis

The algorithmic analysis of the recommended MDS-ATCN-based AQP model is shown in Fig. 9. The MAE of the proposed MDS-ATCN-based AQP model is 51.72%, 48.15%, 33.33%, and 17.65% lesser than the JAYA-MDS-ATCN, DHOA-MDS-ATCN, SSA-MDS-ATCN, and EOO-MDS-ATCN algorithms, respectively for dataset 4.

Prediction evaluation on recommended model

The prediction evaluation of the implemented MDS-ATCN-based AQP model is shown in Fig. 10. The MAE of the proposed MDS-ATCN-based AQP model is 60%, 41.96%, 52.63%, and 28% lesser than the CNN, DNN, LSTM, and TCN prediction models, respectively for dataset 1.

The cost function examination of the suggested MDS-ATCN model for AQP is given in Fig. 11. The cost functions of the recommended MDS-ATCN-based AQP model are 13.04%, 13.04%, 5.88%, and 9.09% lesser than the JAYA-MDS-ATCN, DHOA-MDS-ATCN, SSA-MDS-ATCN, and EOO-MDS-ATCN algorithms, respectively for iteration 10 of dataset 1.

The algorithmic and prediction model comparison of the RMSE of the developed MDS-ATCN-based AQP framework is given in Figs. 12 and 13. The RMSE of the proposed MDS-ATCN-based AQP framework is 36.25%, 35.03%, 26.09%, and 25% lesser than the CNN, DNN, LSTM, and TCN prediction models, respectively for dataset 4.

Statistical report comparison

The statistical analysis of the generated MDS-ATCN-based AQP model is provided in Table 3. The median value of the suggested FEO-PFA-MDA-ATCN AQP model is 17.44%, 3%, 17.7%, and 6.86% lesser than the DHOA-MDS-ATCN, EOO-MDS-ATCN, JAYA-MDS-ATCN, and SSA-MDS-ATCN algorithms, respectively for dataset.

Comparison of various algorithms with the suggested AQP framework

The algorithmic analysis of the generated MDS-ATCN-based AQP model with other existing algorithms is provided in Table 4. The RMSE of the suggested FEO-PFA-MDS-ATCN AQP framework is 29.46%, 38.1%, 14.43%, and 44.27% lesser than the SSA-MDS-ATCN, DHOA-MDS-ATCN, EOO-MDS-ATCN, and JAYA-MDS-ATCN algorithms, respectively for dataset 1.

Comparison of other prediction models with the suggested AQP framework

The classifier-based analysis of the built MDS-ATCN-based AQP model with other conventional predictors is provided in Table 5. For dataset 2, the MAE of the recommended FEO-PFA-MDS-ATCN is 74.87%, 60.5%, 81.66%, and 69.09% lesser than the DNN, TCN, CNN, and LSTM classifiers, respectively.

Conclusion

We developed and analyzed a deep learning based AQP framework. The process began with collecting real-time ambient air datasets from multiple publicly accessible databases. The proposed FEO-PFA is utilized to select the optimal features from the cleaned data, which were then weighted optimally. The innovative FEO-PFA was pivotal in optimizing the weights for each chosen feature. With the best-weighted features determined, the MDS-ATCN model forecasts outdoor air quality. Several parameters within the MDS-ATCN model, including kernel size, random state, filter size, and epochs, were further optimized using the FEO-PFA. A series of experiments were conducted, allowing us to measure the efficiency of the proposed MDS-ATCN AQP model against various predictors and existing heuristic algorithms. Our results depict that the average cost-function drop was 5.5%, while those for MAE and RMSE were 28% and 14%, respectively, against traditional methods—evidencing a higher precision and efficiency predicted by the proposed MDS-ATCN framework for ambient air quality. Finally, the optimally selected features along with predicted AQI levels obtained from FEO-PFA-MDS-ATCN model have been integrated in the IoT platform "ThinkSpeak" for future applications.

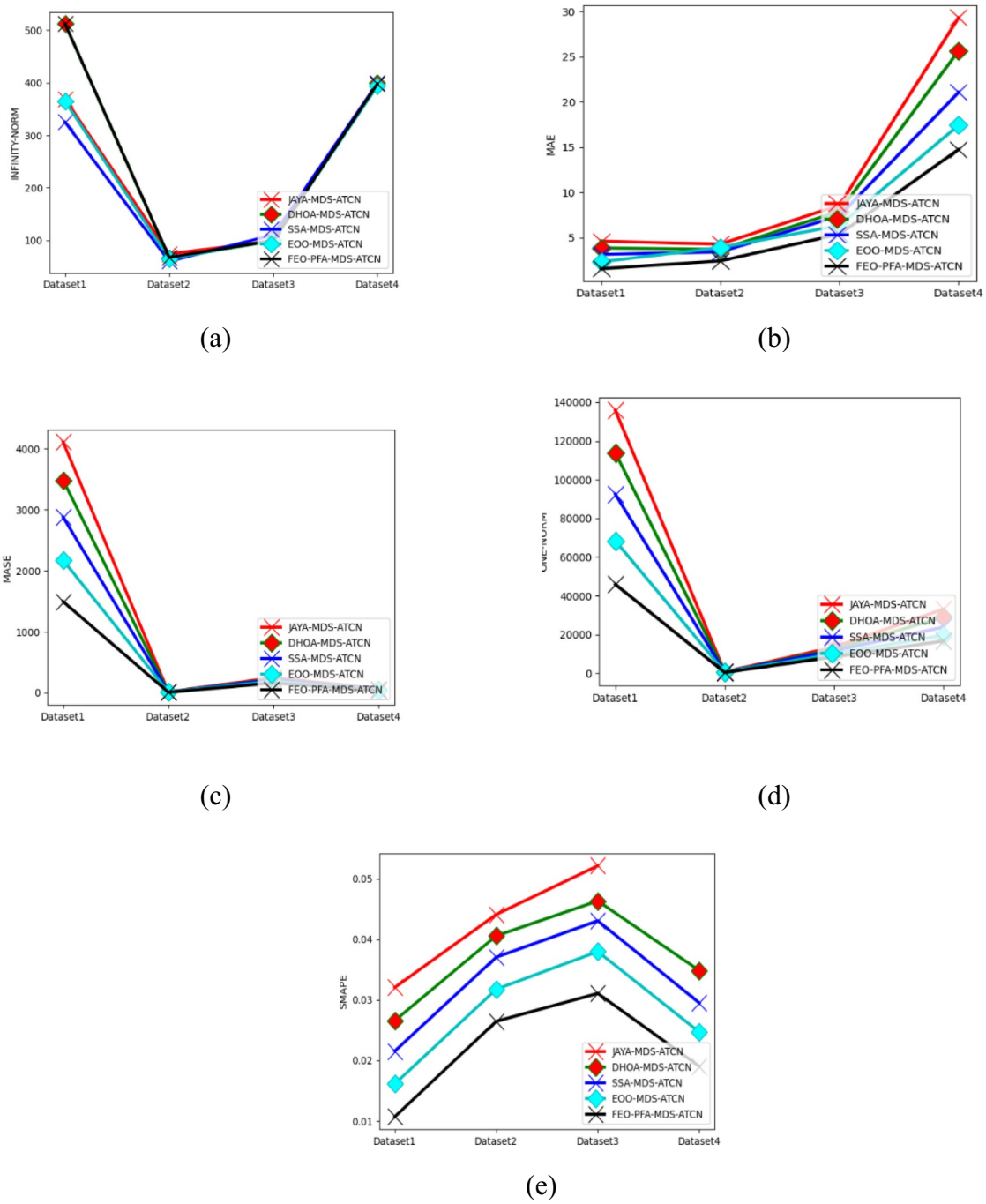

Figure 9. Heuristic algorithm-based evaluation on the recommended AQP framework in terms of "(**a**) Infinity norm, (**b**) MAE, (**c**) MASE, (**d**) One norm, and (**e**) SMAPE".

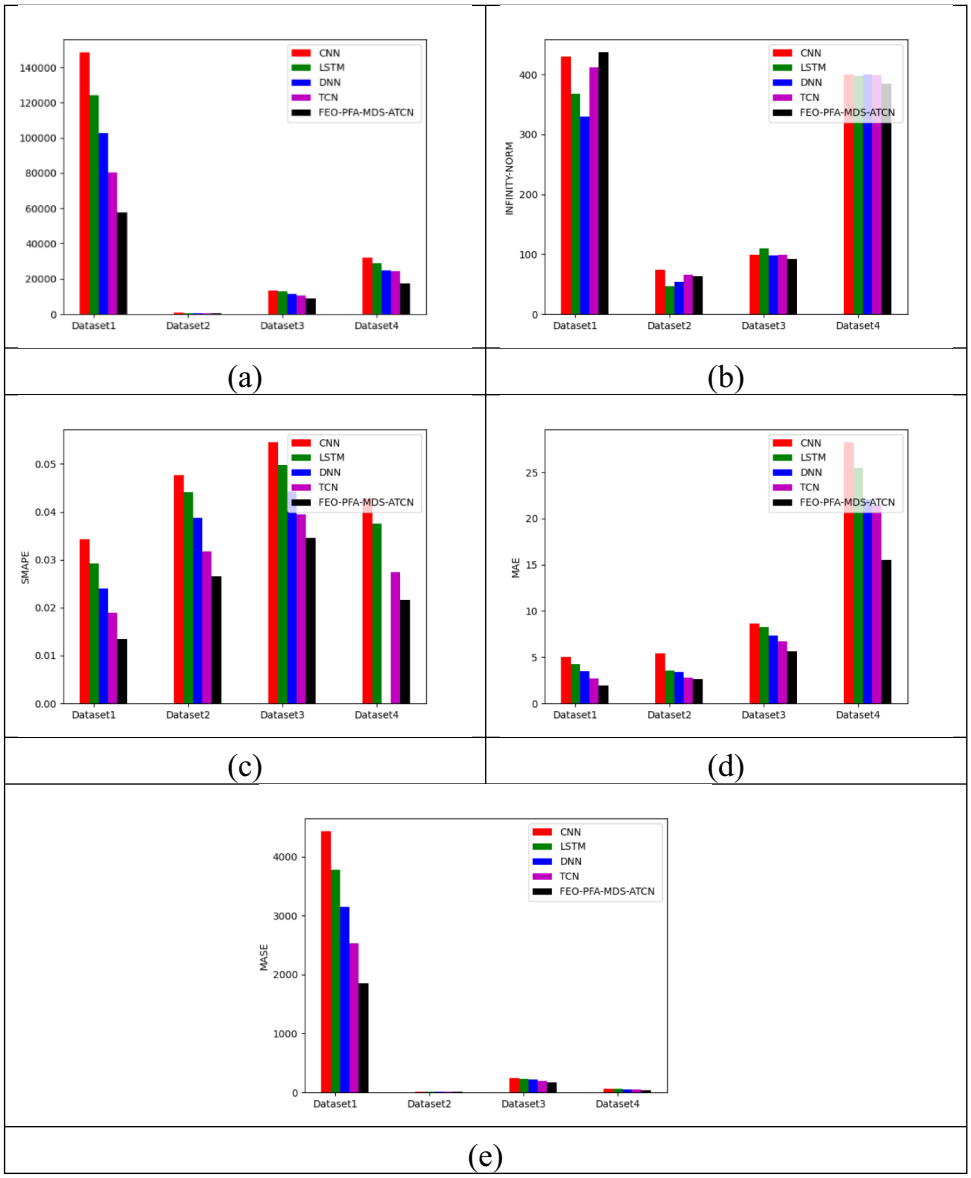

Figure 10. Prediction-based Assessment on the Generated AQP framework in terms of "(**a**) One norm, (**b**) Infinity norm, (**c**) SMAPE, (**d**) MAE, and (**e**) MASE".

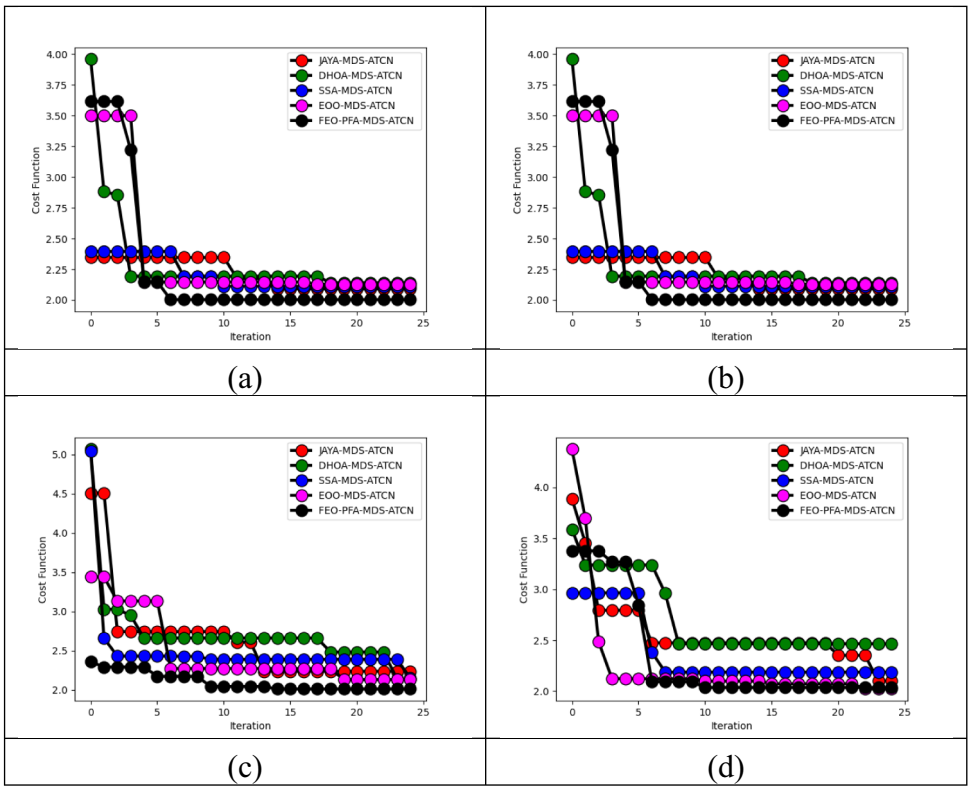

Figure 11. Cost function analysis on the implemented AQP model with respect to "(**a**) Dataset 1, (**b**) Dataset 2, (**c**) Dataset 3, and (**d**) Dataset 4".

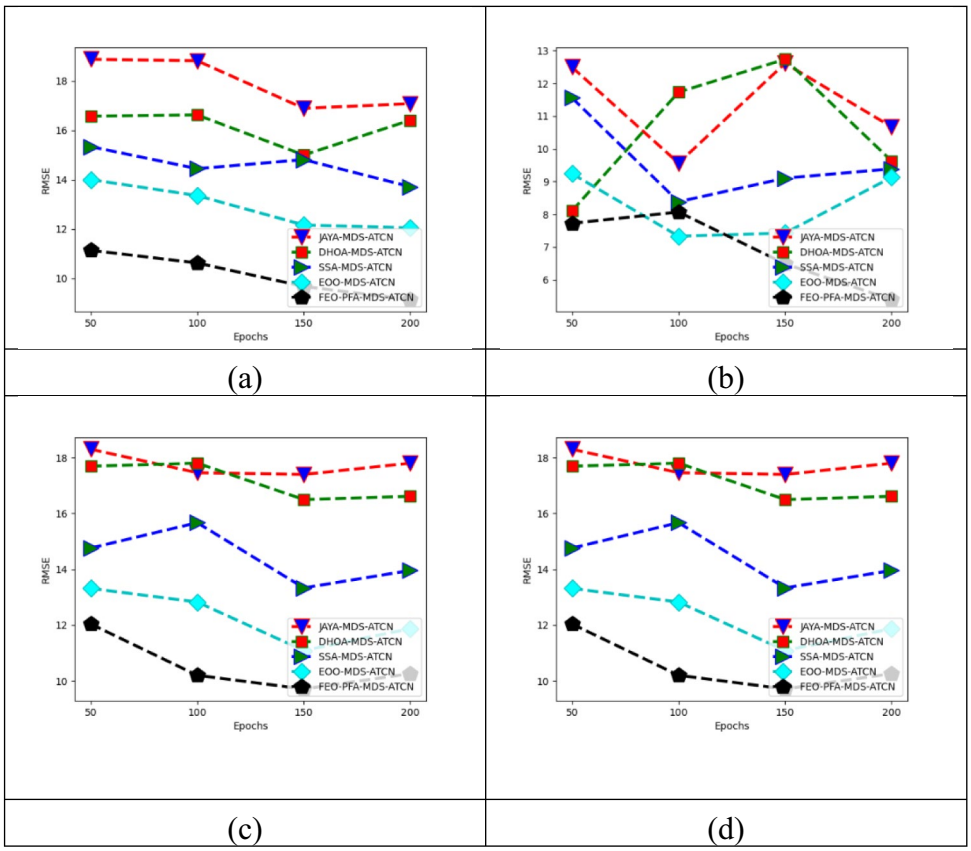

Figure 12. RMSE Assessment on the Suggested AQP model with various algorithms with respect to "(**a**) Dataset 1, (**b**) Dataset 2, (**c**) Dataset 3, and (**d**) Dataset 4".

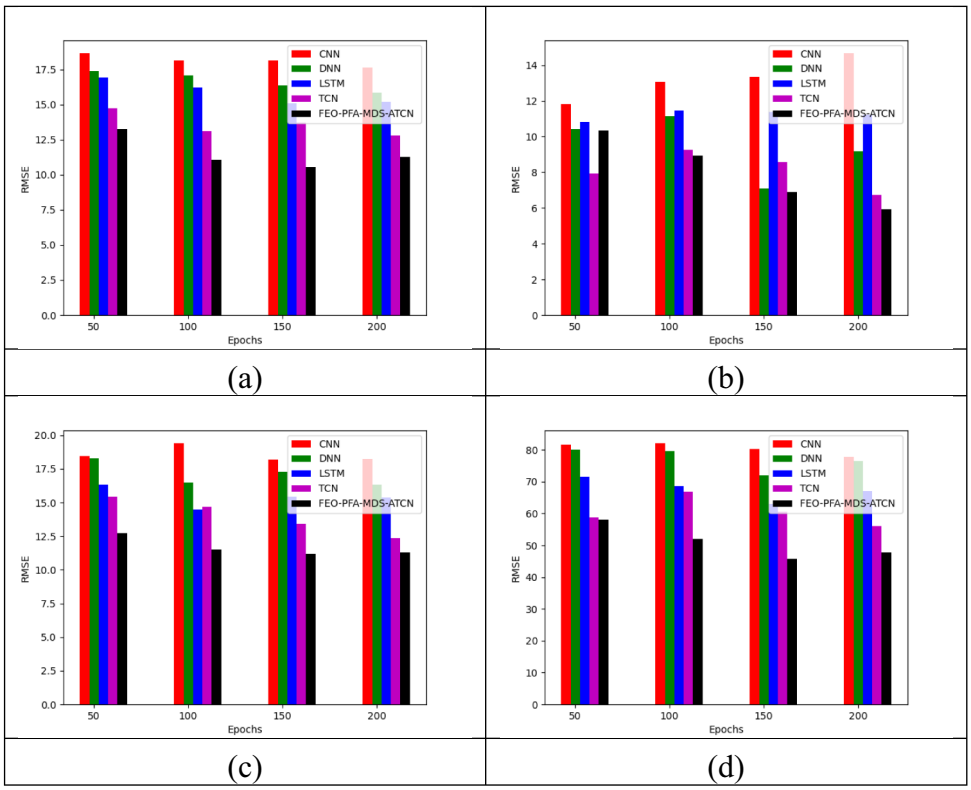

Figure 13. RMSE assessment on the suggested AQP model with other prediction models with respect to "(**a**) Dataset 1, (**b**) Dataset 2, (**c**) Dataset 3, and (**d**) Dataset 4".

Terms/algorithm	SSA-MDS-ATCN[31]	EOO-MDS-ATCN[26]	JAYA-MDS-ATCN[29]	DHOA-MDS-ATCN[30]	FEO-PFA-MDS-ATCN
Dataset 1					
Worst	2.396	3.500	2.346	3.959	3.615
Best	2.112	2.128	2.091	2.141	2.003
Mean	2.201	2.358	2.213	2.304	2.257
Standard deviation	0.124	0.499	0.121	0.387	0.555
Median	2.112	2.148	2.175	2.194	2.003
Dataset 2					
Worst	2.678	4.703	3.111	4.373	3.993
Best	2.158	2.111	2.114	2.019	2.014
Median	2.158	2.111	2.114	2.019	2.070
Standard deviation	0.140	0.508	0.410	0.781	0.543
Mean	2.205	2.215	2.503	2.615	2.246
Dataset 3					
Best	2.150	2.130	2.237	2.136	2.011
Mean	2.507	2.469	2.630	2.720	2.102
Median	2.388	2.272	2.609	2.661	2.042
Standard deviation	0.524	0.440	0.601	0.520	0.116
Worst	5.047	3.439	4.510	5.068	2.358
Dataset 4					
Worst	2.965	4.372	3.886	3.587	3.376
Mean	2.381	2.259	2.576	2.716	2.337
Median	2.186	2.099	2.474	2.466	2.036
Standard deviation	0.330	0.538	0.370	0.375	0.522
Best	2.186	2.022	2.101	2.466	2.036

Table 3. Statistical report of the implemented AQP model with conventional heuristic algorithms for four datasets.

Metrics/algorithm	DHOA-MDS-ATCN[26]	JAYA-MDS-ATCN[31]	EOO-MDS-ATCN[30]	SSA-MDS-ATCN[29]	FEO-PFA-MDS-ATCN
Dataset 1					
MEP	2.155	2.592	1.225	1.693	0.740
SMAPE	0.025	0.030	0.014	0.019	0.008
MASE	3244.126	3936.141	1891.773	2556.906	1227.883
TWO-NORM	2662.716	2957.706	1926.331	2336.788	1648.394
RMSE	15.495	17.211	11.210	13.598	9.592
ONE-NORM	105,309.169	129,941.886	58,980.290	81,207.075	37,557.957
INFINITY-NORM	479.250	390.250	368.000	512.250	394.250
MAE	3.566	4.400	1.997	2.750	1.272
Dataset 2					
INFINITY-NORM	55.000	66.500	57.250	53.000	54.000
SMAPE	0.023	0.030	0.014	0.019	0.009
MASE	6.597	9.129	3.495	4.607	2.576
RMSE	9.634	12.423	6.865	8.349	6.661
ONE-NORM	413.000	568.750	223.000	292.500	161.500
TWO-NORM	122.619	158.125	87.377	106.271	84.780
MAE	2.549	3.511	1.377	1.806	0.997
MEP	2.006	2.623	1.235	1.698	0.772
Dataset 3					
MEP	2.193	2.591	1.240	1.716	0.747
TWO-NORM	585.025	716.476	459.022	556.095	358.345
MASE	112.801	153.848	66.868	96.474	41.447
ONE-NORM	6102.814	8102.064	3541.500	5143.250	2152.750
RMSE	14.751	18.065	11.574	14.021	9.035
MAE	3.880	5.151	2.251	3.270	1.369
INFINITY-NORM	92.250	100.000	98.000	91.250	110.250
SMAPE	0.025	0.030	0.014	0.020	0.009
Dataset 4					
SMAPE	0.024	0.030	0.014	0.020	0.009
MASE	37.607	43.710	21.436	28.546	13.447
MAE	17.258	20.265	9.781	13.293	6.328
TWO-NORM	2295.560	2424.160	1722.663	2026.235	1360.999
MEP	2.117	2.616	1.223	1.746	0.754
INFINITY-NORM	399.000	389.000	399.000	397.000	385.000
ONE-NORM	19,502.000	22,899.000	11,053.000	15,021.000	7151.000
RMSE	68.289	72.114	51.246	60.277	40.487

Table 4. Algorithmic comparison of the proposed AQP framework with four datasets.

Metrics/classifier	LSTM[34]	DNN[33]	TCN[28]	CNN[32]	FEO-PFA-MDS-ATCN
Dataset 1					
MASE	2922.077	3557.102	2252.857	4195.820	1544.217
SMAPE	0.022	0.027	0.017	0.032	0.011
ONE-NORM	94,064.620	116,451.928	70,872.226	138,957.522	47,489.470
MAE	3.185	3.943	2.400	4.705	1.608
RMSE	14.695	15.868	12.720	17.923	10.517
INFINITY-NORM	460.500	355.750	479.250	436.750	411.500
TWO-NORM	2525.343	2726.853	2185.904	3079.980	1807.277
MEP	1.923	2.375	1.458	2.806	0.982
Dataset 2					
MEP	2.006	2.160	1.389	2.932	0.772
INFINITY-NORM	63.250	47.000	64.750	59.500	38.250
MAE	2.148	2.642	1.681	3.620	0.664
TWO-NORM	117.545	120.907	104.248	150.974	52.037
ONE-NORM	348.000	428.000	272.250	586.500	107.500
MASE	5.520	6.809	4.373	9.351	1.705
RMSE	9.235	9.499	8.191	11.862	4.088
SMAPE	0.023	0.025	0.016	0.034	0.009
Dataset 3					
SMAPE	0.023	0.027	0.016	0.033	0.011
RMSE	14.544	15.906	12.796	18.211	10.960
MAE	3.661	4.281	2.684	5.436	1.927
MEP	1.987	2.352	1.414	2.861	1.001
MASE	108.963	125.229	79.095	159.972	57.179
TWO-NORM	576.814	630.831	507.485	722.278	434.682
INFINITY-NORM	95.250	104.500	93.750	99.000	99.000
ONE-NORM	5758.564	6733.750	4222.250	8551.000	3031.750
Dataset 4					
RMSE	64.525	72.708	54.782	76.367	48.716
MASE	34.964	42.352	25.107	48.251	17.183
MAE	16.112	19.439	11.763	22.064	8.294
SMAPE	0.013	0.029	0.017	0.027	0.008
MEP	2.857	2.148	1.487	2.868	0.968
INFINITY-NORM	389.000	399.000	384.000	400.000	397.000
TWO-NORM	2169.034	2444.122	1841.512	2567.121	1637.598
ONE-NORM	18,207.000	21,966.000	13,292.000	24,932.000	9372.000

Table 5. Comparison of the developed AQP model with other classifiers.

Data availability

The datasets used and/or analyzed during the current study are available from the first author on request. rajanand@cse.sastra.ac.in.

Received: 2 December 2023; Accepted: 29 July 2024
Published online: 08 August 2024

References

1. Zhao, G., Huang, G., He, H. & Wang, Q. Innovative spatial-temporal network modeling and analysis method of air quality. *IEEE Access* **7**, 26241–26254 (2019).
2. Gu, K., Qiao, J. & Lin, W. Recurrent air quality predictor based on meteorology-and pollution-related factors. *IEEE Trans. Ind. Inform.* **14**, 3946–3955 (2018).
3. Yi, X. *et al.* Predicting fine-grained air quality based on deep neural networks. *IEEE Trans. Big Data* **8**, 1326–1339 (2020).
4. Ha, Q. P., Metia, S. & Phung, M. D. Sensing data fusion for enhanced indoor air quality monitoring. *IEEE Sens. J.* **20**, 4430–4441 (2020).
5. Zhang, D. & Woo, S. S. Real time localized air quality monitoring and prediction through mobile and fixed IoT sensing network. *IEEE Access* **8**, 89584–89594 (2020).
6. El Fazziki, A., Benslimane, D., Sadiq, A., Ouarzazi, J. & Sadgal, M. An agent based traffic regulation system for the roadside air quality control. *IEEE Access* **5**, 13192–13201 (2017).
7. Metia, S., Oduro, S. D., Duc, H. N. & Ha, Q. Inverse air-pollutant emission and prediction using extended fractional Kalman filtering. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **9**, 2051–2063 (2016).
8. Zhao, G., Huang, G., He, H., He, H. & Ren, J. Regional spatiotemporal collaborative prediction model for air quality. *IEEE Access* **7**, 134903–134919 (2019).
9. Yu, Y., James, J., Li, V. O. & Lam, J. C. A novel interpolation-SVT approach for recovering missing low-rank air quality data. *IEEE Access* **8**, 74291–74305 (2020).
10. Mokhtari, I., Bechkit, W., Rivano, H. & Yaici, M. R. Uncertainty-aware deep learning architectures for highly dynamic air quality prediction. *IEEE Access* **9**, 14765–14778 (2021).
11. Ando, B., Baglio, S., Graziani, S. & Pitrone, N. Models for air quality management and assessment. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **30**, 358–363 (2000).
12. Chen, J. *et al.* An adaptive Kalman filtering approach to sensing and predicting air quality index values. *IEEE Access* **8**, 4265–4272 (2020).
13. Chhikara, P., Tekchandani, R., Kumar, N., Guizani, M. & Hassan, M. M. Federated learning and autonomous UAVs for hazardous zone detection and AQI prediction in IoT environment. *IEEE Internet Things J.* **8**, 15456–15467 (2021).
14. Wang, Y. & Kong, T. Air quality predictive modeling based on an improved decision tree in a weather-smart grid. *IEEE Access* **7**, 172892–172901 (2019).
15. Qi, Z. *et al.* Deep air learning: Interpolation, prediction, and feature analysis of fine-grained air quality. *IEEE Trans. Knowl. Data Eng.* **30**, 2285–2297 (2018).
16. Yang, Y., Zheng, Z., Bian, K., Song, L. & Han, Z. Real-time profiling of fine-grained air quality index distribution using UAV sensing. *IEEE Internet Things J.* **5**, 186–198 (2017).
17. Zhang, Y. *et al.* A predictive data feature exploration-based air quality prediction approach. *IEEE Access* **7**, 30732–30743 (2019).
18. Liu, B. *et al.* A sequence-to-sequence air quality predictor based on the n-step recurrent prediction. *IEEE Access* **7**, 43331–43345 (2019).
19. Yang, Y., Mei, G. & Izzo, S. Revealing influence of meteorological conditions on air quality prediction using explainable deep learning. *IEEE Access* **10**, 50755–50773 (2022).
20. Chen, H., Guan, M. & Li, H. Air quality prediction based on integrated dual LSTM model. *IEEE Access* **9**, 93285–93297 (2021).
21. Huang, Y., Xiang, Y., Zhao, R. & Cheng, Z. Air quality prediction using improved PSO-BP neural network. *IEEE Access* **8**, 99346–99353 (2020).
22. Soh, P.-W., Chang, J.-W. & Huang, J.-W. Adaptive deep learning-based air quality prediction model using the most relevant spatial-temporal relations. *IEEE Access* **6**, 38186–38199 (2018).
23. Sridhar, K. *et al.* A modular IOT sensing platform using hybrid learning ability for air quality prediction. *Meas. Sens.* **25**, 100609 (2023).
24. Liu, C.-C., Lin, T.-C., Yuan, K.-Y. & Chiueh, P.-T. Spatio-temporal prediction and factor identification of urban air quality using support vector machine. *Urban Clim.* **41**, 101055 (2022).
25. Xu, X. & Yoneda, M. Multitask air-quality prediction based on LSTM-autoencoder model. *IEEE Trans. Cybern.* **51**, 2577–2586 (2019).
26. Salim, A., Jummar, W. K., Jasim, F. M. & Yousif, M. Eurasian oystercatcher optimiser: New meta-heuristic algorithm. *J. Intell. Syst.* **31**, 332–344 (2022).
27. Yapici, H. & Cetinkaya, N. A new meta-heuristic optimizer: Pathfinder algorithm. *Appl. Soft Comput.* **78**, 545–568 (2019).
28. Samal, K. K. R., Babu, K. S. & Das, S. K. Temporal convolutional denoising autoencoder network for air pollution prediction with missing values. *Urban Clim.* **38**, 100872 (2021).
29. Kollias, D. & Zafeiriou, S. Exploiting multi-CNN features in CNN-RNN based dimensional emotion recognition on the omg in-the-wild dataset. *IEEE Trans. Affect. Comput.* **12**, 595–606 (2020).
30. Kwon, H., Pellauer, M., Parashar, A. & Krishna, T. Flexion: A quantitative metric for flexibility in DNN accelerators. *IEEE Comput. Archit. Lett.* **20**, 1–4 (2020).
31. Shu, X., Zhang, L., Sun, Y. & Tang, J. Host–parasite: Graph LSTM-in-LSTM for group activity recognition. *IEEE Trans. Neural Netw. Learn. Syst.* **32**, 663–674 (2020).
32. Elgebaly, A. E., Taha, I. B., Azmy, A. M. & Abd El-Ghany, H. A. Optimal design and control of SSSCs for TLs considering technical and economic indices using GA and SAMPE-JAYA algorithms. *IEEE Access* **9**, 38907–38919 (2021).
33. Yuan, Z., Wang, W., Wang, H. & Ashourian, M. Parameter identification of PEMFC based on convolutional neural network optimized by balanced deer hunting optimization algorithm. *Energy Rep.* **6**, 1572–1580 (2020).
34. Mirjalili, S. *et al.* Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. *Adv. Eng. Softw.* **114**, 163–191 (2017).

Author contributions

Conceptualization, Koteeswaran Seerangan; Data curation, Raj Anand Sundaramoorthy and Antony Ananth; Formal analysis, Malarvizhi Nandagopal; Funding acquisition, Shitharth Selvarajan; Investigation, Shitharth Selvarajan; Methodology, Koteeswaran Seerangan and Malarvizhi Nandagopal; Project administration, Balamurugan Balusamy and Shitharth Selvarajan; Resources, Antony Ananth, Malarvizhi Nandagopal, Shitharth

Selvarajan; Software, Raj Anand Sundaramoorthy, Antony Ananth and Balamurugan Balusamy; Supervision, Balamurugan Balusamy; Validation, Raj Anand Sundaramoorthy, Antony Ananth and Balamurugan Balusamy; Visualization, Antony Ananth and Shitharth Selvarajan; Writing—original draft, Raj Anand Sundaramoorthy and Malarvizhi Nandagopal; Writing—review and editing, Koteeswaran Seerangan, Balamurugan Balusamy, Shitharth Selvarajan. All authors have read and agreed to the published version of the manuscript. Shitharth Selvarajan.

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to S.S.

Reprints and permissions information is available at www.nature.com/reprints.

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