Air Quality Index (AQI) Prediction System: A Serverless Machine Learning Pipeline

Zohaib Aslam

November 9, 2025

1 Introduction

This project implements an end-to-end serverless system for predicting Air Quality Index (AQI) for the next 3 days using historical weather and pollutant data. The system integrates a feature pipeline, training pipeline, model registry, and web application with automated CI/CD workflows.

2 Exploratory Data Analysis and Feature Engineering

2.1 Data Collection and Cleaning

Historical data spanning one year was collected from OpenWeather API, containing weather variables (temperature, humidity, wind speed) and pollutant concentrations (PM2.5, PM10, CO, NO2, SO2, O3, NH3). Data quality issues in days 356-357 were addressed using linear interpolation, preserving temporal continuity. AQI values were converted from OpenWeather's 1-5 scale to the standard US EPA scale (0-500) for finer granularity.

2.2 Exploratory Analysis

Correlation analysis revealed strong relationships between pollutants and AQI:

- PM2.5 showed the highest correlation (r = 0.97) with US AQI
- Other pollutants exhibited strong positive correlations (0.87-0.94)
- Weather variables showed moderate negative correlations: temperature (r = -0.67), wind speed (r = -0.63), and humidity (r = -0.60)

Time series analysis revealed significant seasonal patterns, with AQI values ranging from 150-400 during winter months (November-February) and dropping to 20-100 during summer (July-September). This indicated the critical importance of temporal features.

Autocorrelation analysis demonstrated strong persistence in AQI values, with significant autocorrelation up to 14 days. However, practical considerations led to selecting lag periods of 1, 3, and 7 days.

2.3 Feature Engineering

Based on EDA insights, the following features were engineered:

Lag Features: For all important predictors (AQI, PM2.5, PM10, CO, NO2, SO2, O3, NH3, temperature, humidity, wind speed), lag features at 1, 3, and 7 days were created. These capture temporal dependencies and pollution persistence patterns.

Temporal Features: Month feature was added to capture strong seasonal variations observed in the data. This enables the model to distinguish between winter and summer pollution patterns.

The final dataset contained 358 rows (after dropping 7 rows due to lag feature creation) with approximately 35 engineered features. All features were stored in Hopsworks Feature Store for version control and automated pipeline access.

3 Model Development and Evaluation

3.1 Models Evaluated

Multiple regression models were trained and evaluated:

- Linear Regression
- Random Forest Regressor
- Gradient Boosting Regressor
- XGBoost Regressor
- Neural Network

3.2 Training Methodology

A chronological train-test split (80:20) was implemented to respect time series nature, with training on earlier dates and testing on future dates. This prevents data leakage and simulates real-world forecasting scenarios.

3.3 Model Performance

Model	**RMSE**	**R^2 Score**
Linear Regression	8.15	0.95
Random Forest Regressor	6.05	0.98
Gradient Boosting Regressor	3.12	0.99
XGB Regressor	4.56	0.98
Neural Network	9.06	0.86

Table 1: Model Performance Comparison

The Gradient Boosting Regressor was selected for deployment due to its excellent R^2 score of 0.99 and RMSE of approximately 3.12 AQI points, indicating strong predictive accuracy across the 0-500 AQI scale.

4 System Architecture and Deployment

4.1 Pipeline Components

- **Feature Pipeline**: Automated via GitHub Actions to run daily, fetching new data from APIs, computing lag and temporal features, and updating the Hopsworks Feature Store.
- **Training Pipeline:** Scheduled daily to fetch updated features, retrain the model, and push the new version to Hopsworks Model Registry with performance metrics.

- **Streamlit Application:** A user-facing web application was built with Streamlit. This app loads the latest trained model, fetches the most recent features from the Hopsworks feature store, and performs inference to display the AQI prediction to the end-user.

5 Results and Limitations

5.1 Achievements

The system successfully predicts AQI with an average error of 3 points, sufficient for practical air quality categorization (Good, Moderate, Unhealthy, etc.). The automated pipeline ensures continuous model improvement as new data accumulates.

5.2 Limitations

- **Weather Forecast Dependency:** The system relies on external weather forecast accuracy. Inaccurate weather predictions directly impact AQI forecasts.
- **Feature Scope:** The model relies only on weather and pollution data. It does not account for external events like traffic patterns, industrial output, or wildfires, which can all cause sudden and dramatic changes in AQI.
- **Local Applicability:** The model is trained on data from a single location and may not generalize well to other geographic regions with different pollution patterns.

6 Conclusion

This project demonstrates a complete serverless MLOps pipeline for AQI prediction, integrating data engineering, feature engineering, model training, and deployment. The system achieves strong predictive performance ($R^2 = 0.99$, RMSE = 3.12) and provides actionable 3-day forecasts through an interactive web interface. Future improvements could include multi-location training, and integration of additional data sources such as traffic patterns and industrial activity.