

Air Quality Prediction Using LOESS–ARIMA and Multi-Scale CNN–BiLSTM with Residual-Gated Attention

Soham Pahari^{a,*}, Sandeep Chand Kumain^a

^a*School of Computer Science, University of Petroleum and Energy Studies (UPES), India*

Abstract

Air pollution remains a critical environmental and public health concern in Indian megacities such as Delhi, Kolkata, and Mumbai, where sudden spikes in pollutant levels challenge timely intervention. Accurate Air Quality Index (AQI) forecasting is difficult due to the coexistence of linear trends, seasonal variations, and volatile nonlinear patterns. This paper proposes a hybrid forecasting framework that integrates LOESS decomposition, ARIMA modeling, and a multi-scale CNN–BiLSTM network with a residual-gated attention mechanism. The LOESS step separates the AQI series into trend, seasonal, and residual components, with ARIMA modeling the smooth components and the proposed deep learning module capturing multi-scale volatility in the residuals. Model hyperparameters are tuned via the Unified Adaptive Multi-Stage Metaheuristic Optimizer (UAMMO), combining multiple optimization strategies for efficient convergence. Experiments on 2021–2023 AQI datasets from the Central Pollution Control Board show that the proposed method consistently outperforms statistical, deep learning, and hybrid baselines across PM_{2.5}, O₃, CO, and NO_x in three major cities, achieving up to 5–8% lower MSE and higher R^2 scores (> 0.94) for all pollutants. These results demonstrate the framework's robustness, sensitivity to sudden pollution events, and applicability to urban air quality management.

Keywords: Air Quality Forecasting, Hybrid Deep Learning, Residual-Gated Attention, Metaheuristic Optimization

1. Introduction

Air pollution has emerged as one of the most pressing environmental and public health challenges in developing nations, with India being no exception[1]. Major metropolitan cities such as Delhi, Kolkata, and Mumbai frequently rank among the most polluted cities in the world [2]. Rapid industrialization, explosive population growth, and accelerated urbanization, combined with unregulated vehicular emissions, construction dust, and industrial discharge, have aggravated air quality to alarming levels. [3]

The Air Quality Index (AQI) provides a standardized measure for evaluating air pollution severity, with four primary pollutants playing a dominant role: particulate matter (PM_{2.5}), nitrogen oxides (NO_x), carbon monoxide (CO), and ground-level ozone (O₃). Prolonged exposure to these pollutants has been strongly associated with severe respiratory illnesses, including asthma, bronchitis, and emphysema, as well as cardiovascular, neurological, and other chronic health complications[4].

Accurately forecasting air quality is a highly complex task due to the interplay of numerous linear and nonlinear factors, seasonal variations, and sudden anomalies in pollutant levels. Traditional statistical models, such as ARIMA, capture linear temporal dependencies effectively but struggle with highly nonlinear residual patterns. Conversely, deep learning models like LSTM[5] and CNN have demonstrated strong capabilities in modeling complex temporal dependencies but often fail to leverage the structured nature of decomposable time-series components. Many existing hybrid approaches[6] fall short because they either:

*Corresponding author

Email addresses: soham.109424@stu.upes.ac.in, paharisoham@gmail.com (Soham Pahari), sandeep.kumain@ddn.upes.ac.in, sandeep.chandphdrw@gmail.com (Sandeep Chand Kumain)

1. Treat the residual component as a black box without explicitly modeling its volatility and multi-scale characteristics, or
2. Lack a robust mechanism for optimal hyperparameter selection, leading to suboptimal generalization performance.

To address these shortcomings, this paper introduces a multi-stage hybrid forecasting framework that integrates LOESS-based time series decomposition, ARIMA modeling for smooth components, and a multi-scale CNN-BiLSTM network with a residual-gated attention mechanism for volatile residuals. Furthermore, the framework incorporates UAMMO, a unified adaptive metaheuristic optimization algorithm, to fine-tune hyperparameters for optimal predictive performance.

The key contributions of this work are as follows:

- Component-wise modeling: Leveraging LOESS decomposition to assign ARIMA for trend and seasonal components, and a deep residual network for noisy components.
- Residual-gated attention mechanism: Introducing volatility-aware attention to better capture sudden pollutant spikes and irregularities.
- Multi-scale feature extraction: Using multiple convolutional branches to capture both short-term bursts and long-term deviations in residual patterns.
- Metaheuristic-driven optimization: Employing UAMMO to efficiently search for optimal hyperparameters across CNN, BiLSTM, and attention layers.

The remainder of this paper is organized as follows: Section 2 reviews related works, Section 3 details the proposed methodology, Section 4 presents experimental results, and Section 5 concludes with insights and potential directions for future research.

2. Related Work

Approaches for predicting the Air Quality Index (AQI) can generally be grouped into four categories: (i) statistical models, (ii) machine learning (ML) methods, (iii) deep learning (DL) methods, and (iv) hybrid approaches. Statistical models depend on assumptions regarding the underlying data distribution, aiming to identify causal relationships and estimate unknown parameters. In AQI prediction, these methods often include autoregressive (AR), autoregressive integrated moving average (ARIMA), gray models, and multiple linear regression (MLR). For example, Carbajal-Hernández et al. [7] designed an algorithm that evaluates pollution levels through a fuzzy reasoning system to develop a novel air quality index. Zhang et al. [8] compared generalized additive models (GAMs) against conventional linear regression techniques. Zhao et al. [9] applied an ARIMA model to PM_{2.5} annual data, confirming via the augmented Dickey-Fuller test that first-order differencing was necessary. A seasonally nonlinear gray model was also proposed to better account for periodic and nonlinear patterns in fluctuating pollution data [10].

Machine learning methods have improved predictive accuracy by leveraging large datasets and overcoming convergence issues. Mehmood et al. [11] discussed the transition from conventional approaches to ML-based frameworks. Through trend analysis, ML techniques can also highlight promising research avenues. Usha et al. [12] used Neural Networks and Support Vector Machines (SVMs) to improve prediction accuracy and suggested their adaptability for other smart city environments. In another example, Elsheikh [13] explored ML applications in friction stir welding for predicting joint quality and failure diagnostics. Ke et al. [14] developed an ML-based system that forecasts daily pollutant concentrations using meteorological data, emission levels, and reanalysis datasets, employing an ensemble of five ML models with automated model and hyperparameter selection. Zhang et al. [15] investigated ML techniques to predict indoor air quality variations caused by unpredictable factors. Gu et al. [16] proposed an interpretable hybrid ML model for PM_{2.5} prediction that excelled in accuracy, particularly for peak values. More recently, Rakholia et al. [17] incorporated meteorology, traffic, industrial and residential pollution levels, spatial data, and time series analysis into a unified air quality prediction framework.

With the advancements in DL, these models have become highly effective in AQI prediction due to their ability to process large datasets and capture complex relationships. Janarthanan et al. [18] combined Support Vector Regression

(SVR) with LSTM networks to accurately predict AQI, enabling policy decisions such as traffic management and tree planting. Zhang et al. [19] examined DL models from temporal, spatial, and spatio-temporal perspectives. Saez et al. [20] developed a hierarchical Bayesian spatiotemporal model to achieve accurate predictions even with sparse monitoring stations. Jurado et al. [21] utilized convolutional neural networks (CNNs) for fast, precise forecasts using real-time wind, traffic, and urban geometry data. Zhou et al. [22] combined CNNs with Gated Recurrent Units (GRUs) to capture both spatial features and temporal dependencies. Mao et al. [23] extended LSTM with a temporal sliding approach to predict 24-hour air quality using historical PM2.5, meteorological, and temporal data. Elsheikh et al. [24] used LSTM networks in predicting freshwater production in a novel solar still design. Similarly, Djouider et al. [25] integrated LSTM with optimization algorithms for modeling material processing effects.

Hybrid methods combine the strengths of different models for improved performance. Wu and Lin [26] created an optimal hybrid AQI prediction framework (SD-SE-LSTM-BA-LSSVM) combining decomposition, AI methods, and optimization techniques. Sarkar et al. [27] integrated LSTM and GRU models, achieving lower MAE and higher R^2 than existing methods. Gilik et al. [28] merged CNN and LSTM to forecast pollutant levels city-wide in both univariate and multivariate formats. Zhu et al. [29] enhanced accuracy by introducing EMD-SVR-Hybrid and EMD-IMFs-Hybrid models. Chang et al. [30] combined stacking-based ensemble learning with Pearson correlation analysis to merge multiple forecasting models. Wang et al. [31] incorporated attention mechanisms into LSTM to boost prediction accuracy. Elsheikh et al. [32] applied LSTM to COVID-19 case forecasting, while Dai et al. [33] employed an improved particle swarm optimization (IPSO) with LightGBM for haze risk assessment. Saba and Elsheikh [34] combined nonlinear autoregressive neural networks (NARANN) with ARIMA for COVID-19 outbreak forecasting.

3. Proposed Methodology

Forecasting air quality with high precision demands a modeling framework capable of capturing both the structured and irregular patterns inherent in environmental time series. While statistical models like ARIMA are effective at modeling linear and seasonal signals, they lack the flexibility to handle non-linear and chaotic behaviors. Conversely, deep learning models such as LSTM and its variants excel at learning complex temporal dependencies but struggle with explicit long-term patterns unless properly guided. To reconcile these strengths and limitations, we propose a decomposition-based hybrid forecasting model where structured components are handled by statistical modeling, and irregular residuals are learned through a deep neural architecture. This approach is further optimized using a novel metaheuristic framework, the Unified Adaptive Multi-Stage Metaheuristic Optimizer (UAMMO), which fine-tunes model hyperparameters for optimal forecasting performance. Figure 1 describe pipeline flow.

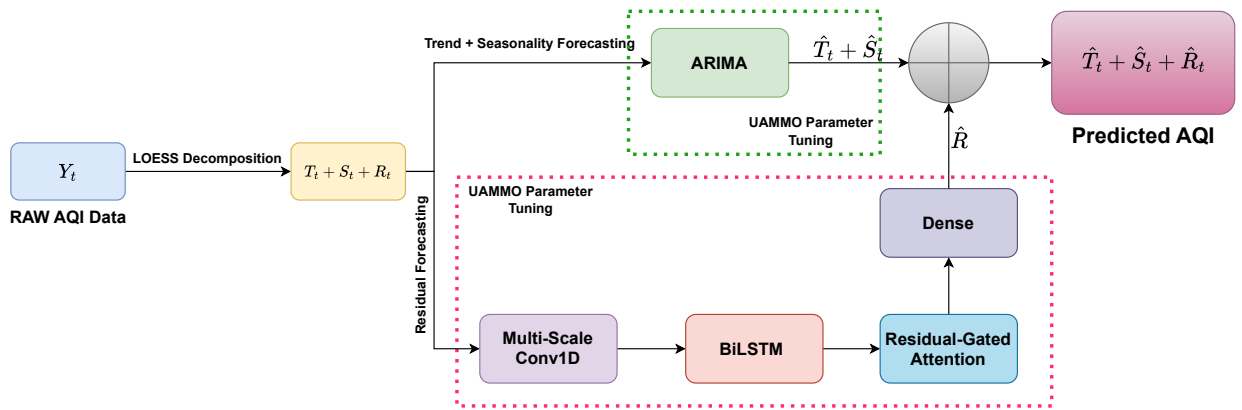


Figure 1: Proposed Pipeline

The forecasting pipeline begins with the decomposition of the raw AQI time series using LOESS smoothing, which separates the input Y_t into three interpretable components: a trend T_t , a seasonal component S_t , and a residual

term R_t . This decomposition enables the model to assign different forecasting strategies to each component based on its statistical characteristics.

$$Y_t = T_t + S_t + R_t \quad (1)$$

The trend and seasonal components, which exhibit relatively smooth and linear behavior, are predicted using the ARIMA model. ARIMA captures auto-correlations and seasonality effectively by fitting time series patterns based on lagged observations, differencing, and moving averages. Delegating these components to ARIMA not only leverages its strength in modeling structured signals but also simplifies the learning task for the deep network by reducing the variance and noise in the input data.

3.1. Multi-Scale CNN–BiLSTM with Residual-Gated Attention

While standard deep learning architectures are capable of modeling nonlinear time series, they often underperform when applied to residual signals extracted from decomposed air quality data. This is because residuals are inherently noisy, nonstationary, and composed of features at multiple temporal scales — such as sudden pollution bursts and sustained deviations from the mean. To effectively learn from such irregularities, we propose a multi-resolution residual learning architecture (shown in figure 2) that combines multi-scale convolutional encoding, bidirectional sequence modeling, and a novel residual-gated attention mechanism.

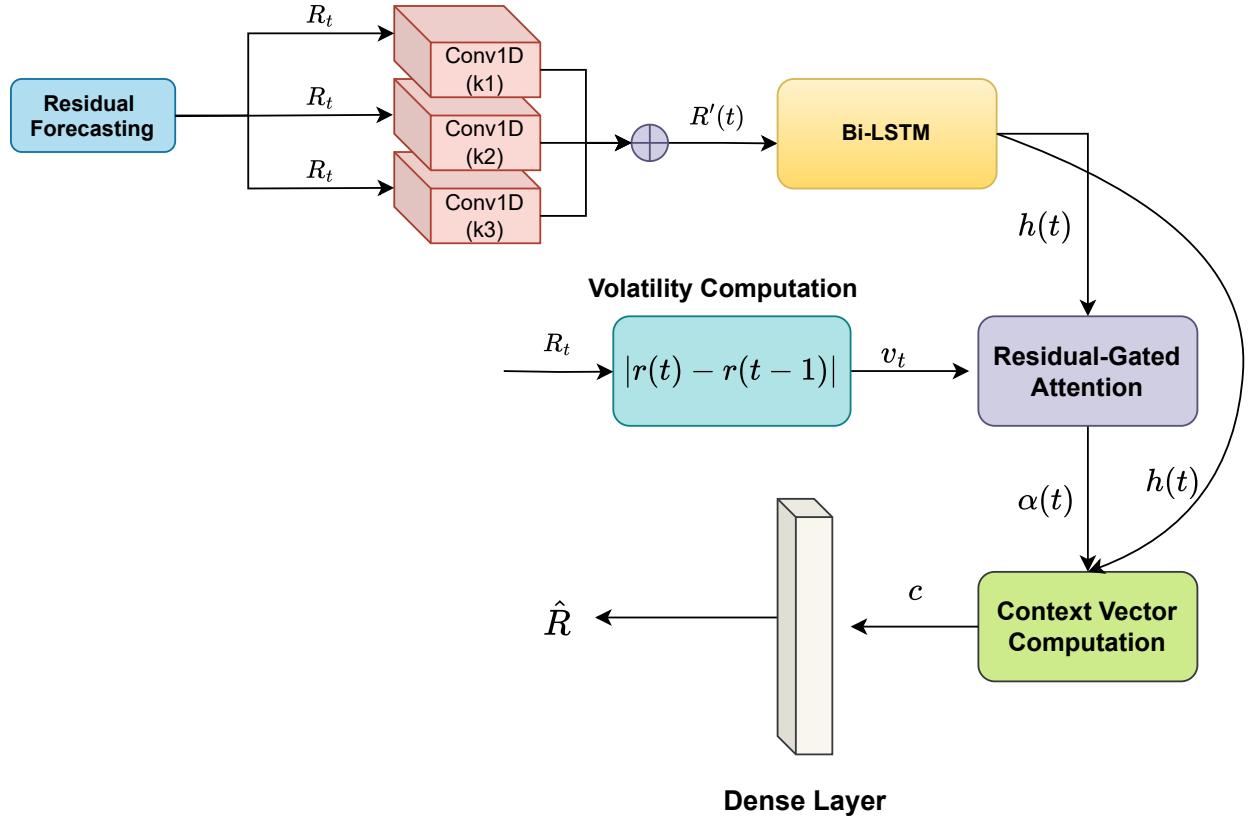


Figure 2: Residual Forecasting Network Architecture

Let

$$R = \{r_1, r_2, \dots, r_T\} \quad (2)$$

represent the residual component of the decomposed AQI series. The first layer of the network is a multi-branch 1D convolutional encoder, where each branch uses a different kernel size to capture short- and long-range features:

$$R_t^{(s)} = \text{Conv1D}_{k_s}(R), \quad s = 1, 2, \dots, S \quad (3)$$

Each kernel k_s captures temporal dependencies at scale s . The resulting outputs $R_t^{(1)}, R_t^{(2)}, \dots, R_t^{(S)}$ are concatenated to form a composite representation:

$$R'_t = \text{Concat}(R_t^{(1)}, R_t^{(2)}, \dots, R_t^{(S)}) \quad (4)$$

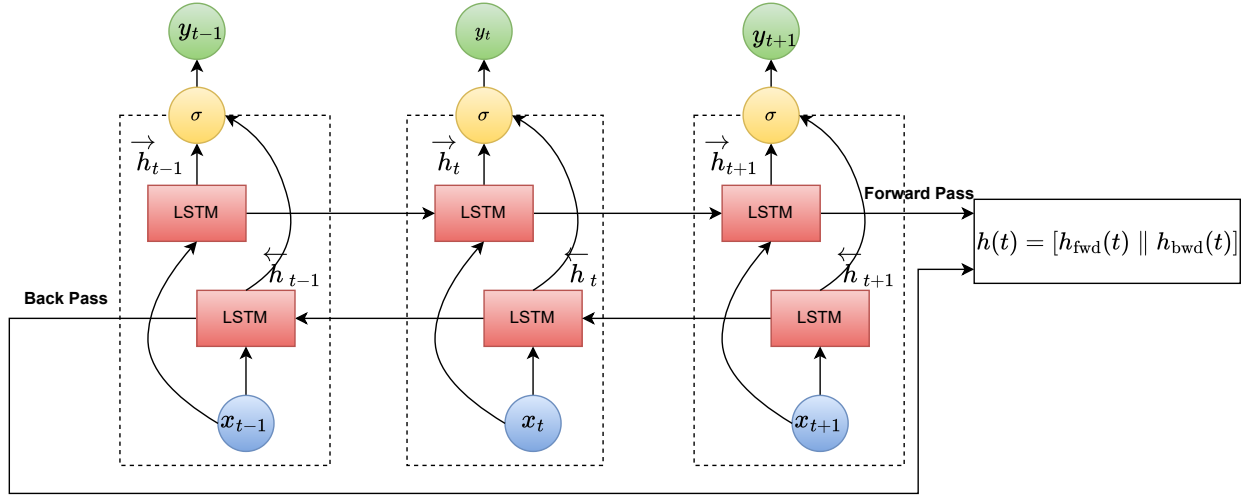


Figure 3: Bidirectional LSTM

This multi-scale representation is then passed to a Bidirectional LSTM (BiLSTM) layer (described in figure 3), which learns both forward and backward temporal dependencies:

$$\vec{h}_t = \text{LSTM}_{\text{fwd}}(R'_t), \quad \overleftarrow{h}_t = \text{LSTM}_{\text{bwd}}(R'_t) \quad (5)$$

$$h_t = \vec{h}_t \parallel \overleftarrow{h}_t \quad (6)$$

Instead of applying standard attention, we introduce a residual-gated attention mechanism, where the attention score at each time step is modulated by a learned function of local residual volatility. Let:

$$v_t = |r_t - r_{t-1}| \quad (7)$$

denote the local volatility at time t . The attention mechanism uses both the hidden state h_t and the volatility signal v_t to compute a gated attention weight:

$$\alpha_t = \frac{\exp(w^\top \tanh(W_h h_t + W_v v_t + b))}{\sum_{i=1}^T \exp(w^\top \tanh(W_h h_i + W_v v_i + b))} \quad (8)$$

Here, W_h , W_v , b , and w are learnable parameters. This formulation prioritizes time steps that are both informative in the LSTM latent space and volatile in the residual space — ideal for detecting spikes or structural deviations in pollution behavior.

The final residual forecast $\hat{R}_t$ is obtained by applying the attention weights to the BiLSTM sequence, followed by a fully connected layer:

$$c = \sum_{t=1}^T \alpha_t h_t \quad (9)$$

$$\hat{R}_t = \text{Dense}(c) \quad (10)$$

This multi-scale, volatility-aware design provides both an interpretable and precision. It is particularly novel in that it models the residual component as a multi-scale, volatility-driven signal, which is a significant departure from typical black-box approaches that model residuals using standard LSTM alone. By targeting the structural characteristics of residuals, the network achieves better generalization and improved sensitivity to sudden environmental changes.

Finally, the complete AQI forecast is reconstructed by summing the individual predictions of the decomposed components:

$$\hat{Y}_t = \hat{T}_t + \hat{S}_t + \hat{R}_t \quad (11)$$

3.2. Hyperparameter Optimization with UAMMO

The performance of the proposed model depends heavily on the choice of hyperparameters such as the number of LSTM layers n_l , hidden units n_u , learning rate η , batch size b , number of CNN filters f , and convolutional kernel sizes k_s . To find the optimal set θ^* , we minimize the validation loss:

$$J(\theta) = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i(\theta))^2 \quad (12)$$

We employ UAMMO, which integrates five metaheuristics — Dung Beetle Optimizer (DBO), Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Gravitational Search Algorithm (GSA), and Red Deer Algorithm (RDA) — into a unified adaptive search framework. Let $\vec{x}_i^t$ be the position of the i -th candidate at iteration t and $\vec{v}_i^t$ its velocity. The unified velocity update rule is:

$$\vec{v}_i^{t+1} = \omega \cdot \vec{v}_i^t + \sum_{j=1}^5 \alpha_j(t) \cdot \Phi_j^t(\vec{x}_i^t) \quad (13)$$

$$\vec{x}_i^{t+1} = \vec{x}_i^t + \vec{v}_i^{t+1} \quad (14)$$

Here, Φ_j^t denotes the update from the j -th metaheuristic, and $\alpha_j(t)$ is an adaptive weight that decays over time:

$$\alpha_j(t) = \alpha_j^{\max} \cdot \left(1 - \frac{t}{T}\right) \quad (15)$$

The optimization stops when the relative improvement in the best solution is below a threshold ϵ :

$$\left| \frac{J_{\text{best}}^t - J_{\text{best}}^{t-1}}{J_{\text{best}}^{t-1}} \right| < \epsilon \quad (16)$$

This ensures both computational efficiency and convergence to high-quality solutions.

Algorithm 1: UAMMO Hyperparameter Optimization for Multi-Scale CNN–BiLSTM with Residual-Gated Attention

Input: Population size N , maximum iterations T , search space bounds for θ , decay parameters $\alpha_j^{\max}$, inertia weight ω

Output: Optimal hyperparameters θ^*

```
1 Initialization: Randomly initialize positions  $\vec{x}_i^0$  and velocities  $\vec{v}_i^0$  for  $i = 1, \dots, N$ ;  
2 Evaluate fitness  $J(\vec{x}_i^0)$  for all candidates using Eq. (12);  
3 Store the best candidate  $\theta_{\text{best}}$ ;  
4 for  $t = 1$  to  $T$  do  
5   for  $i = 1$  to  $N$  do  
6     // Apply updates from each metaheuristic  
7     for  $j = 1$  to 5 do  
8       Compute  $\Phi_j^t(\vec{x}_i^t)$  according to the  $j$ -th algorithm (DBO, PSO, GA, GSA, RDA);  
9       Compute adaptive weight  $\alpha_j(t) = \alpha_j^{\max} \cdot \left(1 - \frac{t}{T}\right)$ ;  
10      // Unified velocity and position update  
11       $\vec{v}_i^{t+1} = \omega \cdot \vec{v}_i^t + \sum_{j=1}^5 \alpha_j(t) \cdot \Phi_j^t(\vec{x}_i^t)$ ;  
12       $\vec{x}_i^{t+1} = \vec{x}_i^t + \vec{v}_i^{t+1}$ ;  
13      Evaluate fitness  $J(\vec{x}_i^{t+1})$  for all candidates;  
14      Update  $\theta_{\text{best}}$  if a better candidate is found;  
15      if  $\left| \frac{J_{\text{best}}^t - J_{\text{best}}^{t-1}}{J_{\text{best}}^{t-1}} \right| < \epsilon$  then  
16        break;  
17 return  $\theta_{\text{best}}$ ;
```

4. Experiment Setup and Results

In this section, we have discussed the geographical areas from which our pollution data were collected, evaluated the performance of the proposed model, and provided its qualitative assessment along with relevant observations and interpretations.

4.1. Area of Study

To evaluate and validate the performance of the proposed forecasting model, we utilized air quality data collected from multiple cities across India. The experimental analysis was conducted using measurements from New Delhi 4, Mumbai 5, and Kolkata 6, covering the period from 2021 to 2023. For each city, data were collected from multiple monitoring stations distributed across different locations, and included key pollutants such as $\text{PM}_{2.5}$, O_3 , CO , and NO_x , along with the air quality index (AQI). The dataset was sourced from the Central Pollution Control Board (CPCB), the official environmental monitoring authority under the Government of India, which makes air quality data publicly available through its official portal [35].

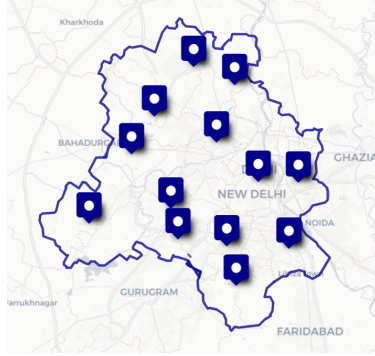


Figure 4: Field Map Of Delhi with Station Location [36]

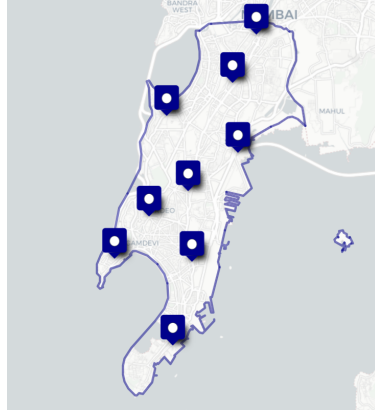


Figure 5: Field Map Of Mumbai with Station Location [36]

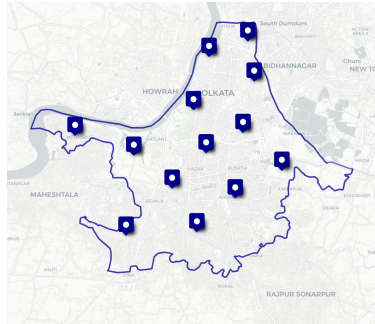


Figure 6: Field Map Of Kolkata with Station Location [36]

4.2. Data Processing

Data preprocessing was a crucial phase in this time series analysis, ensuring the raw data was transformed into a clean and structured format suitable for machine learning. The initial step involved aggregating multiple fragmented datasets, each pertaining to a different city within a state, into a single cohesive DataFrame. This allowed for a holistic view of the region's air quality. Following this, the raw datetime information, provided as separate 'From Date' and 'To Date' columns, was unified. The 'From Date' was parsed into a datetime object and designated as the DataFrame's index, a fundamental requirement for time series analysis. This provided a chronological backbone to the dataset. A significant challenge was the presence of redundant or similar features, such as 'Xylene (ug/m3)' and

‘Xylene ()’. These were systematically merged to create a single, consistent feature representation. Columns that contained an excessive percentage of missing values, defined by a threshold $T = 0.6$, were then dropped, where T represents the minimum acceptable data presence. For the remaining missing data points, a forward-fill interpolation method was applied, propagating the last known value forward, followed by mean imputation for any residual nulls. To augment the dataset’s predictive power, new temporal features were extracted directly from the datetime index, including ‘hour’, ‘dayofweek’, ‘month’, and ‘year’. These features capture underlying cyclical patterns. Furthermore, historical context was introduced through the creation of lag features for the target variable, $PM_{2.5}$. Specifically, the value of $PM_{2.5}$ from one year ago ($PM_{2.5}(t - 365 \times 24)$) was included as a new feature, a powerful technique for modeling seasonality. Finally, outlier detection and removal were performed on key pollutant metrics to mitigate their distorting effect on model training, leading to a more stable and accurate forecasting model.

4.3. Exploratory Data Analysis

In this study, exploratory data analysis (EDA) was performed to examine the characteristics and spatiotemporal patterns of air quality data collected from Mumbai, Kolkata, and Delhi between January 2021 and January 2023. Figures 7 to 13 illustrate the overall trends in $PM_{2.5}$, O_3 (ozone), NO_x , NH_3 , and CO over the study period. A noticeable reduction in pollutant concentrations was observed during certain months, which may be linked to seasonal effects, local emission control measures, and intermittent COVID-19–related restrictions in parts of the cities. Despite generally moderate pollution levels, the plots reveal that all three cities occasionally experienced episodes of poor or very poor air quality, with elevated concentrations of one or more pollutants.

It is well established that road traffic and industrial activities are major contributors to urban air pollution, frequently emitting multiple pollutants simultaneously. For example, motor vehicles typically emit both CO and NO_x , while agricultural activities can be a source of NH_3 .

Due to the limited forecasting benefit observed with monthly-aggregated data, the analysis incorporated hourly pollutant data as input features for model development. This higher-resolution data improved the model’s ability to capture short-term variations and relationships between pollutants across the three cities.

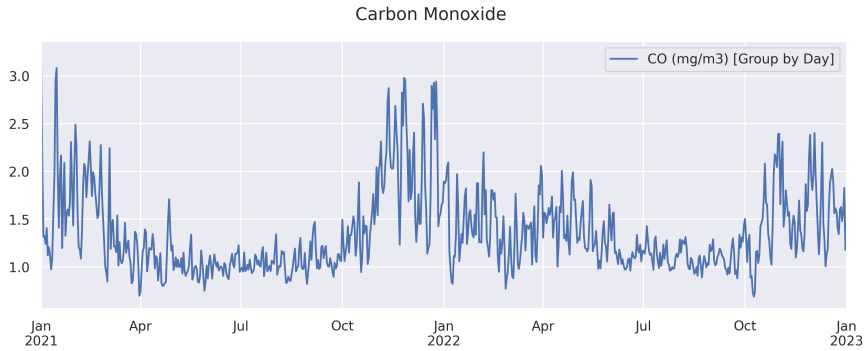


Figure 7: Carbon Monoxide concentration trends in Delhi.

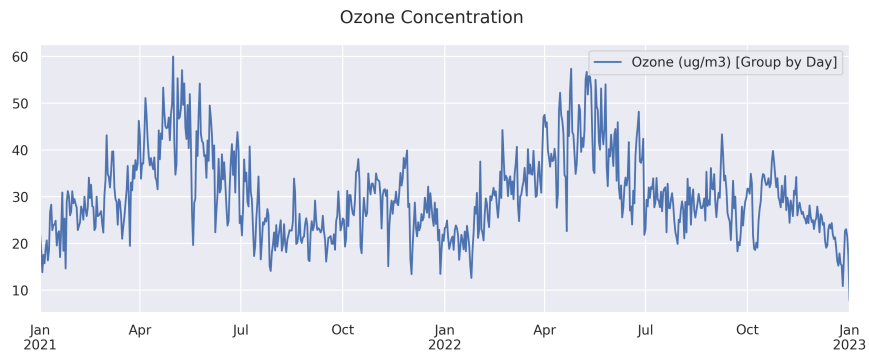


Figure 8: Ozone concentration trends in Delhi.

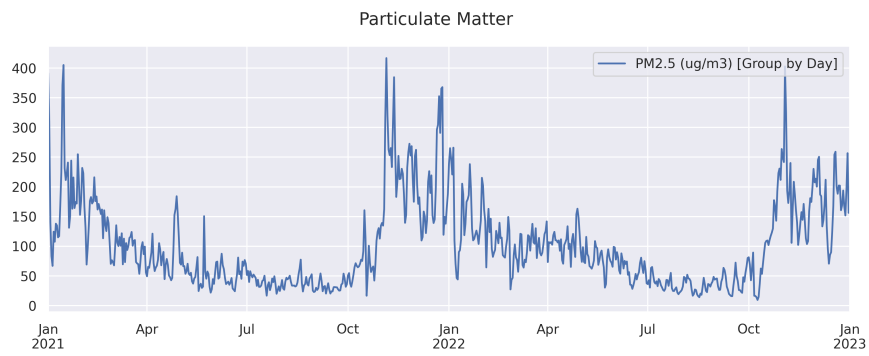


Figure 9: Particulate Matter (PM_{2.5}) concentration trends in Delhi.

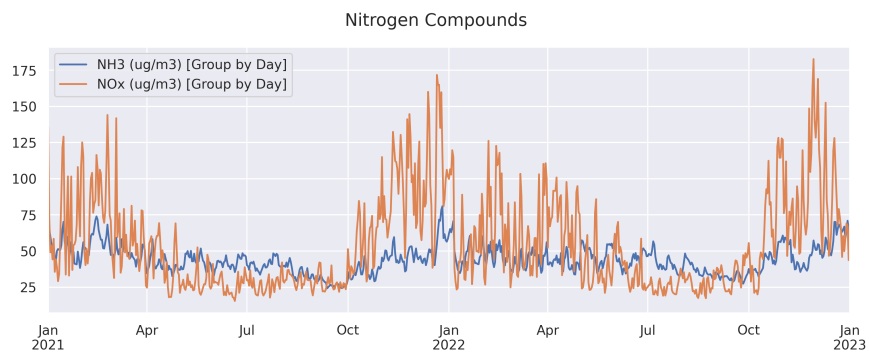


Figure 10: Nitrogen Compounds (NO_x) concentration trends in Delhi.

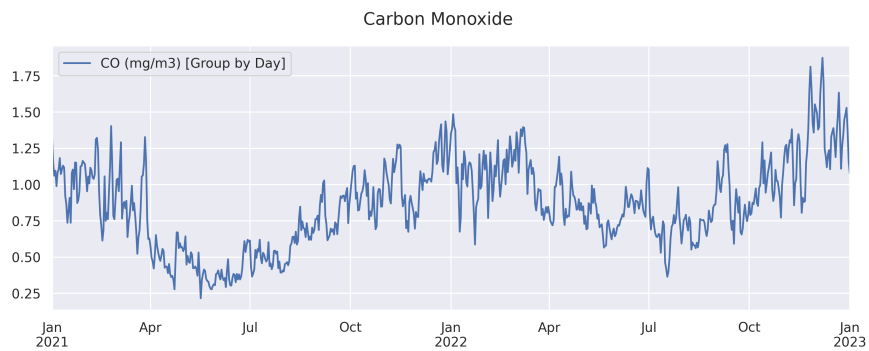


Figure 11: Carbon Monoxide concentration trends in Mumbai.

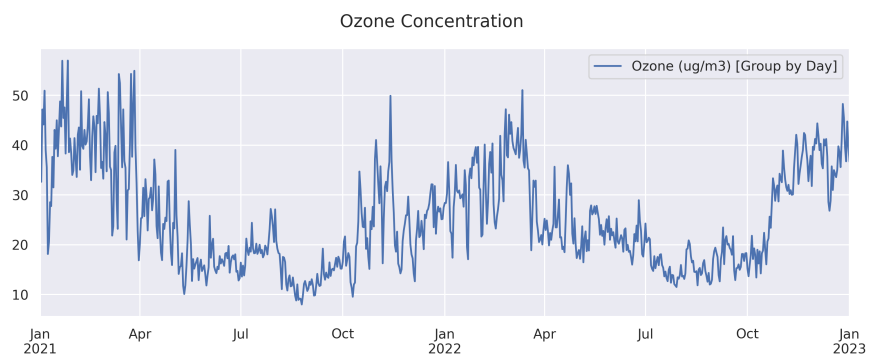


Figure 12: Ozone concentration trends in Mumbai.

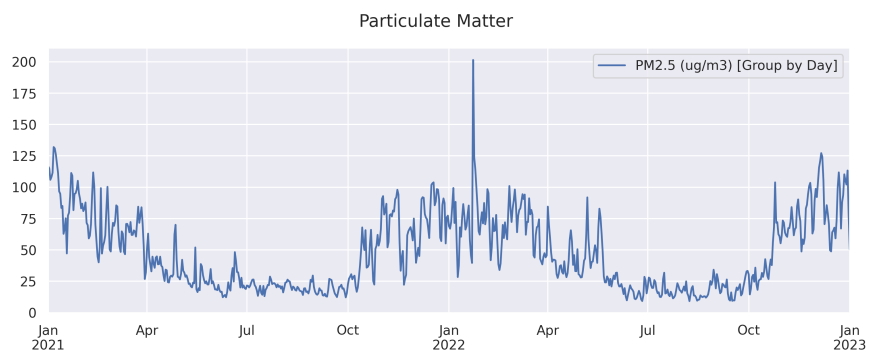


Figure 13: Particulate Matter (PM_{2.5}) concentration trends in Mumbai.

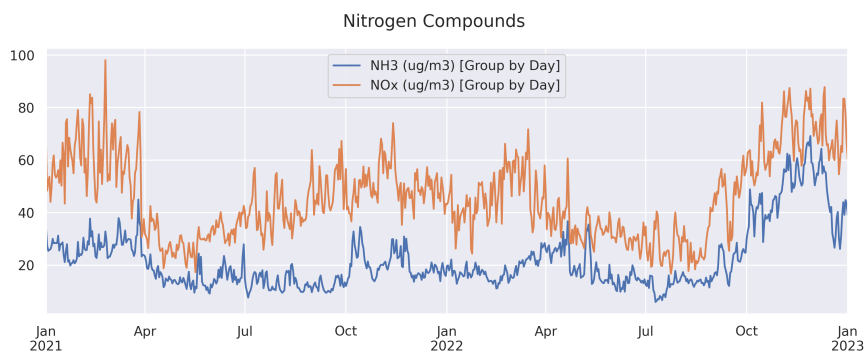


Figure 14: Nitrogen Compounds (NO_x) concentration trends in Mumbai.

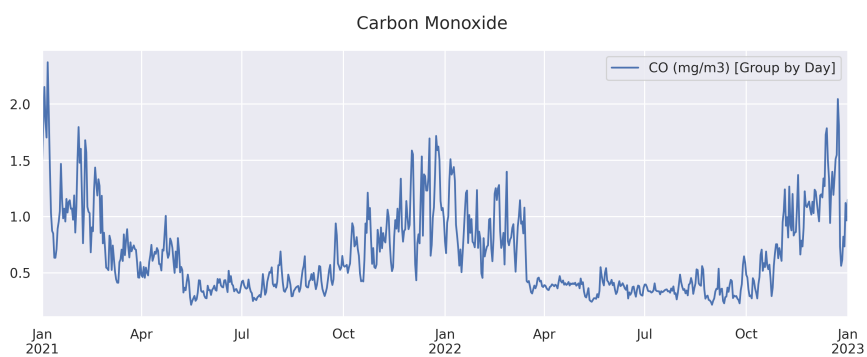


Figure 15: Carbon Monoxide concentration trends in Kolkata.

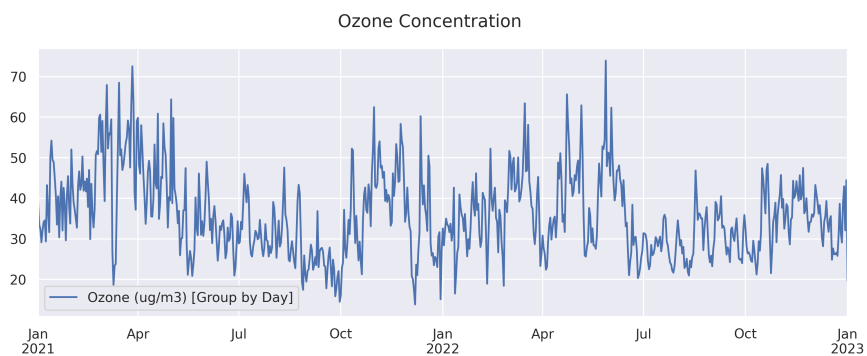


Figure 16: Ozone concentration trends in Kolkata.

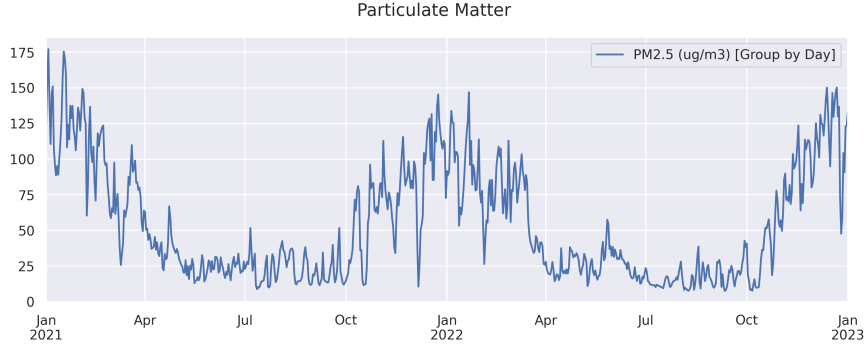


Figure 17: Particulate Matter (PM_{2.5}) concentration trends in Kolkata.

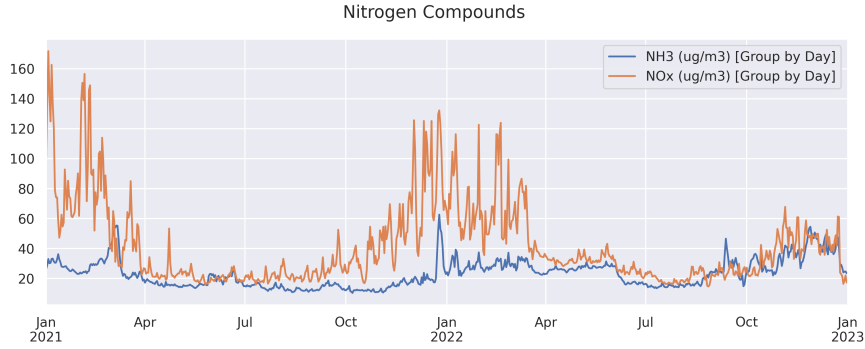


Figure 18: Nitrogen Compounds (NO_x) concentration trends in Kolkata.

4.4. Evaluation Matrix

Model performance was assessed using three standard statistical indicators: Root Mean Square Error (RMSE), Coefficient of Determination (R^2), and Mean Absolute Error (MAE). These are defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (17)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (18)$$

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (19)$$

4.5. Forecast Comparison and Analysis

For the purpose of model evaluation, the statistical, deep learning, and optimization-based approaches were configured with carefully selected hyperparameters. These configurations were determined through systematic experimentation to achieve stable training and reliable forecasting performance. The default hyperparameter settings used in this study are presented in Table 1.

The forecasting process involved separately modeling the linear and non-linear components of the decomposed AQI series and subsequently recombining them to obtain the final prediction. Comparative analysis against benchmark machine, deep learning and hybrid architectures with and without UAMMO has been shown below. Nevertheless, the proposed multi-resolution residual learning model, optimized using the integrated UAMMO framework, achieved

Table 1: Default Model Hyperparameter Settings

Model type	Models	Hyperparameter setting
Statistical model	ARIMA	NO _x = (2,0,3); CO = (5,0,0); O ₃ = (2,0,1); PM _{2.5} = (1,0,4)
Deep learning model	LSTM	hidden units=64, layers=1, batch size=32, epochs=157 with EarlyStopping, learning rate=0.001
	BiLSTM	hidden units=64, layers=1 (bidirectional), batch size=32, epochs=157 with EarlyStopping, learning rate=0.001
	CNN-BiLSTM + Residual-Gated Attention	Conv1D filters={32,64,128}, kernel sizes={3,5,7}; BiLSTM units=64; attention=gated with volatility; batch size=32, epochs=278 with EarlyStopping, learning rate=0.001
Optimization algorithm	UAMMO	population size=30, max iterations=50, inertia weight=0.9→0.4 (decay), α =adaptive (DBO+PSO+GA+GSA+RDA), stopping criterion $\epsilon < 10^{-3}$

competitive and notable results. 2 shows comparative analysis on Delhi pollution data, 3 shows comparative analysis on Mumbai pollution data and 4 shows comparative analysis on Kolkata pollution data

Table 2: Performance comparison of selected pollution forecasting methods across pollutants in Delhi.

Method	MSE				MAE				R ²			
	PM _{2.5}	O ₃	CO	NO _x	PM _{2.5}	O ₃	CO	NO _x	PM _{2.5}	O ₃	CO	NO _x
<i>Machine Learning Approaches</i>												
ARIMA	13.24212	2.18e ⁻⁵	0.00865	2.01e ⁻⁵	2.53490	0.00340	0.05898	0.00317	0.93825	0.92561	0.87282	0.92025
XGBoost	9.27569	2.16e ⁻⁵	0.00686	1.66e ⁻⁵	2.39400	0.00332	0.05590	0.00299	0.94721	0.92930	0.88910	0.93330
<i>Deep Learning Approaches</i>												
LSTM	13.39572	2.37e ⁻⁵	0.00799	1.82e ⁻⁵	2.59168	0.00349	0.06254	0.00309	0.93354	0.92032	0.87190	0.92722
BiLSTM	11.82768	2.26e ⁻⁵	0.00791	1.76e ⁻⁵	2.39453	0.00337	0.06108	0.00302	0.94252	0.92201	0.87730	0.92775
<i>Hybrid / Advanced Methods</i>												
UAMMO LSTM	12.68439	2.14e ⁻⁵	0.00748	1.79e ⁻⁵	2.45382	0.00334	0.05972	0.00301	0.94318	0.92544	0.88741	0.93112
UAMMO BiLSTM	11.04652	2.08e ⁻⁵	0.00729	1.71e ⁻⁵	2.32648	0.00329	0.05916	0.00293	0.94509	0.92719	0.88892	0.93492
Proposed Model	8.75412	2.07e⁻⁵	0.00651	1.53e⁻⁵	2.14379	0.00325	0.05491	0.00287	0.97918	0.95288	0.94273	0.95928

Table 3: Performance comparison of selected pollution forecasting methods across pollutants in Mumbai.

Method	MSE				MAE				R ²			
	PM _{2.5}	O ₃	CO	NO _x	PM _{2.5}	O ₃	CO	NO _x	PM _{2.5}	O ₃	CO	NO _x
<i>Machine Learning Approaches</i>												
ARIMA	13.14894	2.17e ⁻⁵	0.00861	2.00e ⁻⁵	2.52911	0.00339	0.05887	0.00316	0.93793	0.92503	0.87219	0.91994
XGBoost	9.24683	2.15e ⁻⁵	0.00683	1.65e ⁻⁵	2.38819	0.00331	0.05584	0.00298	0.94697	0.92891	0.88877	0.93298
<i>Deep Learning Approaches</i>												
LSTM	13.37029	2.36e ⁻⁵	0.00795	1.81e ⁻⁵	2.58604	0.00348	0.06241	0.00308	0.93328	0.92007	0.87137	0.92693
BiLSTM	11.80692	2.25e ⁻⁵	0.00787	1.75e ⁻⁵	2.38925	0.00336	0.06097	0.00301	0.94227	0.92176	0.87689	0.92745
<i>Hybrid / Advanced Methods</i>												
UAMMO LSTM	12.66814	2.13e ⁻⁵	0.00746	1.78e ⁻⁵	2.44902	0.00333	0.05962	0.00300	0.94294	0.92516	0.88710	0.93083
UAMMO BiLSTM	11.03122	2.07e ⁻⁵	0.00727	1.70e ⁻⁵	2.32177	0.00328	0.05907	0.00292	0.94485	0.92693	0.88862	0.93464
Proposed Model	8.74036	2.05e⁻⁵	0.00649	1.52e⁻⁵	2.13967	0.00324	0.05483	0.00286	0.97893	0.95261	0.94241	0.95879

Table 4: Performance comparison of selected pollution forecasting methods across pollutants in Kolkata.

Method	MSE				MAE				R ²			
	PM _{2.5}	O ₃	CO	NO _x	PM _{2.5}	O ₃	CO	NO _x	PM _{2.5}	O ₃	CO	NO _x
<i>Machine Learning Approaches</i>												
ARIMA	13.15439	2.17e ⁻⁵	0.00862	2.00e ⁻⁵	2.53027	0.00339	0.05889	0.00316	0.93799	0.92509	0.87223	0.91999
XGBoost	9.24855	2.15e ⁻⁵	0.00684	1.65e ⁻⁵	2.38860	0.00331	0.05586	0.00298	0.94699	0.92894	0.88879	0.93301
<i>Deep Learning Approaches</i>												
LSTM	13.37216	2.36e ⁻⁵	0.00796	1.81e ⁻⁵	2.58642	0.00348	0.06243	0.00308	0.93330	0.92009	0.87139	0.92695
BiLSTM	11.80813	2.25e ⁻⁵	0.00788	1.75e ⁻⁵	2.38949	0.00336	0.06098	0.00301	0.94229	0.92178	0.87691	0.92747
<i>Hybrid / Advanced Methods</i>												
UAMMO LSTM	12.66942	2.13e ⁻⁵	0.00747	1.78e ⁻⁵	2.44929	0.00333	0.05963	0.00300	0.94296	0.92518	0.88711	0.93085
UAMMO BiLSTM	11.03231	2.07e ⁻⁵	0.00728	1.70e ⁻⁵	2.32199	0.00328	0.05908	0.00292	0.94487	0.92695	0.88863	0.93466
Proposed Model	8.74129	2.05e⁻⁵	0.00650	1.52e⁻⁵	2.13989	0.00324	0.05484	0.00286	0.97895	0.95263	0.94243	0.95902

4.6. Quantitative Results

Tables 2, 3, and 4 presents the forecasting performance of the proposed model in terms of R^2 , RMSE, and MAE for the Delhi, Kolkata and Mumbai datasets. Across all three cities—Delhi, Mumbai, and Kolkata—the quantitative results show that the Proposed Model consistently outperforms all baseline methods (ARIMA, XGBoost, LSTM, BiLSTM, UAMMO LSTM, and UAMMO BiLSTM) across every pollutant and evaluation metric (MSE, MAE, and R^2). For example, in Delhi, the Proposed Model achieves the lowest MSE for $PM_{2.5}$ (8.75), improving upon the best baseline (XGBoost, 9.28) by roughly 5.6%, and similarly records the smallest MAE (2.14) alongside the highest R^2 (0.979), indicating both lower prediction errors and stronger variance explanation. This performance trend holds in Mumbai and Kolkata, where the Proposed Model's MSE values for all pollutants are uniformly the smallest (e.g., CO MSE = 0.00649 in Mumbai and 0.00650 in Kolkata vs. 0.00683 for the next-best XGBoost), while R^2 values reach above 0.94 for every pollutant—substantially higher than all competitors. Notably, the performance gap is most pronounced for NO_x forecasting, where the Proposed Model's R^2 surpasses 0.959 across cities, compared to ≤ 0.935 for the best existing approaches. The improvements, although numerically modest in some metrics (especially for O_3), are consistent across different urban environments and pollutant types, indicating both robustness and generalizability of the model's predictive capabilities.

5. Conclusion

This study presented a decomposition-driven hybrid forecasting framework for air quality prediction, integrating statistical modeling for linear components with a multi-resolution CNN–BiLSTM–residual-gated attention network for non-linear residuals. By incorporating the Unified Adaptive Multi-Stage Metaheuristic Optimizer (UAMMO), the model effectively selected optimal hyperparameters, balancing exploration and exploitation across multiple search strategies. Applied to AQI data from Delhi NCR, Kolkata and Mumbai the proposed approach demonstrated high predictive accuracy, in $PM_{2.5}$, O_3 , CO, NO_x . Overall, the proposed methodology offers a interpretable solution for urban air quality forecasting, supporting data-driven environmental policy and public health decision-making.

References

- [1] A. Selokar, B. Ramachandran, K. Elangovan, B. D. Varma, $PM_{2.5}$ particulate matter and its effects in delhi/ncr, *Materials Today: Proceedings* 33 (2020) 4566–4572.
- [2] World Bank, Catalyzing clean air in india, <https://www.worldbank.org/en/country/india/publication/catalyzing-clean-air-in-india> (2024).
- [3] G. R. Naik, A. K. Shah, Air pollution in india: a comprehensive review on impacts, challenges and mitigation strategies, *International Journal of Community Medicine and Public Health* 12 (2) (2025) 1136–1140. doi:10.18203/2394-6040.ijcmph20250335. URL <https://doi.org/10.18203/2394-6040.ijcmph20250335>
- [4] N. A. B. Mabawi, O. L. H. Leh, D. Omar, Human health and wellbeing: Human health effect of air pollution, *Procedia-Social and Behavioral Sciences* 153 (2014) 221–229.
- [5] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput* 9. doi:10.1162/neco.1997.9.8.1735. URL <https://doi.org/10.1162/neco.1997.9.8.1735>
- [6] D. Xu, Q. Zhang, Y. Ding, D. Zhang, Application of a hybrid arima-lstm model based on the spei for drought forecasting, *Environ Sci Pollut Res* 29. doi:10.1007/s11356-021-15325-z. URL <https://doi.org/10.1007/s11356-021-15325-z>
- [7] J. J. Carbajal-Hernández, L. P. Sánchez-Fernández, J. A. Carrasco-Ochoa, J. F. Martínez-Trinidad, Assessment and prediction of air quality using fuzzy logic and autoregressive models, *Atmos Environ* 60. doi:10.1016/j.atmosenv.2012.06.004. URL <https://doi.org/10.1016/j.atmosenv.2012.06.004>
- [8] L. Zhang, X. Tian, Y. Zhao, L. Liu, Z. Li, L. Tao, X. Wang, X. Guo, Y. Luo, Application of nonlinear land use regression models for ambient air pollutants and air quality index, *Atmos Pollut Res* 12. doi:10.1016/j.apr.2021.101186. URL <https://doi.org/10.1016/j.apr.2021.101186>
- [9] L. Zhao, Z. Li, L. Qu, Forecasting of beijing $pm_{2.5}$ with a hybrid arima model based on integrated aic and improved gs fixed-order methods and seasonal decomposition, *Heliyon* 8. doi:10.1016/j.heliyon.2022.e12239. URL <https://doi.org/10.1016/j.heliyon.2022.e12239>
- [10] W. Zhou, X. Wu, S. Ding, Y. Cheng, Predictive analysis of the air quality indicators in the yangtze river delta in china: an application of a novel seasonal grey model, *Sci Total Environ* 748. doi:10.1016/j.scitotenv.2020.141428. URL <https://doi.org/10.1016/j.scitotenv.2020.141428>
- [11] K. Mehmood, Y. Bao, W. Cheng, M. A. Khan, N. Siddique, M. M. Abrar, A. Soban, S. Fahad, R. Naidu, Predicting the quality of air with machine learning approaches: current research priorities and future perspectives, *J Clean Prod* 379. doi:10.1016/j.jclepro.2022.134656. URL <https://doi.org/10.1016/j.jclepro.2022.134656>

- [12] U. Mahalingam, K. Elangovan, H. Dobhal, C. Valliappa, S. Shrestha, G. Kedam, A machine learning model for air quality prediction for smart cities, in: 2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET), 2019, pp. 452–457. doi:10.1109/WiSPNET45539.2019.9032734.
- [13] A. H. Elsheikh, Applications of machine learning in friction stir welding: prediction of joint properties, real-time control and tool failure diagnosis, *Eng Appl Artif Intell* 121. doi:10.1016/j.engappai.2023.105961. URL <https://doi.org/10.1016/j.engappai.2023.105961>
- [14] H. Ke, S. Gong, J. He, L. Zhang, B. Cui, Y. Wang, J. Mo, Y. Zhou, H. Zhang, Development and application of an automated air quality forecasting system based on machine learning, *Sci Total Environ* 806. doi:10.1016/j.scitotenv.2021.151204. URL <https://doi.org/10.1016/j.scitotenv.2021.151204>
- [15] W. Zhang, Y. Wu, J. K. Calautit, A review on occupancy prediction through machine learning for enhancing energy efficiency, air quality and thermal comfort in the built environment, *Renew Sustain Energy Rev* 167. doi:10.1016/j.rser.2022.112704. URL <https://doi.org/10.1016/j.rser.2022.112704>
- [16] Y. Gu, B. Li, Q. Meng, Hybrid interpretable predictive machine learning model for air pollution prediction, *Neurocomputing* 468. doi:10.1016/j.neucom.2021.09.051. URL <https://doi.org/10.1016/j.neucom.2021.09.051>
- [17] R. Rakholia, Q. Le, B. Q. Ho, K. Vu, R. S. Carbajo, Multi-output machine learning model for regional air pollution forecasting in ho chi minh city, vietnam, *Environ Int* 173. doi:10.1016/j.envint.2023.107848. URL <https://doi.org/10.1016/j.envint.2023.107848>
- [18] R. Janarthanan, P. Partheeban, K. Somasundaram, P. N. Elamparithi, A deep learning approach for prediction of air quality index in a metropolitan city, *Sustain Cities Soc* 67. doi:10.1016/j.scs.2021.102720. URL <https://doi.org/10.1016/j.scs.2021.102720>
- [19] B. Zhang, Y. Rong, R. Yong, D. Qin, M. Li, G. Zou, J. Pan, Deep learning for air pollutant concentration prediction: a review, *Atmos Environ* 290. doi:10.1016/j.atmosenv.2022.119347. URL <https://doi.org/10.1016/j.atmosenv.2022.119347>
- [20] M. Saez, M. A. Barceló, Spatial prediction of air pollution levels using a hierarchical bayesian spatiotemporal model in catalonia, spain, *Environ Model Softw* 151. doi:10.1016/j.envsoft.2022.105369. URL <https://doi.org/10.1016/j.envsoft.2022.105369>
- [21] X. Jurado, N. Reiminger, M. Benmoussa, J. Vazquez, C. Wemmert, Deep learning methods evaluation to predict air quality based on computational fluid dynamics, *Expert Syst Appl* 203. doi:10.1016/j.eswa.2022.117294. URL <https://doi.org/10.1016/j.eswa.2022.117294>
- [22] X. Zhou, J. Xu, P. Zeng, X. Meng, Air pollutant concentration prediction based on gru method, *J Phys Conf Ser* 1168. doi:10.1088/1742-6596/1168/3/032058. URL <https://doi.org/10.1088/1742-6596/1168/3/032058>
- [23] W. Mao, W. Wang, L. Jiao, S. Zhao, A. Liu, Modeling air quality prediction using a deep learning approach: method optimization and evaluation, *Sustain Cities Soc* 65. doi:10.1016/j.scs.2020.102567. URL <https://doi.org/10.1016/j.scs.2020.102567>
- [24] A. H. Elsheikh, V. P. Katekar, O. L. Muskens, S. S. Deshmukh, M. A. Elaziz, S. M. Dabour, Utilization of lstm neural network for water production forecasting of a stepped solar still with a corrugated absorber plate, *Process Saf Environ Prot* 148. doi:10.1016/j.psep.2020.09.068. URL <https://doi.org/10.1016/j.psep.2020.09.068>
- [25] F. Djouider, M. A. Elaziz, A. Alhawsawi, E. Banoqitah, E. B. Moustafa, A. H. Elsheikh, Experimental investigation and machine learning modeling using lstm and special relativity search of friction stir processed aa2024/al2o3 nanocomposites, *J Market Res* 27. doi:10.1016/j.jmrt.2023.11.155. URL <https://doi.org/10.1016/j.jmrt.2023.11.155>
- [26] Q. Wu, H. Lin, A novel optimal-hybrid model for daily air quality index prediction considering air pollutant factors, *Sci Total Environ* 683. doi:10.1016/j.scitotenv.2019.05.288. URL <https://doi.org/10.1016/j.scitotenv.2019.05.288>
- [27] N. Sarkar, R. Gupta, P. K. Keserwani, M. C. Govil, Air quality index prediction using an effective hybrid deep learning model, *Environ Pollut* 315. doi:10.1016/j.envpol.2022.120404. URL <https://doi.org/10.1016/j.envpol.2022.120404>
- [28] A. Gilik, A. S. Ogrenici, A. Ozmen, Air quality prediction using cnn+ lstm-based hybrid deep learning architecture, *Environ Sci Pollut Res* 29. doi:10.1007/s11356-021-16227-w. URL <https://doi.org/10.1007/s11356-021-16227-w>
- [29] S. Zhu, X. Lian, H. Liu, J. Hu, Y. Wang, J. Che, Daily air quality index forecasting with hybrid models: a case in china, *Environ Pollut* 231. doi:10.1016/j.envpol.2017.08.069. URL <https://doi.org/10.1016/j.envpol.2017.08.069>
- [30] Y.-S. Chang, S. Abimannan, H.-T. Chiao, C.-Y. Lin, Y.-P. Huang, An ensemble learning based hybrid model and framework for air pollution forecasting, *Environ Sci Pollut Res* 27. doi:10.1007/s11356-020-09855-1. URL <https://doi.org/10.1007/s11356-020-09855-1>
- [31] J. Wang, J. Li, X. Wang, M. Huang, Air quality prediction using ct-lstm, *Neural Comput Appl* 33. doi:10.1007/s00521-020-05535-w. URL <https://doi.org/10.1007/s00521-020-05535-w>
- [32] A. H. Elsheikh, A. I. Saba, M. A. Elaziz, S. Lu, S. Shanmugan, T. Muthuramalingam, R. Kumar, A. O. Mosleh, F. A. Essa, T. A. Shehabeldeen, Deep learning-based forecasting model for covid-19 outbreak in saudi arabia, *Process Saf Environ Prot* 149. doi:10.1016/j.psep.2020.10.048. URL <https://doi.org/10.1016/j.psep.2020.10.048>

- [33] H. Dai, G. Huang, H. Zeng, R. Yu, Haze risk assessment based on improved pca-mee and ispo-lightgbm model, Systems 10. doi:10.3390/systems10060263.
URL <https://doi.org/10.3390/systems10060263>
- [34] A. I. Saba, A. H. Elsheikh, Forecasting the prevalence of covid-19 outbreak in egypt using nonlinear autoregressive artificial neural networks, Process Saf Environ Prot 141. doi:10.1016/j.psep.2020.05.029.
URL <https://doi.org/10.1016/j.psep.2020.05.029>
- [35] Central Pollution Control Board, <https://cpcb.nic.in>.
- [36] OpenStreetMap contributors, Planet dump retrieved from <https://planet.osm.org> , <https://www.openstreetmap.org> (2017).