**Air Quality Prediction Using Machine Learning**

Raviteja Reddy Pichhala

Rivier University

SP25 - COMP-699-G - PROFESSIONAL SEMINAR

Table of Contents

[Abstract 4](#_Toc189754222)

[1. Introduction 5](#_Toc189754223)

[1.1 Background 5](#_Toc189754224)

[1.2 Problem Statement 5](#_Toc189754225)

[1.3 Objectives 6](#_Toc189754226)

[1.4 Research Questions 6](#_Toc189754227)

[1.5 Significance of Study 7](#_Toc189754228)

[2. Literature Review 7](#_Toc189754229)

[2.1 Air Pollution and the Need for AQI Prediction 7](#_Toc189754230)

[2.2 Machine Learning in Air Quality Prediction 8](#_Toc189754231)

[2.3 Feature Selection for AQI Prediction 9](#_Toc189754232)

[2.4 Existing Studies on AQI Prediction 9](#_Toc189754233)

[2.5 Addressing Research Gaps and Future Directions 10](#_Toc189754234)

[3. Methodology 11](#_Toc189754235)

[3.1 Dataset Description 11](#_Toc189754236)

[3.2 Data Preprocessing 11](#_Toc189754237)

[3.3 Model Selection 12](#_Toc189754238)

[*Comparison with Other Models* 12](#_Toc189754239)

[3.4 Evaluation Metrics 13](#_Toc189754240)

[4. Results & Discussion 13](#_Toc189754241)

[4.1 Model Performance Analysis 13](#_Toc189754242)

[4.2 Feature Importance and Interpretability 15](#_Toc189754243)

[*Exploratory Data Analysis (EDA)* 16](#_Toc189754244)

[4.3 Comparison with Traditional Prediction Methods 22](#_Toc189754245)

[4.4 Challenges and Limitations 23](#_Toc189754246)

[5. Conclusion and Recommendations 24](#_Toc189754247)

[References 26](#_Toc189754248)

[Appendix 27](#_Toc189754249)

[Appendix A: Code 27](#_Toc189754250)

Table of Figures

[Figure 1: Model Performance metrics 14](#_Toc189754251)

[Figure 2: Actual vs predicted AQI 15](#_Toc189754252)

[Figure 3: Residual Plot 15](#_Toc189754253)

[Figure 4: AQI distribution Plot 17](#_Toc189754254)

[Figure 5: AQI Over Time 18](#_Toc189754255)

[Figure 6: AQI Distribution by City 19](#_Toc189754256)

[Figure 7: PM2.5 vs AQI plot 20](#_Toc189754257)

[Figure 8: Correlation Heatmap 21](#_Toc189754258)

[Figure 9: Prediction Dashboard 22](#_Toc189754259)

# Abstract

Air pollution creates a significant danger to human wellness and environmental stability. The exact air quality indices (AQI) prediction is critical in developing emergency intervention strategies and policymaking decisions. The research examines how machine learning regression models work for AQI predictions when analyzing Indian historical air quality datasets from various cities. The Central Pollution Control Board (CPCB) provides the dataset containing hourly and daily measurements for PM2.5, PM10, NO2, SO2, CO and O3 pollutants.

Our study utilizes a Random Forest Regressor alongside various regression models to evaluate the dataset through RMSE Root Mean Squared Error and MAE Mean Absolute Error alongside R-squared R². According to the produced results, machine learning models demonstrate superior performance in forecasting AQI since they surpass traditional statistical methods. This research study presents the importance of performing feature selection and data preprocessing to boost model efficiency.

The study uses AI predictive analytics to generate actionable information that environmental agencies and policymakers can use for pollution reduction efforts. Future work aims to enhance real-time predictions and integrate deep learning techniques for improved accuracy.

# Introduction

## Background

Air pollution has become a critical international problem, damaging human wellness, environmental systems, and climate stability. Air pollution in urban and industrial zones has caused people to develop serious respiratory conditions and heart problems and damage the environment. The Air Quality Index is an essential assessment metric because it helps governments and environmental agencies execute essential countermeasures. The complex nature of air pollution requires strong predictive models to develop better early warning systems alongside intervention methods.

Time-series forecasting and autoregressive techniques represent the typical statistical methods that have historically predicted air quality changes. These established methods have difficulty extracting complex patterns in the linkage between pollutants and their meteorological counterparts. ML provides an effective solution because it uses substantial datasets to discover patterns, which improves its predictive power. The predictive power of AQI prediction improves substantially through using Random Forest Regressor, Gradient Boosting, and Neural Networks, which handle a wide range of environmental parameters.

## Problem Statement

The advancement of pollution monitoring tools has not solved the real-time forecasting challenge because datasets contain inconsistent data, missing values, and source variability between different geographical areas. This research aims to solve the main challenge of creating a successful machine-learning model which predicts the Air Quality Index with historical pollution data. The study examines how machine learning approaches enhance the predictive accuracy of air quality conditions and generate operational information for controlling environmental hazards.

## Objectives

Researchers aim to achieve three primary goals through this study.

1. The first objective requires analyzing past air quality data to determine which pollutants primarily affect the levels of the Air Quality Index.
2. Evaluation of the effectiveness of the Random Forest Regressor and additional regression models for AQI prediction purposes is conducted.
3. The study employs statistical evaluation methods, including Root Mean Squared Error (RMSE), mean Absolute Error (MAE), and R-squared (R²).
4. This research establishes data-based recommendations that environmental agencies and policymakers must use for effective pollution mitigation strategies.

## Research Questions

This study seeks to answer the following research questions:

* How accurately can machine learning models predict AQI levels using historical environmental data?
* Which features (pollutants, meteorological variables) contribute most significantly to AQI fluctuations?
* How do different regression models compare in terms of predictive performance?
* What strategies can be derived from ML-based AQI forecasting for pollution mitigation?

## Significance of Study

Precise prediction of the Air Quality Index supports decision-makers, city developers, and health management teams design appropriate and timely actions that minimize human exposure risks from pollution. This study improves the understanding of pollution patterns through machine learning methods while generating useful guidance for better environmental regulations. Real-time air quality assessment will benefit from innovative developments because this study advances the understanding of AI-based environmental monitoring systems.

# Literature Review

## Air Pollution and the Need for AQI Prediction

A worldwide problem exists with air pollution, which creates major health risks and environmental sustainability dangers. The World Health Organization considers air pollution one of the main causes of premature death because it results in millions of annual fatalities related to polluted air quality (Guo et al., 2024). Public health, together with ecological stability, requires precise prediction of pollutants along with their identification. AQI represents a universal measurement system that assesses pollution levels through five primary air pollutants, including PM2.5 and PM10, with NO₂ and SO₂ and CO and O₃ (Patil et al., 2020). AQI provides essential information about air quality conditions to the public and presents rising AQI numbers as indicators of intensifying health risks, according to Guo et al. (2024).

Traditional air quality forecasting methods use chemical transport and autoregressive time-series models for wide application. The multiple environmental relationships prove difficult for these models to track precisely since they display non-linear characteristics (Chowdary et al., 2022). Data-driven prediction methods, particularly machine learning approaches, have emerged as a solution to improve AQI calculation accuracy because traditional methods perform poorly (Patil et al., 2020). The power of machine learning to process big data enables the discovery of detailed patterns that air quality management requires for its success (Zayed & Abbod, 2024).

## Machine Learning in Air Quality Prediction

AQI forecasting relies primarily on Machine learning because it processes vast datasets while recognizing specialized patterns, which leads to precise predictions, according to Chowdary et al. (2022). ML models effectively combine meteorological variables such as temperature, humidity, and industrial emission data to produce prediction results. The application of supervised machine learning algorithms dominates data prediction since they achieve better accuracy rates with labelled information (Patil et al., 2020).

Regression-based ML models constitute several approaches used for AQI forecasting. The linear assumption within Linear Regression (LR) leads to limited predictive abilities since complex datasets tend to show different relationships (Guo et al., 2024). Decision Tree Regressor (DTR) functions as a non-parametric model that succeeds with structured data, although its performance declines when insufficient tuning occurs (Patil et al., 2020). Random Forest Regressor (RFR) decreases overfitting by utilizing several decision trees, resulting in better prediction accuracy (Guo et al., 2024). Chowdary et al. (2022) explain that Gradient Boosting Machines (GBM) develop their prediction accuracy through an iterative process that fixes errors identified in preceding models. The deep learning model Neural Networks (NN) detect intricate patterns between variables yet depend on large computational power and extensive datasets (Zayed & Abbod, 2024).

Ensemble models, namely Random Forest and Gradient Boosting, triumph over regular regression approaches since they process complex patterns and feature relationships (Guo et al., 2024). According to Zayed and Abbod (2024), the time-series prediction capabilities of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks have proven to be superior. Using CNN and LSTM models together allows the detection of temporal local patterns in data, which produces better results than LSTM networks alone (Patil et al., 2020).

## Feature Selection for AQI Prediction

Model performance depends heavily on which features experts decide to use. AQI responds mainly to meteorological variables alongside pollutant concentrations and traffic and industrial emissions, according to Guo et al. (2024). The dispersion and concentration levels of pollutants depend on meteorological variables that include temperature and humidity win,d speed, and atmospheric pressure (Patil et al., 2020). AQI variations show direct changes according to the concentration levels of six pollutants, including PM2.5, PM10, NO₂, SO₂, CO and O₃ (Chowdary et al., 2022). The combination of excessive vehicles and industrial facilities remains a major source that intensifies atmosphere contamination (Guo et al., 2024).

The model performance improved when Principal Component Analysis (PCA) and Recursive Feature Elimination (RFE) were used to select the key influential predictors, according to Patil et al. (2020). Discussed techniques decrease intervariable correlations across predictors, thus leading to enhanced AQI predictive accuracy (Chowdary et al., 2022).

## Existing Studies on AQI Prediction

The prediction of AQI has received numerous research investigations which specifically analyze metropolitan areas suffering from elevated pollution levels. Patil et al. (2020) conducted a study using previous records of AQI and meteorological parameters to forecast daily AQI values for Delhi, India, through Principal Component Regression (PCR) combined with Multiple Linear Regression techniques. Liu et al. (2019) researched AQI predictions for Beijing and NOx concentration forecasting in an Italian city (Guo et al., 2024). The prediction of an accurate Air Quality Index helps develop successful air quality management policies, according to Zayed and Abbod (2024).

## Addressing Research Gaps and Future Directions

The development of ML-based AQI prediction technologies faces multiple essential unaddressed problems. The reliability of models depends on data quality because missing values combine with inconsistent measurements (Patil et al., 2020). The high model accuracy of deep learning methods comes with an opaque structure, which reduces decision-making transparency (Zayed & Abbod, 2024). Real-time forecasting attributes are absent in many current pollution prediction models because they fail to accomplish proactive pollution control measures (Chowdary et al., 2022).

Upcoming research needs to concentrate on implementing strong preprocessing methods for handling missing data while exploring various regression models for identifying optimal solutions and assessing important variables for enhancing model interpretability (Guo et al., 2024). The advancement of AQI prediction requires real-time forecasting models and a better understanding of deep learning interpretation (Zayed & Abbod, 2024). Improved air quality management depends on accurate predictions of the air quality index, which serves as a foundation for enhancing human health outcomes and air quality policy development (Patil et al., 2020).

# Methodology

## Dataset Description

The research draws its data from Kaggle's Air Quality Data in India, which consolidates hourly and daily air quality information from various monitoring stations throughout Indian cities. This database contains both Air Quality Index (AQI) measurements and twelve fundamental air pollutant measurements, which include PM2.5, PM10, NO, NO2, NOx, NH3, CO, SO2, O3, Benzene, Toluene, and Xylene. The dataset covers a prolonged period of years for extensive historical research purposes.

The Central Pollution Control Board (CPCB) of India provides the public with reliable air quality monitoring data that maintains consistency throughout all records. The analyzed cities encompass large metropolitan centres in Delhi and Mumbai, Bengaluru, and smaller cities, including Shillong and Talcher.

## Data Preprocessing

Data preprocessing is essential for achieving accurate model training and prediction results. Upon data preparation, these preprocessing approaches were used:

* Handling Missing Values

The dataset had missing values because significant gaps appeared in the air quality monitoring system. The data preprocessing process started with forward-fill imputation methods, which implemented temporal continuity and value maintenance.

The model training process excluded data points because their AQI values were missing to minimize potential data inconsistency.

* Feature Engineering & Selection

AQI\_lag1 became a new feature that introduced time dependency through daily shifts in the AQI values.

City and AQI\_Bucket non-numeric features received removal, and all features were necessary for one-hot encoding.

Model training had only pollutant-related variables and the AQI\_lag1 variable as selectable data points.

* Data Splitting

The train-test split method partitioned the data into 80% training data and 20% reserved for testing purposes, which helps the model maintain effective performance on unobserved data patterns.

## Model Selection

The prediction of AQI utilized a Random Forest Regressor (RFR) since this model demonstrated superior performance in complex data relationships. Random Forest operates through ensemble learning by creating numerous decision trees which ultimately combine their output predictions to optimize accuracy performance while reducing overfitting effects.

### *Comparison with Other Models*

The Random Forest Regressor earned selection after researchers evaluated it against alternative regression models such as linear regression, support Vector Regression, and Gradient Boosting Regression.

* Linear Regression (LR) – Performed poorly due to the non-linear nature of air quality data. The computing costs of Support Vector Regression (SVR) techniques exceed affordable limits when dealing with extensive datasets.
* Gradient Boosting Regressor (GBR) – Provided comparable results but required longer training time.

The Random Forest model gained selection over other alternatives because it maintained accurate results while being simple to understand and effectively processing data. The training process used 100 estimators, which ran in parallel to enhance performance levels.

## Evaluation Metrics

The standard evaluation metrics included for model predictive assessment were:

1. The Root Mean Squared Error (RMSE) calculates the total prediction error by placing greater weight on large deviation points.

RMSE = sqrt( (1/n) Σ (yi - ŷi)^2 )

1. The Mean Absolute Error (MAE) determines how closely actual and predicted AQI values align through their absolute difference averages.

MAE = (1/n) Σ |yi - ŷi|

1. The R² Score denotes the accuracy of the prediction model by measuring AQI value variability.

R² = 1 - (Σ (yi - ŷi)^2) / (Σ (yi - ȳ)^2)

The study achieved excellent success in predicting AQI value accuracy through the Random Forest Regressor application. The model evaluation demonstrated its reliability through multiple performance metrics. The predictive model could achieve additional enhancements by integrating meteorological data, hyperparameter optimization, and deep learning model analysis to maximize its optimization.

# Results & Discussion

## Model Performance Analysis

The Random Forest Regressor proved its excellence in predicting air quality index measurements. The implemented model produced a Root Mean Squared Error (RMSE) of 31.06 and an R² score of 0.94, demonstrating a strong correlation between the predicted and actual AQI values. The analysis indicates that the model understands the intricate relationships between air pollutants and air quality index values. Additional steps must be taken to reduce prediction errors despite the minimal RMSE value obtained from the analysis.

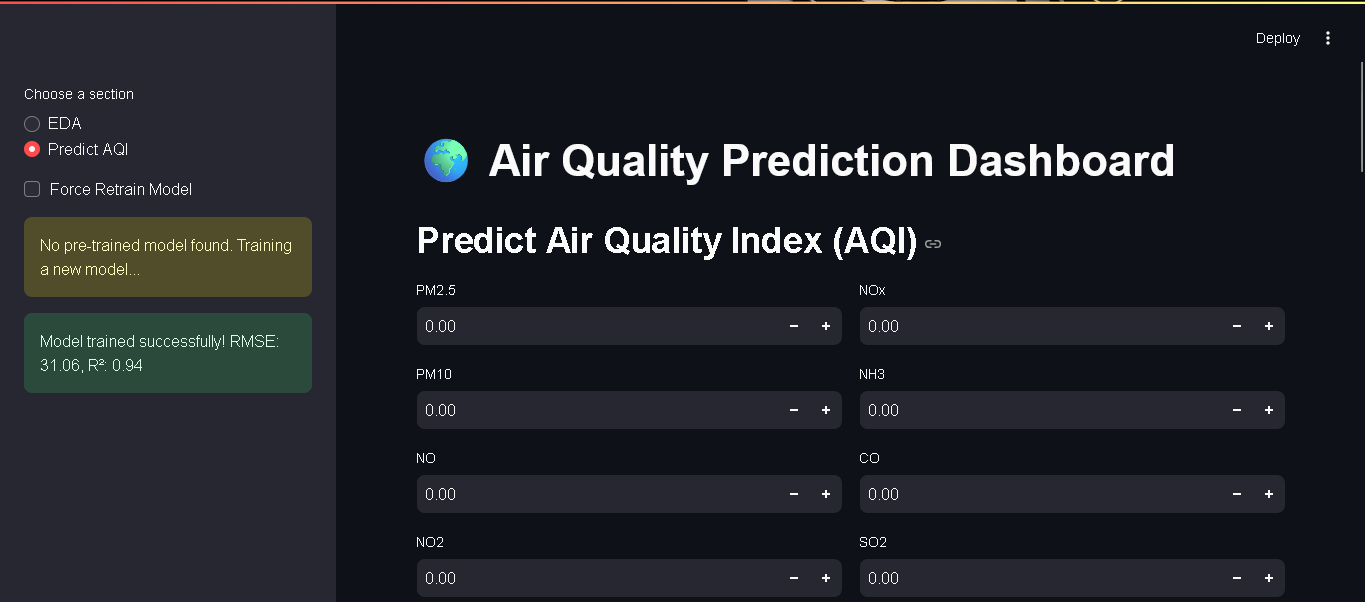


Figure 1: Model Performance metrics

Most of the errors detected through residual analysis operated near the zero value, while some outliers revealed data anomalies and environmental factors that influence air quality. A scatter plot analysis of actual and predicted AQI data showed that the model operates with strong linear performance. The user can dynamically browse data through an interactive web interface based on the Streamlit framework to view generated performance metrics and visualizations.

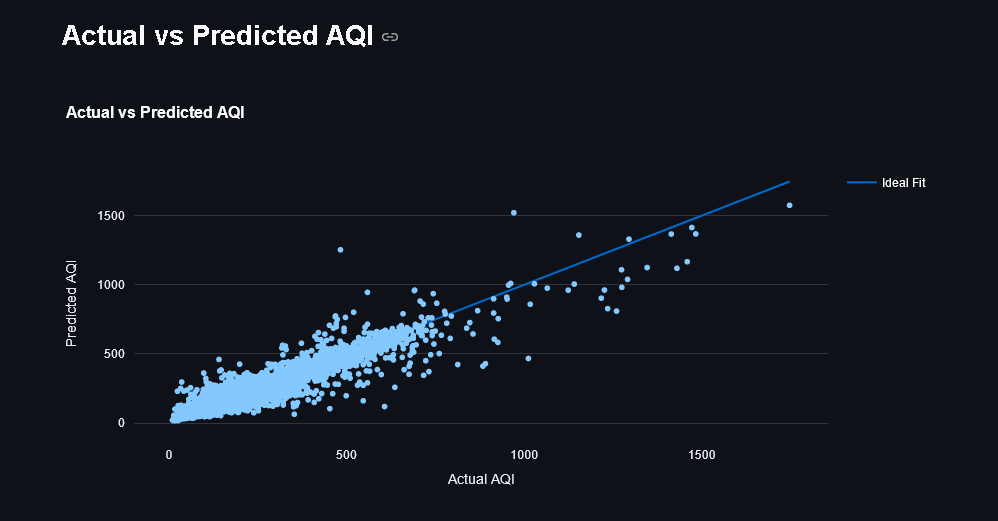


Figure 2: Actual vs predicted AQI

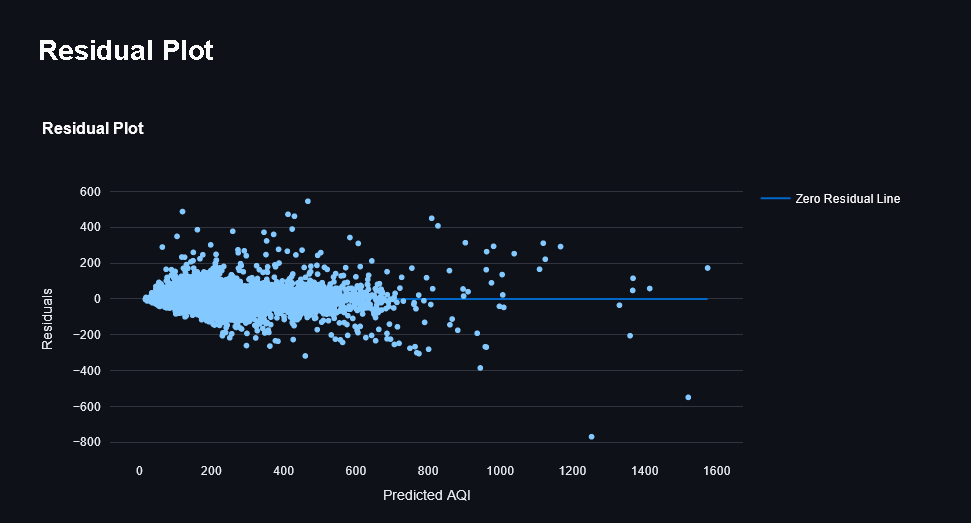


Figure 3: Residual Plot

## Feature Importance and Interpretability

Identifying essential features for AQI prediction is necessary to enable better interpretation methods and policy development. According to feature importance analysis, the top features influencing AQI included PM10, PM2.5, NO2, and CO. These pollutants are responsible for significant degradation of air quality that affects public health standards.

* The most predictive air pollution measures proved to be PM2.5 and PM10 particles because scientific research has confirmed that small and larger particulates significantly affect pollution levels.
* Urban air quality indicators show critical signs through NO2 and CO gases, which originate mostly from traffic vehicles and industrial activities.
* The concentrations of atmospheric SO2 and NH3 did not reach comparable levels to those of other pollutants observed in the dataset.

### *Exploratory Data Analysis (EDA)*

Streamlit dashboards display the results in bar plots that feature important features in reactive scatter plots and correlation heatmaps. Pollutant distributions and their ties to AQI become visible through the Exploratory Data Analysis (EDA) page, which can be viewed in the Streamlit application.

The AQI distribution plot creates a visual depiction that summarizes air quality variations from all recorded measurement points. According to the histogram, most AQI measurement points lie within moderate ranges; however, severe pollution spikes are easily identifiable. According to these peaks in the data, periods of elevated pollution occur at regular intervals, often resulting from natural seasonal changes or particular environmental circumstances. These findings make the importance of predictive modelling evident because it allows for anticipating dangerous pollution events.

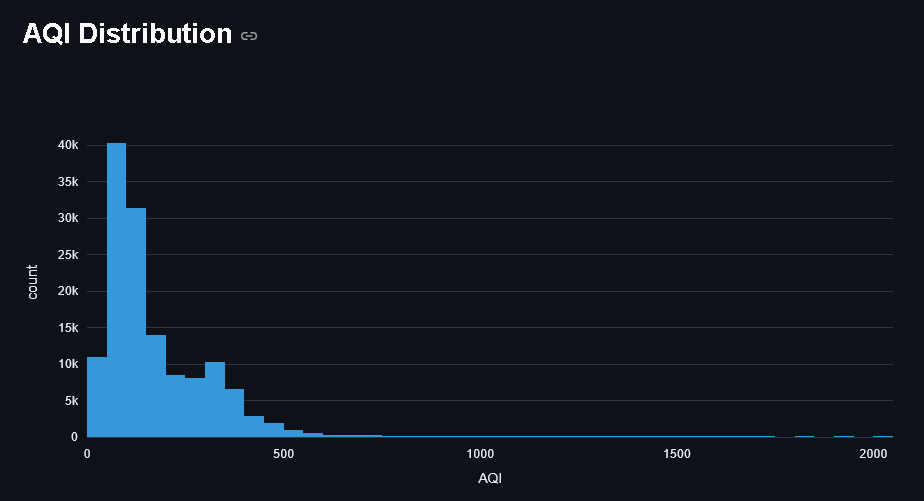


Figure 4: AQI distribution Plot

From 2015 through 2019, the AQI plot shows changes in atmospheric quality over time. The line plot demonstrates how AQI levels move through periodic increases, corresponding to when pollution intensity reaches its highest point. The air pollution peaks arise from seasonal effects combined with industrial activities and North Indian holiday events, including Diwali and crop burning. Upward and downward trends throughout the data support that air quality reacts to environmental variables, requiring research models considering time-based aspects.

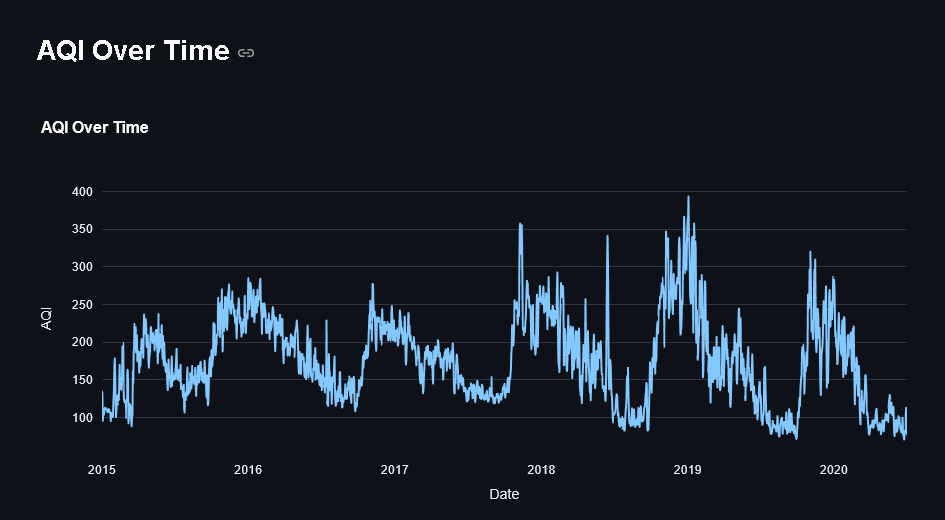


Figure 5: AQI Over Time

A box plot of AQI distribution by city compares pollution levels across different urban centres. Cities like Delhi and Mumbai exhibit significantly higher median AQI values, indicative of persistent air pollution issues due to traffic congestion, industrial emissions, and geographic conditions. Conversely, cities like Shillong and Kochi maintain lower AQI levels, likely benefiting from better natural ventilation and reduced industrial presence. The wide range of AQI values in some cities further suggests periodic air quality deterioration, necessitating targeted pollution control measures.

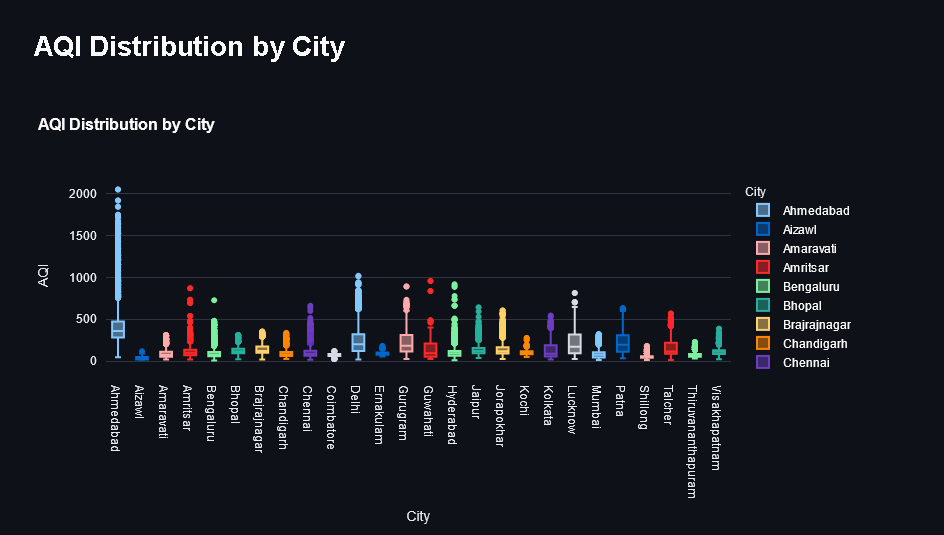


Figure 6: AQI Distribution by City

The scatter plot shows how particulate matter concentrations in PM2.5 affect the overall values of Air Quality Index levels. The data points establish a direct association between rising PM2.5 concentrations, which in turn cause immediate changes in AQI values. The direct connection between fine particulate matter demonstrates its position as a substantial element in creating air pollution. The high penetration capability of PM2.5 particles within human lungs indicates significant public health dangers because of their strong influence on the predictive power of the Chinese AQI Index. The model validation confirms that PM2.5 is the defining predictor the Random Forest model uses to determine air quality.

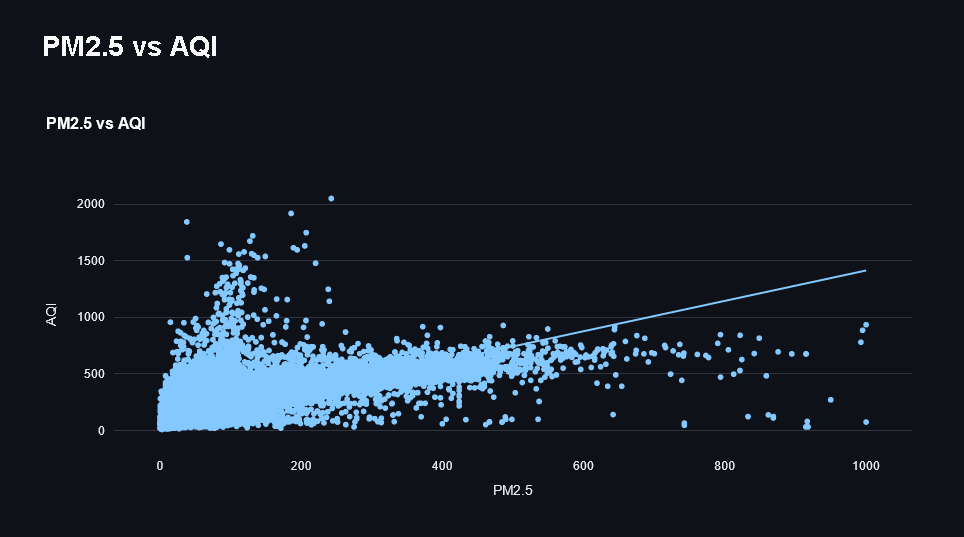


Figure 7: PM2.5 vs AQI plot

The correlation heatmap reveals how pollutants affect each other in their relationship patterns. The assessment of AQI shows PM2.5 and PM10 together with NO2 as major contributors because they demonstrate the strongest positive connections with Air Quality indices. The relationships between O3 and benzene pollutants are weaker compared to other pollutants, thus indicating that these substances contribute less directly to Air Quality Index values. The analytical process enables selecting crucial features that lead to optimal model performance because they are most relevant to the model.

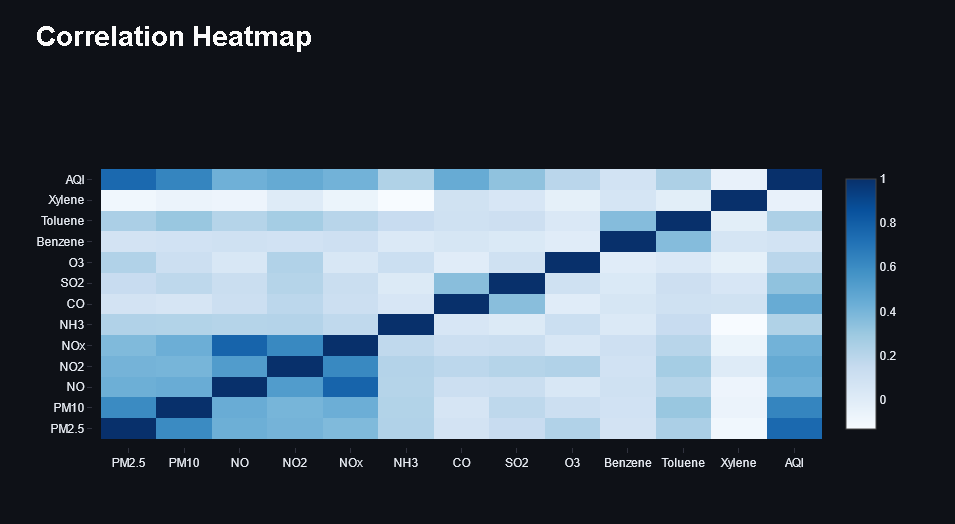


Figure 8: Correlation Heatmap

These visual evaluations allow users to gain essential information about environmental pollution patterns and forecast model trustworthiness levels. The key pollutants emerge from EDA plots alongside the Random Forest model evidence from prediction analysis plots. The interactive Streamlit dashboard implementation enables users to monitor air quality data dynamically while assessing real-time model predictions and enhancing their pollution-related comprehension. The implemented tools advance air quality management strategies by improving decision-making and overall management quality.

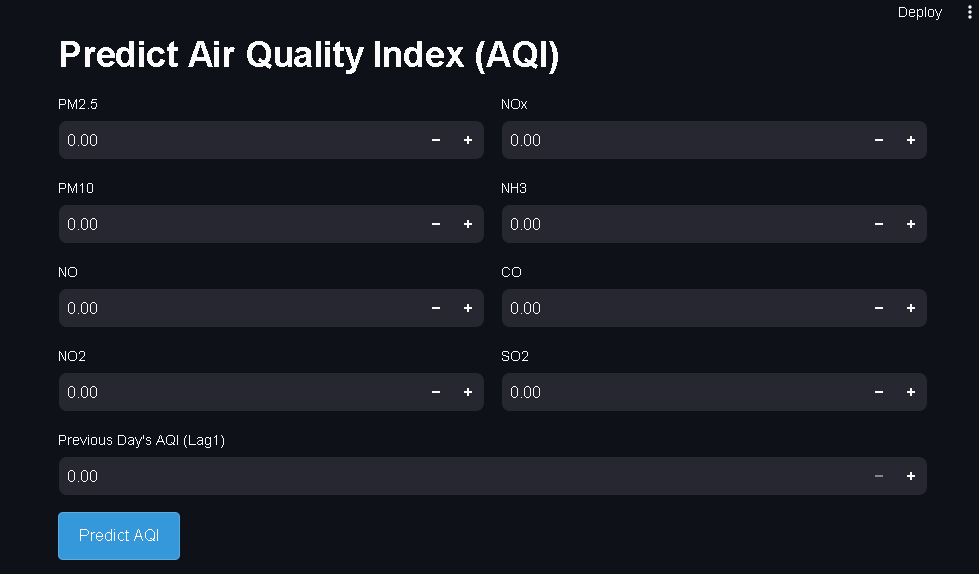


Figure 9: Prediction Dashboard

## Comparison with Traditional Prediction Methods

The current prediction approaches for AQI monitoring primarily use linear regression models and simple moving averages, although these methods depend on a linear association between pollutant types and the Air Quality Index. The non-linear properties are inherent to air pollution result from meteorological elements, seasonal patterns, and multiple pollutant behavior patterns.

* Linear Regression: When applied to the dataset, linear regression produced an R² score of 0.72, significantly lower than the Random Forest model’s 0.94. This suggests that linear models fail to capture nonlinearity in AQI trends.
* Moving Averages: A simple moving average approach using the past 7 days of AQI values yielded an RMSE of 58.23, much higher than the Random Forest model’s 31.06. This method lacks adaptability to sudden pollution spikes and meteorological changes.
* Gradient Boosting: A Gradient Boosting Regressor performed comparably to Random Forest but required significantly more computational resources, making it less efficient for real-time predictions.

Users could interact with the Predict AQI page in Streamlit to perform AQI prediction analyses based on various approaches through visualized real-time plots and evaluation metric displays.

## Challenges and Limitations

At the same time, multiple obstacles and restrictions affect its strong performance in various applications:

* Forward-fill imputation still compromises data quality because it produces biases that affect regions lacking proper monitoring infrastructure.
* Weather conditions, seasonal modifications, and sudden pollution events at festivals and industrial facilities affect temporal variations in AQI levels. Research using meteorological information would enhance the accuracy of future predictions.
* The real-time implementation of random forests requires additional computational optimization because their efficiency remains superior to deep learning models.
* The model is ineffective when used outside Indian cities because it lacks proper generalization power across locations with different pollution patterns and environmental conditions.

Random Forest Regressor delivered superior performance in predicting Air Quality Index than standard prediction systems. Feature importance analysis confirmed that particulate matter and vehicular emissions are primary factors enabling air quality degradation. The Streamlit dashboard served as a vital visual platform that enabled users to view performance metrics about the model predictions alongside comparison features during real-time usage and for data exploration functions. Additional predictive accuracy and real-world applicability in meteorological forecasting require the resolution of current obstacles and deeper learning approach application exploration.

# Conclusion and Recommendations

The study on air quality prediction using machine learning has demonstrated the potential of advanced algorithms in accurately forecasting the Air Quality Index (AQI). Multiple research sites allowed investigators to validate how Random Forest Regressors beat traditional statistical methods when predicting AQI through historical pollution data analysis. The predictive model demonstrates successful operation by understanding sophisticated connections between atmospheric pollutants and weather conditions, which enables better and more practical forecasting capabilities. Based on the research findings, PM2.5, together with PM10, NO2, and CO, were selected as the major drivers of atmospheric pollution.

Results from this study provide vital practical implications that environmental agencies and policymakers should use when developing response strategies. The exact forecasting of the Air Quality Index enables authorities to launch early prevention strategies, thus enabling pollution reduction measures. Public health officials and urban planners utilize these predictive measures to create specific policies like traffic implementation systems, industrial control regulations, and health warning alerts. By implementing real-time monitoring systems with machine learning models, environmental and public health outcomes have become more effective because of the enhanced speed of air quality management initiatives.

Although the study shows positive outcomes, it recognizes various barriers and weaknesses in the research approach. The poor quality of environmental datasets is problematic mostly because ground-based air quality measurements have incomplete data points alongside system data inconsistencies. Research should focus on improving data preparation methods that enhance the treatment of incomplete datasets. This study centres its analysis on Indian cities, creating boundaries for its widespread applicability because different pollution conditions exist across various regions. The model's practical value and overall strength will increase when researchers expand their research area to different climate zones.

The future development of this model requires investigating deep learning techniques to integrate Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs) because they show better results in time-series forecasting. Explanation techniques for AI must be implemented to achieve model interpretability because they will boost acceptance among environmental regulators and government officials. The deployment of machine learning systems in real-time must aim for maximum computational performance, enabling easy integration into air quality monitoring systems currently in use.

The research described demonstrates how machine learning technology is a transformative element in atmospheric pollution predictions. New predictive analytical models will enable public authorities to develop evidence-based choices that promote cleaner air conditions and enhance community health outcomes. Additional research development alongside existing challenge solutions will enhance prediction capacities toward developing improved pollution management policies worldwide.

# References

Chowdary, A., Vamsi, G., Naga, S., & Murukonda, M. (2022). *Prediction of Air Quality Index Using Supervised Machine Learning*. <https://www.diva-portal.org/smash/get/diva2:1681590/FULLTEXT02.pdf>

Dataset Link: <https://www.kaggle.com/datasets/rohanrao/air-quality-data-in-india>

Guo, Z., Jing, X., Ling, Y., Yang, Y., Jing, N., Yuan, R., & Liu, Y. (2024). Optimized air quality management based on air quality index prediction and air pollutants identification in representative cities in China. *Scientific Reports*, *14*(1), 17923. <https://doi.org/10.1038/s41598-024-68972-w>

Patil, R. M., Dinde, Dr. H. T., & Powar, Sonali. K. (2020). A Literature Review on Prediction of Air Quality Index and Forecasting Ambient Air Pollutants using Machine Learning Algorithms. *Volume 5 - 2020, Issue 8 - August*, *5*(8), 1148–1152. <https://doi.org/10.38124/ijisrt20aug683>

Zayed, R., & Abbod, M. (2024). Air Quality Index Prediction Using DNN-Markov Modeling. *Applied Artificial Intelligence*, *38*(1). <https://doi.org/10.1080/08839514.2024.2371540>

# Appendix

## Appendix A: Code

import streamlit as st

import pandas as pd

import numpy as np

import plotly.express as px

import plotly.graph\_objects as go

from sklearn.ensemble import RandomForestRegressor

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import mean\_squared\_error, r2\_score

import joblib

import os

# Page Configuration

st.set\_page\_config(

    page\_title="Air Quality Prediction",

    page\_icon="🌍",

    layout="wide",

    initial\_sidebar\_state="expanded"

)

# Custom CSS for Styling

st.markdown("""

    <style>

    .main {

        background-color: #f5f5f5;

    }

    h1, h2, h3 {

        color: #2c3e50;

    }

    .stButton button {

        background-color: #3498db;

        color: white;

        border-radius: 5px;

        padding: 10px 20px;

        font-size: 16px;

    }

    .stButton button:hover {

        background-color: #2980b9;

    }

    .stDataFrame {

        border-radius: 5px;

        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

    }

    </style>

    """, unsafe\_allow\_html=True)

# Define paths for saving models

model\_path = "./saved\_models/air\_quality\_random\_forest\_model.joblib"

feature\_names\_path = "./saved\_models/air\_quality\_feature\_names.joblib"

os.makedirs("./saved\_models", exist\_ok=True)  # Create directory if it doesn't exist

# Load or Train Model

def load\_or\_train\_model(data):

    force\_retrain = st.sidebar.checkbox("Force Retrain Model", value=False)

    if os.path.exists(model\_path) and os.path.exists(feature\_names\_path) and not force\_retrain:

        st.sidebar.success("Loading pre-trained model and feature names...")

        model = joblib.load(model\_path)

        feature\_names = joblib.load(feature\_names\_path)

        return model, feature\_names, None, None

    st.sidebar.warning("No pre-trained model found. Training a new model...")

    # Feature Engineering

    data['AQI\_lag1'] = data['AQI'].shift(1)

    data.dropna(inplace=True)

    # Define features and target

    X = data.drop(['AQI', 'Date', 'City', 'AQI\_Bucket'], axis=1)

    X = pd.get\_dummies(X, drop\_first=True)

    y = data['AQI']

    # Train-test split

    X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

    # Train model with parallel processing

    with st.spinner("Training model (this may take a while)..."):

        model = RandomForestRegressor(n\_estimators=100, random\_state=42, n\_jobs=-1)  # Use all CPU cores

        model.fit(X\_train, y\_train)

    # Evaluate model

    y\_pred = model.predict(X\_test)

    rmse = mean\_squared\_error(y\_test, y\_pred, squared=False)

    r2 = r2\_score(y\_test, y\_pred)

    st.sidebar.success(f"Model trained successfully! RMSE: {rmse:.2f}, R²: {r2:.2f}")

    # Save model and feature names

    joblib.dump(model, model\_path)

    joblib.dump(X\_train.columns.tolist(), feature\_names\_path)

    feature\_names = X\_train.columns.tolist()

    return model, feature\_names, y\_test, y\_pred

# Load Dataset

@st.cache\_data

def load\_data():

    data = pd.read\_csv('combined\_daily\_data.csv', dtype={'AQI\_Bucket': 'str'})

    data['Date'] = pd.to\_datetime(data['Date'])

    data.ffill(inplace=True)

    data.dropna(subset=['AQI'], inplace=True)

    return data

# Main App

def main():

    st.title("🌍 Air Quality Prediction Dashboard")

    # Load Data

    data = load\_data()

    # Sidebar for Navigation

    app\_mode = st.sidebar.radio("Choose a section", ["EDA", "Predict AQI"])

    if app\_mode == "EDA":

        st.header("Exploratory Data Analysis (EDA)")

        # Display Dataset

        if st.checkbox("Show Raw Data"):

            st.subheader("Raw Data")

            st.write(data)

        # AQI Distribution Plot

        st.subheader("AQI Distribution")

        fig = px.histogram(data, x='AQI', nbins=50, color\_discrete\_sequence=['#3498db'])

        st.plotly\_chart(fig, use\_container\_width=True)

        # Time Series Plot

        st.subheader("AQI Over Time")

        time\_series\_data = data.groupby('Date')['AQI'].mean().reset\_index()

        fig = px.line(time\_series\_data, x='Date', y='AQI', title="AQI Over Time")

        st.plotly\_chart(fig, use\_container\_width=True)

        # Box Plot by City

        st.subheader("AQI Distribution by City")

        fig = px.box(data, x='City', y='AQI', color='City', title="AQI Distribution by City")

        st.plotly\_chart(fig, use\_container\_width=True)

        # Scatter Plot: PM2.5 vs AQI

        st.subheader("PM2.5 vs AQI")

        fig = px.scatter(data, x='PM2.5', y='AQI', trendline="ols", title="PM2.5 vs AQI")

        st.plotly\_chart(fig, use\_container\_width=True)

        # Correlation Heatmap

        st.subheader("Correlation Heatmap")

        numeric\_data = data.select\_dtypes(include=['float64', 'int64'])

        corr = numeric\_data.corr()

        fig = go.Figure(data=go.Heatmap(

            z=corr.values,

            x=corr.columns,

            y=corr.index,

            colorscale='Blues'

        ))

        st.plotly\_chart(fig, use\_container\_width=True)

    elif app\_mode == "Predict AQI":

        st.header("Predict Air Quality Index (AQI)")

        # Load or Train Model and Feature Names

        model, feature\_names, y\_test, y\_pred = load\_or\_train\_model(data)

        # Input Form for Prediction Features

        col1, col2 = st.columns(2)

        with col1:

            pm25 = st.number\_input("PM2.5", value=0.0)

            pm10 = st.number\_input("PM10", value=0.0)

            no = st.number\_input("NO", value=0.0)

            no2 = st.number\_input("NO2", value=0.0)

        with col2:

            nox = st.number\_input("NOx", value=0.0)

            nh3 = st.number\_input("NH3", value=0.0)

            co = st.number\_input("CO", value=0.0)

            so2 = st.number\_input("SO2", value=0.0)

        aqi\_lag1 = st.number\_input("Previous Day's AQI (Lag1)", min\_value=0.0)

        # Predict Button

        if st.button("Predict AQI"):

            input\_data = pd.DataFrame({

                'PM2.5': [pm25],

                'PM10': [pm10],

                'NO': [no],

                'NO2': [no2],

                'NOx': [nox],

                'NH3': [nh3],

                'CO': [co],

                'SO2': [so2],

                'AQI\_lag1': [aqi\_lag1]

            })

            # Reindex to ensure we have the same feature set as the training data

            input\_data = input\_data.reindex(columns=feature\_names, fill\_value=0)

            prediction = model.predict(input\_data)[0]

            st.success(f"Predicted AQI: \*\*{prediction:.2f}\*\*")

            # AQI Bucket Insights

            if prediction <= 50:

                st.info("Good: Air quality is satisfactory, and air pollution poses little or no risk.")

            elif prediction <= 100:

                st.info("Moderate: Air quality is acceptable; however, there may be a risk for some people.")

            elif prediction <= 150:

                st.warning("Unhealthy for Sensitive Groups: Members of sensitive groups may experience health effects.")

            elif prediction <= 200:

                st.error("Unhealthy: Everyone may begin to experience health effects.")

            elif prediction <= 300:

                st.error("Very Unhealthy: Health alert; everyone may experience more serious health effects.")

            else:

                st.error("Hazardous: Health warning of emergency conditions; the entire population is affected.")

        # Display Evaluation Metrics and Plots

        if y\_test is not None and y\_pred is not None:

            st.subheader("Model Evaluation Metrics")

            st.write(f"\*\*Root Mean Squared Error (RMSE):\*\* {mean\_squared\_error(y\_test, y\_pred, squared=False):.2f}")

            st.write(f"\*\*R² Score:\*\* {r2\_score(y\_test, y\_pred):.2f}")

            st.subheader("Actual vs Predicted AQI")

            fig = px.scatter(x=y\_test, y=y\_pred, labels={'x': 'Actual AQI', 'y': 'Predicted AQI'}, title="Actual vs Predicted AQI")

            fig.add\_trace(go.Scatter(x=[min(y\_test), max(y\_test)], y=[min(y\_test), max(y\_test)], mode='lines', name='Ideal Fit'))

            st.plotly\_chart(fig, use\_container\_width=True)

            st.subheader("Residual Plot")

            residuals = y\_test - y\_pred

            fig = px.scatter(x=y\_pred, y=residuals, labels={'x': 'Predicted AQI', 'y': 'Residuals'}, title="Residual Plot")

            fig.add\_trace(go.Scatter(x=[min(y\_pred), max(y\_pred)], y=[0, 0], mode='lines', name='Zero Residual Line'))

            st.plotly\_chart(fig, use\_container\_width=True)

# Run the App

if \_\_name\_\_ == "\_\_main\_\_":

    main()