PEARLS AQI PREDICTOR

PROJECT REPORT

Contents

Project Overview 1
Data Collection 1
Feature Engineering 2
Model Training and Evaluation 2
Feature Selection with SHAP 3
CI/CD Pipelines 3
Predictions and Alerts 3
Web Dashboard (Streamlit) 3
Technology Stack 4
Future Improvements 4

Project Overview

The Pearls AQI Predictor forecasts the Air Quality Index (AQI) for the next 3 days in Karachi using real-time and historical weather and pollutant data through a serverless ML pipeline.

Data Collection

- **Real-time data:** Fetched from OpenWeatherMap API.
- **Historical data:** Obtained by combining two APIs:
  - AQI and pollutant data from OpenWeatherMap (which had no free API for historical weather data)
  - Historical weather data from Open-Meteo API
- Both datasets were merged into a single historical dataset (historical_data.csv) and stored in Hopsworks feature groups:

    - historical_data (raw historical data)
    - raw_observations (real-time data)

Feature Engineering

- Computed features include:
    - Time-based: Hour, day of week, month, and cyclic transformations (sin/cos) to capture periodic patterns.
    - Lag features: Previous 1, 3, 6, 12, and 24-hour values for AQI and pollutants.
    - Rolling statistics: Mean and standard deviation over 3, 6, 12, 24-hour windows.
    - Targets: AQI for 12, 24, 48, and 72 hours ahead.
- Cleaning process:
    - Replace impossible zeros with NaN (except AQI and wind speed).
    - Forward/backward fill and median imputation for remaining missing values.
- Resulting cleaned features stored in computed_features_historical feature group for training.

Model Training and Evaluation

- Models trained: Random Forest, Ridge Regression, Gradient Boosting.
- Metrics used for evaluation:
    - RMSE (Root Mean Square Error): Measures average prediction error magnitude.
    - MAE (Mean Absolute Error): Measures average absolute difference between predicted and actual AQI.
    - R^2 (Coefficient of Determination): Indicates proportion of variance explained by the model.
- Data split: 80% training, 20% testing.

- Best performing model: Random Forest, stored as randomForest_test_3_model in Hopsworks model registry.

Feature Selection with SHAP

- SHAP used to identify top 25–30 features impacting AQI predictions.
- Model retrained on these top 30 features and stored as randomForest_shap_30_model.
- Same features used for real-time data computation (computed_features_realtime).

CI/CD Pipelines

- Implemented using GitHub Actions:
    1. Real-Time Data Pipeline (Hourly): Fetch real-time data and store in raw_observations.
    2. Daily AQI Model Retraining: Retrain model daily with new data; store improved model versions in registry.
- Feature group retraining_checkpoint tracks last processed datetime for efficient incremental training.

Predictions and Alerts

- Model uses computed real-time features to predict AQI for next 3 days.
- Alerts displayed based on predicted AQI levels.
- **Note**: Current AQI values are on a scale of 1–5 due to free API limitations; future improvements will use exact numeric AQI.

Web Dashboard (Streamlit)

1. Real-Time Page: Displays latest observations and visualization graphs.
2. Model Training Page: Shows computed historical features and model comparison table.
3. Model Insights Page: Displays SHAP feature importance and model metrics.
4. Predict AQI Page: Shows predicted AQI for next 3 days with alerts and visuals.

Technology Stack

- ✓ APIs: OpenWeatherMap, Open-Meteo
- ✓ Python, Scikit-learn, TensorFlow
- ✓ Hopsworks Feature Store
- ✓ GitHub Actions (CI/CD)
- ✓ Streamlit (Dashboard)
- ✓ SHAP (Feature Importance)
- ✓ Git

Future Improvements

- Replace AQI scale 1–5 with exact numeric AQI values.
- Explore deep learning models for improved forecasting accuracy.