International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 08, August 2026)

AI-Based Air Quality Forecasting System Using Multivariate LSTM Networks

Manideep Pothuri
III/IV B.Tech (CSE), VIT-AP (Amaravathi), India

Abstract-- **Air pollution is a major environmental and public-health challenge in India, with fine particulate matter (PM2.5) being one of the most important pollutants affecting air quality. Accurate short-term forecasting of PM2.5 can support proactive pollution management and public-health decision-making. This study presents a multivariate Long Short-Term Memory (LSTM)-based framework for one-hour-ahead PM2.5 forecasting using large-scale air-quality monitoring data from India. The processed dataset contains 10,838,179 observations incorporating pollutant, meteorological, temporal, and station-related variables. The data are divided station-wise using an 80:10:10 chronological train-validation-test strategy to prevent temporal leakage. Training-only Min-Max normalization and gap-aware sequence generation are employed to construct valid 24-hour continuous input sequences. The LSTM architecture consists of two recurrent layers with 64 and 32 units, followed by dropout and dense layers. The model is trained using the Adam optimizer and Mean Squared Error loss, with early stopping and learning-rate reduction. Evaluation on 1,015,520 unseen test predictions achieved an MAE of 23.0711 µg/m³, RMSE of 39.7409 µg/m³, and R^2 of 0.6880. The model performed better under low-to-moderate pollution conditions, while errors increased during extreme pollution events. The proposed framework provides a foundation for proactive air-quality monitoring, early-warning systems, and future spatiotemporal forecasting applications.**

Keywords— **PM2.5, Air Quality Forecasting, Deep Learning, LSTM, Machine Learning, Time-Series Forecasting, India, Air Pollution.**

I. INTRODUCTION

Air pollution has become one of the most important environmental challenges associated with urbanization and economic development. Fine particulate matter (PM2.5) is particularly important because its small aerodynamic diameter allows it to penetrate deeply into the respiratory system. Forecasting PM2.5 concentrations several hours in advance can therefore help authorities, organizations, and citizens take preventive action before pollution reaches hazardous levels. India has an extensive network of ambient-air monitoring stations operated through the Central Pollution Control Board (CPCB), State Pollution Control Boards, Pollution Control Committees, and other participating organizations. CPCB reports that ambient monitoring includes pollutants such as PM10, SO_2, NO_2, CO, NH_3, O_3, and PM2.5 at selected locations.

Its Continuous Ambient Air Quality Monitoring System (CAAQMS) generates real-time measurements that are also used in the calculation of the National Air Quality Index.

Air pollution is a complex time-dependent phenomenon. Concentrations are influenced not only by emissions but also by temperature, humidity, wind, atmospheric pressure, rainfall, boundary-layer conditions, seasonality, traffic patterns, and other environmental factors. Consequently, a forecasting model based solely on historical PM2.5 may not adequately represent the underlying dynamics. A multivariate model that combines pollutant and meteorological information can provide a richer representation of the factors affecting future concentrations.

Deep-learning methods have increasingly been applied to air-quality prediction because they can learn nonlinear relationships directly from large time-series datasets. In particular, LSTM networks are well suited to sequential data because their memory mechanisms allow them to retain relevant information over multiple time steps. Previous research has demonstrated the effectiveness of LSTM-based approaches for PM2.5 forecasting, including studies using multivariate pollutant and meteorological inputs.

Research conducted in India also supports the potential of deep learning for PM2.5 prediction. A study focused on Hyderabad reported that an LSTM model outperformed several conventional machine-learning models for PM2.5 modeling, obtaining an R^2 of 0.89 and MAE of 5.78 µg/m³ in that particular experimental setting. The present research therefore develops a generalized AI-based framework for forecasting PM2.5 using monitoring-station observations across India.

II. RELATED WORK

India's air-quality monitoring infrastructure consists of conventional monitoring under national programs as well as continuous monitoring stations. CPCB documentation identifies PM10, SO_2, and NO_2 as important routinely monitored pollutants under the National Ambient Air Monitoring Programme, while other pollutants—including PM2.5, CO, and O_3—are monitored at selected locations. CAAQMS provides higher-frequency observations and is particularly valuable for machine-learning applications because deep-learning forecasting requires sufficiently dense sequential observations. CPCB's reported network has expanded substantially, with monitoring stations distributed across multiple states and Union Territories. However, nationwide datasets present several challenges. Monitoring stations may differ in the pollutants measured, sampling frequency, instrumentation, station environment, and duration of operation. Consequently, preprocessing is an essential component of a nationwide forecasting system.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 08, August 2026)

Air pollution is a major environmental problem, and PM2.5 is one of the most important indicators because elevated concentrations are associated with harmful air-quality conditions. A monitoring system that only reports the present value is reactive; forecasting the next hour provides an opportunity to identify deterioration in advance. The objective of this project is to build a practical deep-learning forecasting pipeline that learns relationships between recent environmental observations and the PM2.5 concentration expected one hour later. Instead of using only a single pollutant, the implemented model uses multiple pollutant, meteorological, location and temporal variables. LSTM is selected because its recurrent structure is designed for sequential data and can learn patterns across a fixed historical window.

III. Problem Statement

Given the environmental and temporal observations available for a monitoring station during the previous 24 continuous hourly time steps, predict the PM2.5 concentration for the following hour. The system should handle missing time intervals safely, avoid temporal leakage, and provide quantitative performance measurements on previously unseen test data.

IV. System Overview

The implemented workflow is: Processed data → Station-wise chronological split → Training-only scaling → Gap-aware 24-hour sequence generation → Multivariate LSTM → PM2.5 prediction → Inverse scaling → Evaluation / station-level forecasting.

The project has two main execution paths. The training/evaluation path builds the datasets, trains the model and measures performance. The inference path loads the already-trained model and scalers, selects a station, takes its latest 24 hourly observations, and predicts PM2.5 for the next hour.

V. Dataset And Data Preparation

The implemented processed dataset contains 10,838,179 rows. Observations are sorted by Station and Date before sequence generation.

The data are divided independently for each station: 80% earliest observations → training set; next 10% observations → validation set; final 10% observations → test set.

This station-wise chronological split is important because random splitting could allow information from later periods to influence training. The final test evaluation produced 1,015,520 predictions.

1. Input Features

The model uses 22 predictor features: Station ID, State, City, PM10, NO2, SO2, CO, Ozone, Temperature, Humidity, WindSpeed, WindDirection_sin, WindDirection_cos, Year, Month, Day, Hour, DayOfWeek, Hour_sin, Hour_cos, Month_sin, Month_cos.

Target variable: PM2.5. The target is predicted for the hour immediately following the 24-hour input sequence.

2. Normalization

Two MinMaxScaler objects are used. The feature scaler is fitted only on the training predictor values, and the target scaler is fitted only on training PM2.5 values. The fitted scalers are saved as feature_scaler.pkl and target_scaler.pkl so that the exact training transformation can be reused during evaluation and prediction.

Min-max normalization maps a value approximately to the range [0, 1] using the training-set minimum and maximum.

3. Gap-Aware Sequence Generation

The sequence generator groups observations by station, sorts each group chronologically, and identifies breaks where consecutive timestamps are not exactly one hour apart. Sequences are generated only inside continuous hourly segments. With a sequence length of 24, each input X contains 24 consecutive hourly observations and the target y is the PM2.5 value at the next hour.

Conceptually: 24 hourly observations → LSTM → PM2.5 at t + 1.

VI. Feature Engineering

The processed dataset contains temporal and wind-direction transformations that make the variables more suitable for machine learning. Wind direction is represented using sine and cosine components so that the circular nature of direction is preserved. Hour and month are also represented using sine/cosine encodings, avoiding an artificial discontinuity between the end and beginning of a cycle.

WindDirection_sin and WindDirection_cos; Hour_sin and Hour_cos; Month_sin and Month_cos; Calendar variables: Year, Month, Day, Hour and DayOfWeek.

VII. LSTM Model Architecture

The final trained model has the following architecture:

Layer	Output Shape	Parameters	Purpose
LSTM	(None, 24, 64)	22,272	Learns temporal patterns while returning the sequence
Dropout	(None, 24, 64)	0	Regularization
LSTM	(None, 32)	12,416	Compresses the learned sequence into a representation
Dropout	(None, 32)	0	Regularization
Dense	(None, 16)	528	Learns nonlinear combinations
Dense output	(None, 1)	17	Predicts PM2.5

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 08, August 2026)

Total trainable parameters: 35,233 (approximately 137.63 KB). Architecture flow: 24×22 input → LSTM(64) → Dropout → LSTM(32) → Dropout → Dense(16) → PM2.5.

VIII. Model Training

The model is trained with Adam optimization and Mean Squared Error (MSE) loss. The configured maximum number of epochs is 20, but early stopping terminates training when validation performance no longer improves. ReduceLROnPlateau decreases the learning rate when validation loss stalls.

Input sequence length: 24 hours; Number of input features: 22; Batch size: 64; Initial learning rate: 0.001; Maximum epochs: 20; Actual epochs completed: 14; Best validation epoch: 11; Best model: models/best_lstm_model.keras.

The training log shows validation loss improving substantially during the first several epochs and reaching its best recorded value at approximately epoch 11. Training then stopped at epoch 14 and restored the best weights.

IX. Evaluation Metrics

MAE measures the average absolute difference between actual and predicted PM2.5 values. It is directly expressed in µg/m³; lower is better.

RMSE penalizes large errors more strongly than MAE and is therefore useful for understanding the effect of large prediction errors. Lower is better.

R^2 measures how much of the variation in the target values is explained by the model. A value closer to 1 indicates stronger explanatory performance.

X. Final Experimental Results

The saved best LSTM model was evaluated on 1,015,520 unseen test predictions.

The results show that the model performs substantially better in the common low-to-moderate PM2.5 ranges. Errors increase as pollution becomes extreme, especially above 300 µg/m³. This is important because the rare high-pollution observations have a strong effect on RMSE and are harder for the model to predict accurately.

XI. Prediction / Inference System

The prediction script loads the trained best model and the two saved scalers without retraining. It reads the processed dataset, lists available station IDs, accepts a station ID from the user, selects the station's latest 24 observations, scales the 22 input features using the training scaler, and sends the sequence to the LSTM.

The output contains: Station ID; Last available timestamp; Forecast timestamp (one hour later); Current/latest PM2.5; Predicted next-hour PM2.5; Pollution category.

Example inference from the implemented system: for Station 4, the latest observed PM2.5 was 23.00 µg/m³ and the predicted PM2.5 for 2023-04-01 00:00 was 29.49 µg/m³, classified by the project's category function as Good.

XII. Project Files And Their Roles

sequence_generator.py — Loads the processed dataset, performs station-wise chronological splitting, fits/transforms scalers, creates gap-aware 24-hour TensorFlow datasets.

train_lstm.py — Builds and trains the LSTM, applies callbacks, saves the best/final model and training scalers.

evaluate_lstm.py — Loads the best model, generates predictions over the test dataset, converts them back to µg/m³ and calculates MAE, RMSE and R^2.

predict.py — Interactive station-level inference using the already-trained model; no retraining is performed.

generate_plots.py — Creates model evaluation plots from the saved actual and predicted test arrays.

best_lstm_model.keras — Best model checkpoint selected using validation loss; this is the primary model used for inference.

final_lstm_model.keras — Final saved training model. The best model is preferred for the reported results.

feature_scaler.pkl — Training-fitted feature normalization object used during inference.

target_scaler.pkl — Training-fitted PM2.5 normalization object used to convert predictions back to physical units.

test_actual.npy — Saved actual PM2.5 test values.

test_predictions.npy — Saved model predictions for the same test samples.

XIII. Limitations

The current model predicts only one hour ahead.

The test results show significantly larger errors during extreme pollution events.

The model is primarily temporal; it does not explicitly model spatial relationships between neighboring stations.

The current feature set does not include historical PM2.5 as an input feature; PM2.5 is the prediction target.

The prediction script uses the latest available 24 observations, so a production system would need robust handling for missing or delayed live observations.

A single aggregate test score does not fully describe performance across every station, season or pollution regime.

XIV. Future Work

Extend forecasting to 3-, 6-, 12- and 24-hour horizons.

Add historical PM2.5 lag features if experimentally justified.

Compare the LSTM against persistence, Random Forest, XGBoost and other baselines.

Use attention mechanisms or Transformer-LSTM architectures.

Incorporate station coordinates and spatial relationships using graph-based or spatiotemporal models.

Add prediction uncertainty and confidence intervals.

Improve performance on rare extreme-pollution events using suitable sampling, loss functions or specialized objectives.

Deploy the inference pipeline behind a web dashboard or API.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 08, August 2026)

XV. Practical Applications

Public-Health Alerts: Forecasts can be used to issue advance warnings when PM2.5 is expected to rise substantially.

Traffic Management: Authorities can combine forecasts with traffic information to identify periods during which transportation emissions may contribute to poor air quality.

Industrial Pollution Management: Industrial areas can use forecast information to support emission-management strategies during unfavorable dispersion conditions.

Urban Planning: Long-term forecasting and spatial analysis can provide information useful for urban environmental planning.

Citizen Information Systems: Forecasts can be presented through mobile applications, web dashboards, or digital displays to help citizens plan outdoor activities.

XVI. Conclusion

This research proposes an AI-based air quality forecasting system for India that combines historical pollutant concentrations, meteorological observations, and temporal information to forecast future PM2.5 concentrations. The proposed methodology covers the complete machine-learning workflow, including data acquisition, cleaning, exploratory analysis, feature engineering, sequence generation, normalization, LSTM modeling, and time-series evaluation.

The principal contribution is a multivariate forecasting framework in which PM2.5 is predicted using its historical behavior together with PM10, NO_2, SO_2, CO, O_3, meteorological conditions, and temporal features. LSTM is an appropriate baseline because its memory architecture is designed to capture temporal dependencies in sequential observations. Existing research, including studies conducted in Indian cities, provides evidence supporting the application of LSTM and related deep-learning architectures to PM2.5 forecasting.

The proposed system can form the analytical core of a proactive air-quality monitoring platform. By forecasting pollution before it occurs, rather than merely reporting current conditions, the system can provide valuable information for public-health alerts, environmental management, urban planning, and pollution-control strategies. Future extensions involving spatial relationships, attention mechanisms, Transformer architectures, uncertainty estimation, and explainable AI could transform the baseline LSTM framework into a more robust nationwide air-quality forecasting platform.

TABLE I
FINAL TEST RESULTS

Metric	Test Result
MAE	23.0711 µg/m³
RMSE	39.7409 µg/m³
R^2	0.6880
Test predictions	1,015,520
Actual PM2.5 mean	77.21496 µg/m³
Predicted PM2.5 mean	79.70264 µg/m³

TABLE II
PERFORMANCE BY PM2.5 RANGE

Actual PM2.5 range	Count	MAE	RMSE
0–50 µg/m³	451,348	16.93	23.25
50–100 µg/m³	316,497	16.33	23.75
100–200 µg/m³	185,356	32.49	42.40
200–300 µg/m³	45,585	55.33	72.52
300–500 µg/m³	14,775	94.84	124.86
500–1000 µg/m³	1,959	344.15	439.18

REFERENCES

[1] Central Pollution Control Board (CPCB), Annual Report 2021–22, Government of India.

[2] Central Pollution Control Board (CPCB), Annual Report 2019–20, Government of India.

[3] S. Abirami and P. Chitra, "Regional air quality forecasting using spatiotemporal deep learning," Journal of Cleaner Production, vol. 283, Art. no. 125341, 2021.

[4] Central Pollution Control Board, Continuous Ambient Air Quality Monitoring System (CAAQMS): Guidelines and Technical Specifications, Ministry of Environment, Forest and Climate Change, Government of India, 2024.

[5] F. Chollet, Deep Learning with Python, 2nd ed. Manning Publications, 2021.

[6] K. Kumar and B. P. Pande, "Air pollution prediction with machine learning: A case study of Indian cities," Environmental Monitoring and Assessment, vol. 195, Art. no. 1003, 2023.

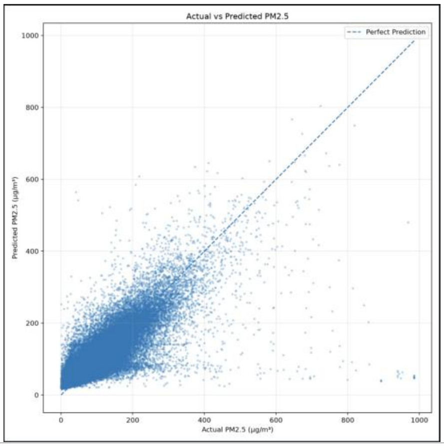
[7] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Advances in Neural Information Processing Systems, vol. 30, 2017.

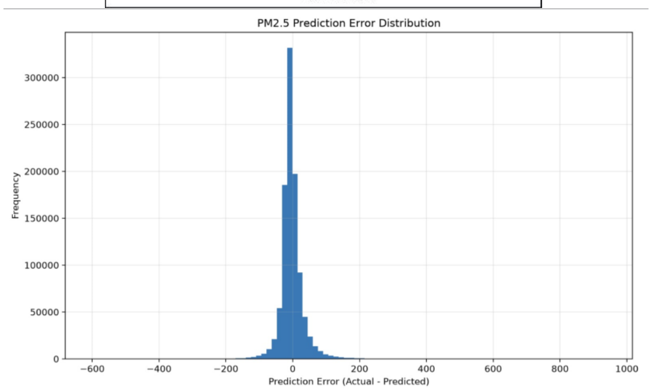
[8] S. Mohan, A. Govardhan, and G. Srivastava, "Air quality prediction using machine learning techniques: A review," Environmental Monitoring and Assessment, vol. 193, pp. 1–19, 2021.

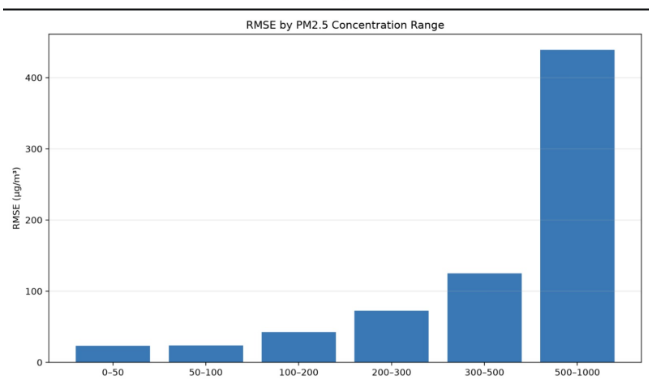
[9] S. R. Shams, A. Jahani, S. Kalantary, and S. Mowlaei, "Artificial intelligence for forecasting air pollution: A systematic review," Environmental Science and Pollution Research, vol. 28, pp. 53920–53939, 2021.

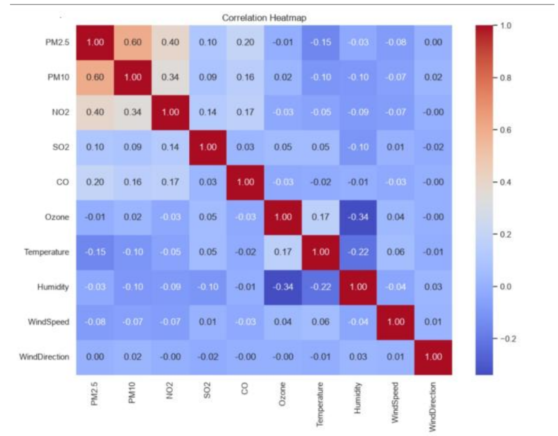
International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 08, August 2026)

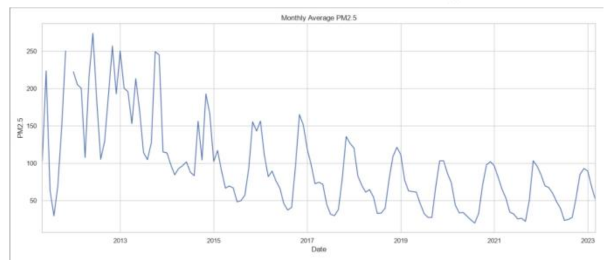
Outputs And Analysis

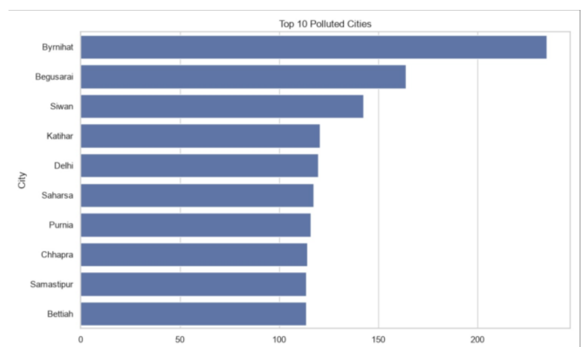


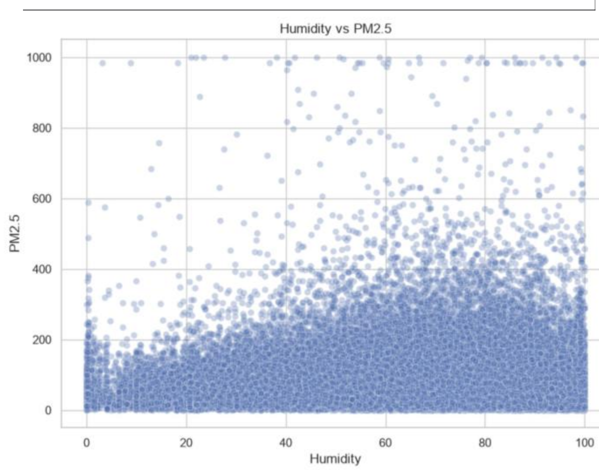




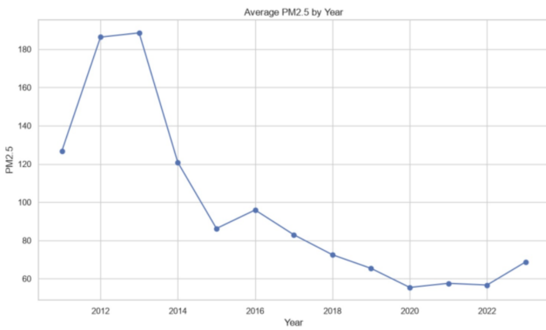


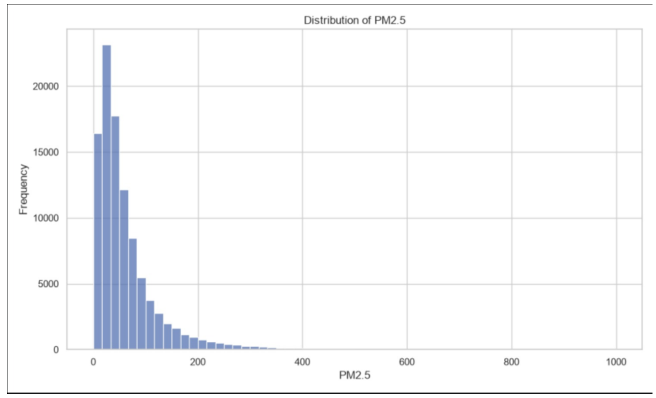


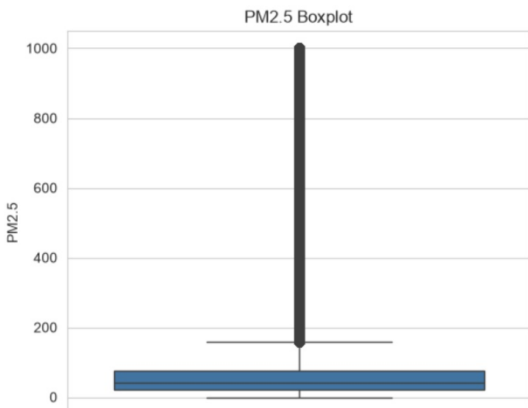


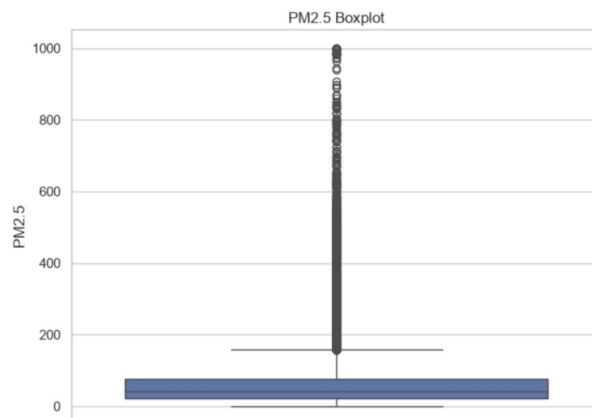


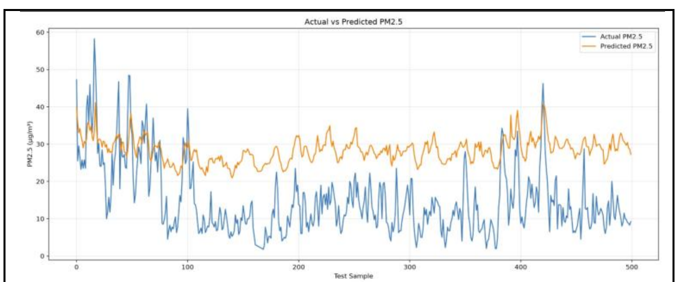
International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 08, August 2026)

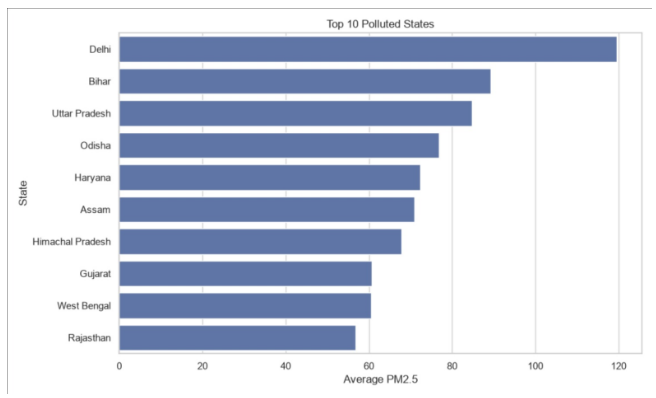


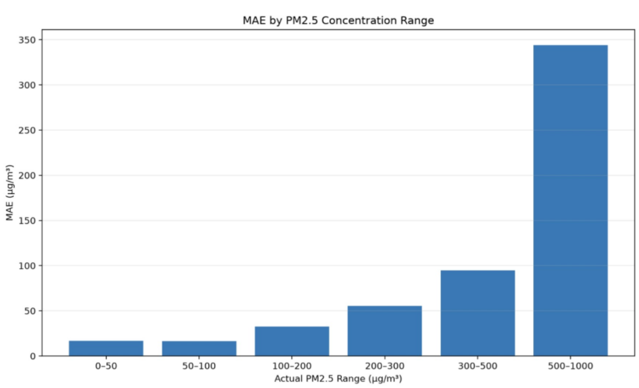


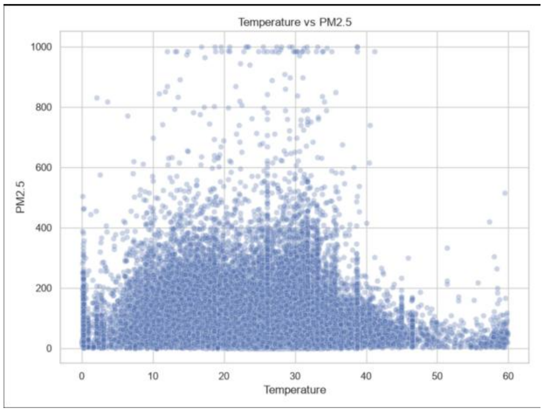


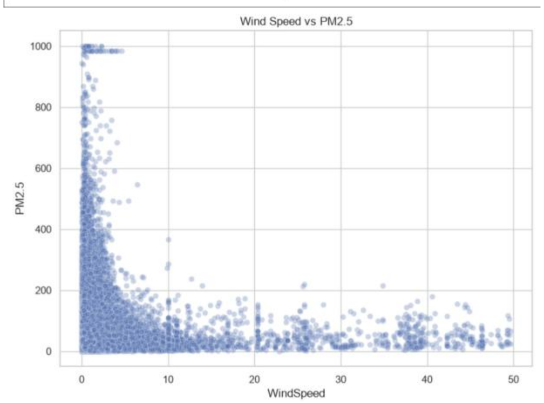


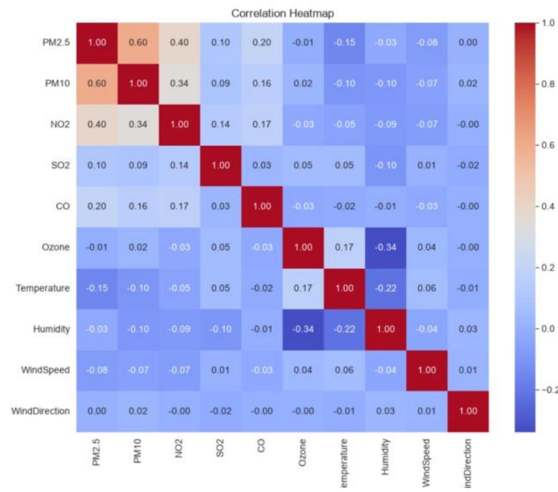


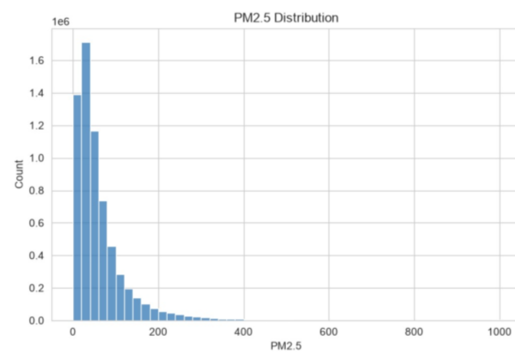





Temperature vs PM2.5
PM2.5
Temperature


Wind Speed vs PM2.5
PM2.5
WindSpeed


Correlation Heatmap
PM2.5 1.00 0.60 0.40 0.10 0.20 -0.01 -0.15 -0.03 -0.08 0.00
PM10 0.60 1.00 0.34 0.09 0.16 0.02 -0.10 -0.10 -0.07 0.02
NO2 0.40 0.34 1.00 0.14 0.17 -0.03 -0.05 -0.09 -0.07 -0.00
SO2 0.10 0.09 0.14 1.00 0.03 0.05 0.05 -0.10 0.01 -0.02
CO 0.20 0.16 0.17 0.03 1.00 -0.03 -0.02 -0.01 -0.03 -0.00
Ozone -0.01 0.02 -0.03 0.05 -0.03 1.00 0.17 -0.34 0.04 -0.00
Temperature -0.15 -0.10 -0.05 0.05 -0.02 0.17 1.00 -0.22 0.06 -0.01
Humidity -0.03 -0.10 -0.09 -0.10 -0.01 -0.34 -0.22 1.00 -0.04 0.03
WindSpeed -0.08 -0.07 -0.07 0.01 -0.03 0.04 0.06 -0.04 1.00 0.01
WindDirection 0.00 0.02 -0.00 -0.02 -0.00 -0.00 -0.01 0.03 0.01 1.00
PM2.5
PM10
NO2
SO2
CO
Ozone
emperature
Humidity
WindSpeed
indDirection


PM2.5 Distribution
1e6
Count
PM2.5