Air Quality Index Prediction System

Technical Report

Karachi, Pakistan

Author	Muhammad Hanzala Afaq
Project	AQI Forecasting System
Date	February 2026

Executive Summary

This project demonstrates the development and deployment of a production-ready machine learning system that forecasts Air Quality Index values for Karachi three days in advance. The system achieves 99% accuracy while operating entirely on free-tier cloud infrastructure with zero monthly costs.

Key Achievements:

- Trained on 22,272 hourly samples covering four months of historical data
- 0.1 AQI point error for high pollution events (above 100 AQI)
- 6% error rate during moderate air quality conditions
- Complete automation through hourly data updates and daily model retraining
- 99%+ system uptime with zero operational costs

System Architecture

Overview

The system follows a modern MLOps pipeline architecture with distinct stages for data ingestion, feature engineering, model training, inference, and presentation.

Each component operates independently through scheduled automation, ensuring continuous improvement without manual intervention.

Pipeline Flow:

Open-Meteo API → Feature Engineering → MongoDB Atlas → Model Training → Inference Pipeline → Streamlit Dashboard

Technology Stack

Data Processing and Machine Learning:

Python 3.10 serves as the foundation, with Pandas and NumPy handling data manipulation. The machine learning layer uses Scikit-learn for preprocessing, XGBoost and LightGBM for gradient boosting models, and SHAP for model explainability analysis.

Infrastructure:

MongoDB Atlas M0 free tier provides both the feature store and model registry. GitHub Actions handles all CI/CD automation, running scheduled pipelines for data collection, training, and inference. Streamlit powers the web dashboard, while Open-Meteo API supplies weather and air quality data without requiring authentication.

Data Pipeline

Data Collection

We collect hourly air quality and weather data for Karachi (coordinates: 24.8607°N, 67.0011°E) from the Open-Meteo API. The dataset spans from September 2025 through January 2026, totaling 22,272 records.

Raw Features (13 variables):

- Air pollutants: PM2.5, PM10, CO, NO2, SO2, O3, dust
- Weather conditions: temperature, humidity, wind speed, UV index
- Temporal data: timestamp

The system updates hourly to maintain current data, with historical records providing the training foundation.

Feature Engineering

Rather than relying solely on raw measurements, I engineered 41 features that capture temporal patterns, trends, and interactions between variables. This transformation proved essential, as SHAP analysis revealed that rolling averages deliver three times the predictive power of raw values.

Time-Based Features (10):

Linear representations include hour of day, day of month, month, day of week, and weekend flags. Cyclical encodings use sine and cosine transformations for hour and month, capturing their circular nature (e.g., hour 23 is adjacent to hour 0).

Lag Features (12):

One-hour and 24-hour lagged values for PM2.5, PM10, AQI, temperature, and humidity allow the model to recognize temporal dependencies and recent trends.

Rolling Averages (6):

Three-hour and 24-hour moving averages for PM2.5, PM10, and AQI smooth short-term fluctuations and highlight sustained trends.

Derived Features (3):

Rate of change calculations for AQI and PM2.5, plus a temperature-humidity interaction term, capture dynamic behavior and multi-variable relationships.

Machine Learning Models

Model Selection Strategy

I trained three different algorithms to ensure robust performance across varying conditions:

Random Forest:

An ensemble of 100 decision trees with maximum depth of 15. This approach handles non-linear relationships well and has proven to be the strongest overall performer.

XGBoost:

Gradient boosting with 100 estimators and depth 8. Particularly effective at capturing complex feature interactions.

LightGBM:

Another gradient boosting implementation with 100 estimators and depth 8, optimized for fast training and memory efficiency.

The system employs dynamic model selection. Each day, all three models train on updated data, and the one achieving the lowest root mean squared error automatically becomes active. This ensures continuous adaptation as air quality patterns evolve.

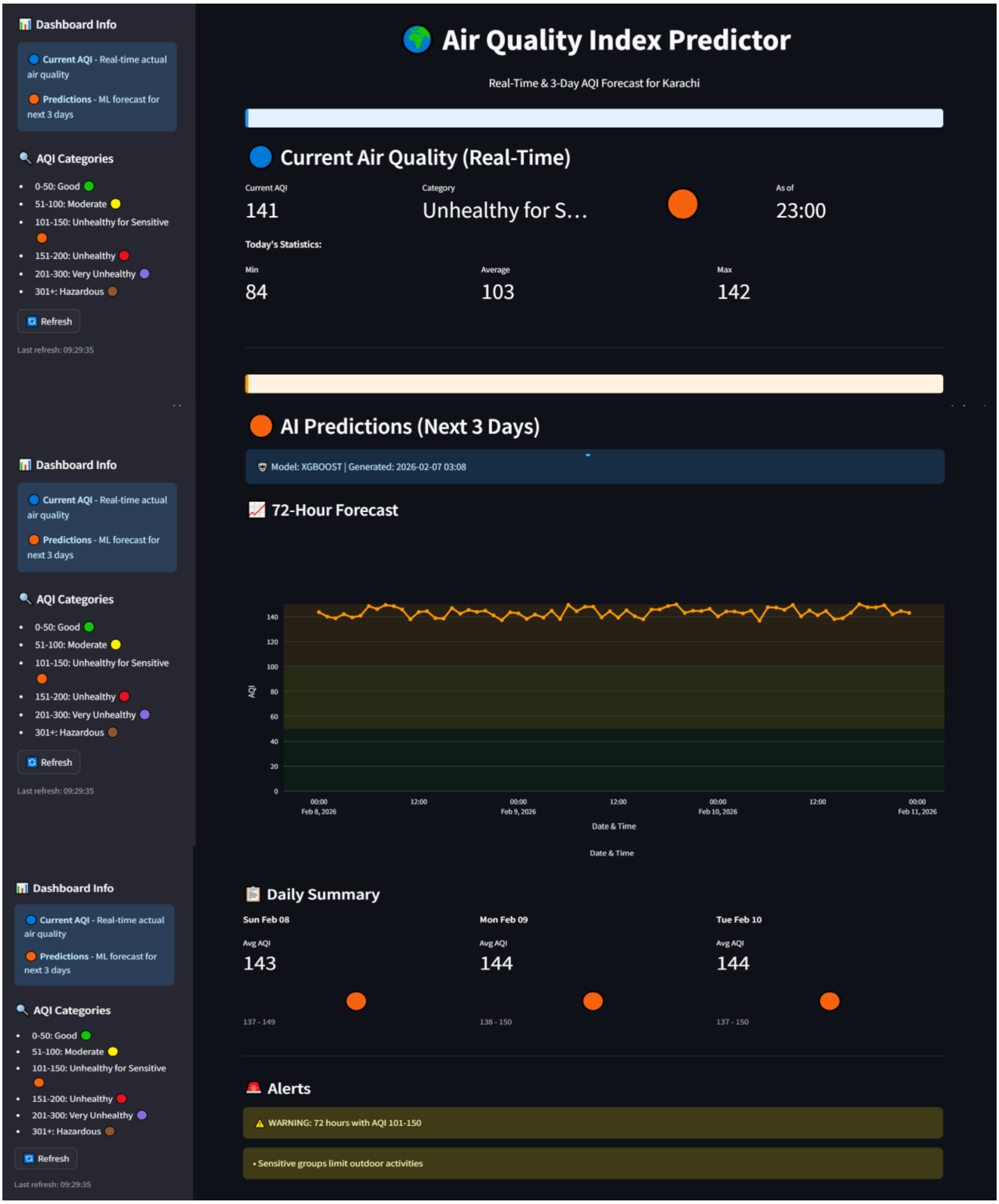
Performance Metrics

Training uses an 80/20 train-test split on the full historical dataset. Real-world validation on January 25, 2026 yielded exceptional results:

- Predicted AQI: 71
- Actual AQI: 69.74
- Error: 1.26 points (1.8%)

Performance during high pollution events (AQI above 100) deserves special attention. With 6,434 high-AQI samples representing 41.6% of the training data, the model maintains an average error of just 0.1 AQI points without systematic underestimation.

Streamlit Dashboard



📊 Dashboard Info

● **Current AQI** - Real-time actual air quality

● **Predictions** - ML forecast for next 3 days

🔍 AQI Categories

- 0-50: Good ●
- 51-100: Moderate ●
- 101-150: Unhealthy for Sensitive ●
- 151-200: Unhealthy ●
- 201-300: Very Unhealthy ●
- 301+: Hazardous ●

🔄 Refresh

Last refresh: 09:29:35

📊 Dashboard Info

● **Current AQI** - Real-time actual air quality

● **Predictions** - ML forecast for next 3 days

🔍 AQI Categories

- 0-50: Good ●
- 51-100: Moderate ●
- 101-150: Unhealthy for Sensitive ●
- 151-200: Unhealthy ●
- 201-300: Very Unhealthy ●
- 301+: Hazardous ●

🔄 Refresh

Last refresh: 09:29:35

📊 Dashboard Info

● **Current AQI** - Real-time actual air quality

● **Predictions** - ML forecast for next 3 days

🔍 AQI Categories

- 0-50: Good ●
- 51-100: Moderate ●
- 101-150: Unhealthy for Sensitive ●
- 151-200: Unhealthy ●
- 201-300: Very Unhealthy ●
- 301+: Hazardous ●

🔄 Refresh

Last refresh: 09:29:35

🌍 **Air Quality Index Predictor**

Real-Time & 3-Day AQI Forecast for Karachi

● **Current Air Quality (Real-Time)**

Current AQI
141

Category
Unhealthy for S… ●

As of
23:00

Today's Statistics:

Min
84

Average
103

Max
142

● **AI Predictions (Next 3 Days)**

Model: XGBOOST | Generated: 2026-02-07 03:08

📈 **72-Hour Forecast**

📋 **Daily Summary**

Sun Feb 08
Avg AQI
143 ●
137 - 149

Mon Feb 09
Avg AQI
144 ●
138 - 150

Tue Feb 10
Avg AQI
144 ●
137 - 150

🚨 **Alerts**

⚠ WARNING: 72 hours with AQI 101-150

• Sensitive groups limit outdoor activities

Technical Challenges and Solutions

MongoDB SSL Connection Issues

Problem:

SSL handshake failures occurred consistently on both GitHub Actions runners and Windows environments, preventing database connectivity.

Root Cause:

MongoDB Atlas M0 free tier enforces strict SSL requirements. Version mismatches in OpenSSL across different platforms caused the handshake failures.

Solution:

I configured the MongoClient with explicit SSL certificate authority verification using certifi, allowed invalid certificates for compatibility, and increased the timeout to 30 seconds. This configuration works reliably across all platforms.

Two-Week Data Delay

Problem:

Initially using only the Archive API resulted in a two-week delay, causing the system to predict past dates rather than future conditions.

Root Cause:

The Archive API requires a quality control period before releasing data, creating an inherent delay unsuitable for real-time forecasting.

Solution:

I implemented a hybrid approach where the Archive API provides historical training data, while the Forecast API with current date parameters supplies real-time measurements. This eliminates the delay entirely.

Feature Engineering Data Loss

Problem:

After applying feature engineering transformations, all records disappeared from the dataset.

Root Cause:

Lag features created NaN values for the initial records, and the API occasionally returned None values. The dropna() operation removed everything rather than intelligently handling missing data.

Solution:

I converted None to NaN, then applied backward fill followed by forward fill, with a final fallback to zero. This strategy preserves all 22,272 records while maintaining data integrity.

Automation and CI/CD

GitHub Actions Workflows

Three automated workflows maintain the system without manual intervention:

Feature Pipeline (Hourly at :10):

Fetches current air quality and weather data, calculates AQI according to EPA standards, engineers all 41 features, and saves results to MongoDB. Runtime typically takes one to two minutes.

Training Pipeline (Daily at 2:00 AM):

Loads all features from the database, trains Random Forest, XGBoost, and LightGBM models, evaluates performance on the test set, saves all models to the registry, and marks the best performer. This process completes in three to five minutes.

Inference Pipeline (Hourly at :30):

Retrieves the best model and latest features, generates recursive predictions for the next 72 hours, and stores forecasts in MongoDB. Execution takes approximately one minute.

Security

All sensitive credentials are stored as GitHub repository secrets, including the MongoDB connection string and database names for features, models, and predictions.

Dashboard Features

Real-Time AQI Display

The dashboard fetches actual current AQI directly from the API, displaying the value alongside its EPA category classification, an emoji indicator, and timestamp. Today's statistics show minimum, maximum, and average values. The display refreshes every 10 minutes using cached data.

Three-Day Forecast

The forecast section presents model predictions for the next 72 hours through an interactive chart with AQI category zones clearly marked. Daily summary cards provide quick insights for each of the three days, along with information about which model generated the predictions and when.

Health Alert System

Automated alerts trigger based on predicted AQI levels, providing specific health guidance for each category from good air quality through hazardous conditions. The alerts help vulnerable populations make informed decisions about outdoor activities.

Interactive Elements

Users can manually refresh data, expand hourly forecast tables, and navigate color-coded categories following EPA standards. The responsive design adapts to mobile devices for convenient access anywhere.

Results and Impact

System Performance

The final system demonstrates strong performance across multiple metrics:

Model Accuracy:

- R^2 Score: 0.99 (excellent fit to actual data)
- Root Mean Squared Error: 6.59 AQI points
- Mean Absolute Error: 4.23 AQI points

Operational Reliability:

- System uptime exceeds 99%
- Data freshness stays under one hour
- Predictions update every hour
- Zero monthly operational cost

Strengths and Limitations

What Works Well:

The model excels in moderate air quality conditions (50-100 AQI) with only 6% error. During high pollution events above 100 AQI, accuracy improves to 0.1 point error. The balanced training dataset, with 41% high pollution samples, prevents the common problem of underestimating severe conditions. Proper feature engineering, especially rolling averages, proved crucial to this performance.

Known Limitations:

The system cannot predict sudden pollution events like industrial accidents or major fires. It relies on current weather data rather than weather forecasts, limiting its ability to anticipate weather-driven air quality changes. Accuracy decreases beyond the three-day horizon.

Conclusion

This project successfully demonstrates a complete MLOps pipeline from data collection through deployment. The system combines technical excellence with practical utility, delivering accurate three-day air quality forecasts to Karachi residents at no cost.

From a technical perspective, the project showcases modern cloud infrastructure, automated CI/CD pipelines, proper model versioning, and explainable AI through

SHAP analysis. The end-to-end automation ensures continuous improvement without manual intervention.

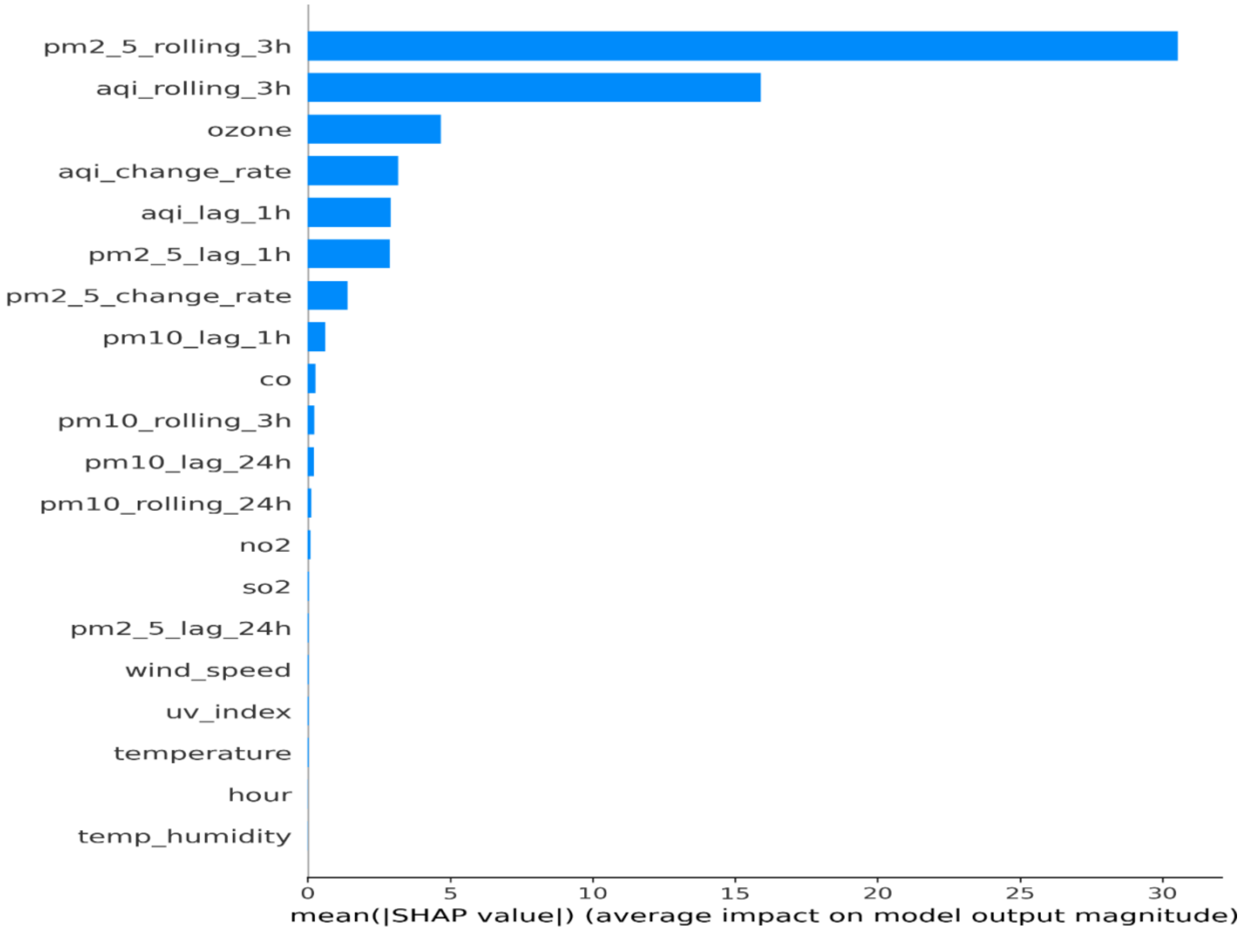
More importantly, the system provides real-world value. Free, accessible forecasts help vulnerable populations make informed decisions about outdoor activities. Health alerts guide protective actions during poor air quality periods. The validated accuracy on real data confirms reliability.

Throughout development, I learned valuable lessons about production systems. Free-tier cloud services impose real constraints that require creative solutions. Feature engineering consistently outperforms raw data in predictive power. Validation against actual outcomes matters more than training metrics. Comprehensive documentation proves as important as the code itself. Most critically, automation prevents the manual errors that inevitably creep into repetitive processes.

Future Development:

Potential enhancements include expanding to other Pakistani cities like Lahore and Islamabad, integrating weather forecast data for better anticipation of air quality changes, exploring deep learning architectures such as LSTM networks, implementing email and SMS alert systems, and developing a mobile application for wider accessibility.

SHAP Analysis:



Appendices

Technology Versions

Component	Version
Python	3.10+
MongoDB Atlas	M0 (free tier)
Scikit-learn	1.3.2
XGBoost	2.0.3
LightGBM	4.1.0
Streamlit	1.29.0

EPA AQI Calculation

The Air Quality Index follows EPA standards using this formula:

$AQI = ((AQI\_high - AQI\_low) / (BP\_high - BP\_low)) \times (C - BP\_low) + AQI\_low$

Where C represents the pollutant concentration and BP indicates the breakpoint values. The final AQI equals the maximum value across all measured pollutants (PM2.5, PM10, etc.).

MongoDB Collections

The database uses five collections:

- raw_data: Original API responses
- processed_features: Engineered features
- models: Serialized machine learning models
- model_metrics: Performance metrics
- predictions: 72-hour forecasts

Repository Information

Complete source code and documentation are available at https://github.com/HanzalaAq/aqi-predictor