10 PEARLS AQI PREDICTION
PROJECT REPORT

By Ayesha Nasir

Tools Used: Python, Streamlit, MongoDB, GitHub Actions, Google Colab, VS Code

Project Overview

The goal of this project is to predict the Air Quality Index (AQI) in Karachi for the next 3 days using a fully serverless, automated pipeline. The system fetches real-time weather and pollutant data, computes derived features, trains multiple ML models, and provides an interactive dashboard for monitoring AQI trends.

Data and Feature Pipeline

Fetched hourly data from OpenWeather Air Pollution API for pollutants (PM2.5, PM10, CO, NO2, SO2, O3) and Open-Meteo Archive API for weather (temperature, humidity, pressure, wind speed).

Merged datasets and computed derived features:

- US EPA AQI (0–500 scale) calculated from raw pollutant concentrations using official breakpoint interpolation formula — PM2.5, PM10, NO2, and O3 sub-indices computed separately; final AQI = maximum sub-index
- AQI lag features: aqi_lag_1, aqi_lag_3, aqi_lag_6, aqi_lag_24
- Time features: hour, day, month, day of week, is_weekend

Stored features in **MongoDB Atlas** (aqi_database → merged_features collection). Hopsworks Feature Store was attempted but could not be used due to persistent connection and authentication errors during implementation.

Historical backfill performed to cover past 3 months (~2,160 hourly records), providing sufficient training data for all five ML models.

Model Training Pipeline

Five models trained using a chronological time-based train-test split (80% train / 20% test) with no data shuffling, to simulate real-world forecasting.

Evaluation metrics: MAE, RMSE, R^2

Model	Test RMSE	Test MAE	Test R^2
Gradient Boosting	9.01	3.54	0.9692
XGBoost	11.04	3.99	0.9538
Random Forest	20.20	4.42	0.8453
Ridge Regression	0.115	0.07	1.0
Lasso Regression	0.149	0.115	1.0

Ridge and Lasso show artificially perfect R^2=1.0 due to partial contamination from old European 1–5 scale data still present in MongoDB during training. These models are excluded from selection.

Gradient Boosting selected as best model based on lowest Test RMSE (9.01) among valid models. It achieved Test R^2=0.9692, meaning it explains 96.9% of AQI variance on unseen data. Gradient Boosting builds trees sequentially, each correcting errors of the previous one, making it well-suited

to the temporal patterns in AQI data.

SHAP (TreeExplainer) applied to the best model for feature importance. Top contributing features: aqi_lag_1, PM2.5, PM10, CO, temperature.

Web Dashboard

Built with **Streamlit**. Features:

- Hourly AQI forecast for 3 days.
- Color-coded AQI alerts.

Interactive plots: actual vs predicted AQI. Summary metrics (MAE, RMSE, R^2).
SHAP feature importance table.

Automation s CI/CD

GitHub Actions automate:
Feature script runs **hourly**.
Training script runs **daily**.
Ensure that the pipeline is **scalable and fully automated**.

Conclusion

Successfully implemented **end-to-end AQI prediction system**. Random Gradient Boost provides most accurate forecasts.
Dashboard shows **real-time alerts and trends**.
Future improvements: deep learning models, multi-city predictions, and cloud deployment.