AQI Predictor project report

1. Introduction

Air pollution has become one of the most serious environmental problems, especially in large cities like Karachi. Monitoring and predicting air quality helps people take safety measures and helps authorities plan environmental policies.

This project focuses on predicting the Air Quality Index (AQI) for Karachi using machine learning techniques. The model uses historical air quality and weather data to forecast air quality for the next three days. The main goal is to automate the full pipeline — from data collection to prediction — and display the results on a Streamlit dashboard that updates automatically through GitHub Actions (CI/CD).

2. Project Overview

The project predicts short-term European AQI using real-time and historical data. It includes several main components:

1. **Feature Pipeline:** Collects and processes raw data, including pollutants like PM2.5, PM10, nitrogen dioxide, ozone, carbon monoxide, sulphur dioxide, and weather features such as temperature and humidity.
2. **Training Pipeline:** Trains and updates a machine learning model daily using the hopsworks.
3. **Streamlit Dashboard:** Displays current AQI, predicted AQI, and a 3-day forecast chart for the city.
4. **CI/CD Automation:** Automatically runs feature and training pipelines using GitHub Actions on a daily schedule.

3. Data Collection

Dataset is collected from open meteo api which is open source

The dataset includes hourly readings from sensors. Each record contains:

- **Air Pollutants:** PM2.5, PM10, NO_2, CO, SO_2, O_3
- **Meteorological Factors:** Temperature, humidity, and wind
- **Target Variable:** European AQI

Data is managed and stored using **Hopsworks Feature Store**, which provides version control and online access to features using hopsworks API .

API keys (HOPSWORKS_API_KEY and HOPSWORKS_PROJECT) are stored securely in **GitHub Secrets**, allowing safe authentication when pipelines run automatically.

4.Feature selection

Before model training, data cleaning steps were applied such as:

- Handling missing or incorrect readings
- Add time-based features

5. Model Training and Evaluation

The model was trained using historical hourly data, split into training and validation sets based on time order. Performance was measured using these metrics:

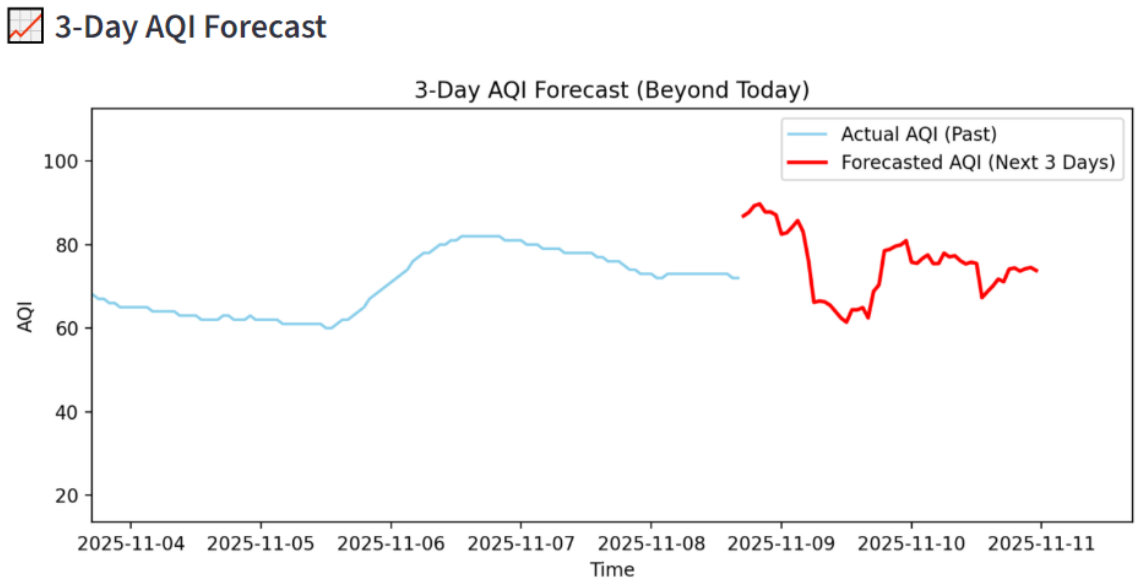
- **RMSE (Root Mean Squared Error):** Measures average prediction error
- **MAE (Mean Absolute Error):** Measures average deviation
- **R² Score:** Explains how well the model captures variance in data

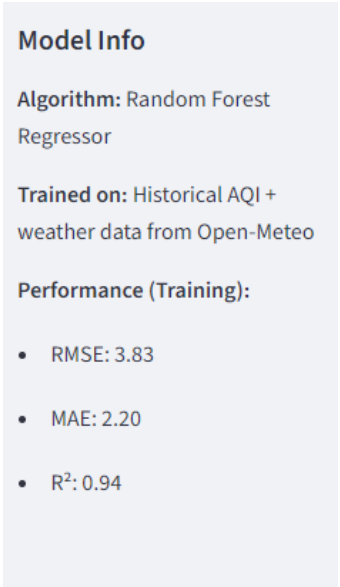
The Random Forest model showed good accuracy and generalization, predicting AQI values close to actual observations.

6. Streamlit Dashboard

A **Streamlit web app** was created to visualize predictions. It includes:

- Current **European AQI** and **Predicted AQI** values
- A clear AQI condition label (e.g., *Good, Moderate, Unhealthy*)
- A **3-day AQI Forecast chart**, showing past data in blue and forecasted values in red
- An alert message that warns users if predicted AQI crosses unhealthy limits
- Model info





7. CI/CD Automation

Automation was handled through **GitHub Actions**:

- The **feature pipeline** runs every hour to update new data.
- The **training pipeline** runs daily at midnight (UTC) to retrain the model.
- After training, the new model file (aqi_random_forest.pkl) is uploaded as an artifact.

This CI/CD setup ensures continuous data updates, automatic training, and consistent deployment without manual effort.

8. Conclusion

This project demonstrates an end-to-end machine learning system for predicting and monitoring air quality using **European AQI standards**. The integration of **Hopsworks**, **Streamlit**, and **GitHub Actions** ensures full automation — from data collection to model deployment.

Key Achievements

- Accurate AQI prediction for the next 3 days using Random Forest
- Automated CI/CD pipeline with hourly data updates and daily training
- Interactive visualization dashboard for real-time monitoring
- Secure API and model management through Hopsworks