AQI Forecasting System Report

This report summarizes the design, implementation, and performance of a fully automated MLOps pipeline for hourly Air Quality Index (AQI) forecasting in Karachi, Pakistan. The system delivers a **96-hour forecast** every hour and visualizes historic and future data on a live Streamlit dashboard.

Feature Development & Data Engineering

This part handles data acquisition, cleaning, feature engineering, and exploratory analysis:

- **Data Acquisition & US-EPA Conversion:** Hourly pollutant data ($PM_{2.5}$, PM_{10}, O_3, NO_2, SO_2, CO) is fetched from **OpenWeatherMap**. Proprietary AQI is converted to US-EPA AQI using EPA breakpoints; overall AQI is the maximum of sub-AQIs.
- **Data Cleaning:** Duplicates removed, negative values clipped, outliers capped at 99.5th percentile, and missing values imputed via forward/backward/median fill.
- **Feature Engineering:** Temporal/cyclical features (hour, month, day, weekend + sine/cosine transforms), interaction/derived features (pm_ratio, total_pm, total_gases, no2_o3_ratio), and time-series features (3-hour rolling means, lagged AQI values) for historical context.
- **Exploratory Data Analysis (EDA):** Statistical summaries, correlation analysis, and visualizations (distributions, trends, scatter plots, heatmaps) were used to guide feature selection for model training.

Feature Store & Data Engineering (Hopsworks)

This part includes manages clean feature preparation, validation, and integration with Hopsworks Feature Store:

- **Feature Selection:** 17 features selected from prior EDA, including pollutant concentrations, AQI lag values, cyclical temporal features, and interaction/rolling metrics.
- **Data Cleaning & Validation:** Datetime parsing, missing values filled, invalid timestamps dropped, numeric columns cast to float64, NaN/inf replaced with 0. Deduplication ensures unique timestamps.
- **Hopsworks Integration:** Cleaned data uploaded as a versioned feature group (aqi_features v4), including an auto-generated id primary key. Feature group can be listed, deleted, and re-uploaded safely.
- **Data Fetch & Verification:** Features fetched for training or analysis, with optional date filtering. Local CSV and Hopsworks data compared to ensure consistency (max numeric difference < tolerance).
- **Training Data Ready:** Validated dataset is fully prepared for downstream AQI model training.

Model Training, Evaluation, and Checkpointing

The training dataset consisted of 17,521 samples with 17 features, fetched from the **Hopsworks** Feature Store.

- **Model Suite:** XGBoost, LightGBM, CatBoost, RandomForest, GradientBoosting, Ridge, and Linear Regression.
- **Time-Series Split:** The data was split **chronologically** (70% Train, 10% Validation, 20% Test) to accurately simulate a real-world deployment scenario and prevent **data leakage**.
- **Hyperparameter Tuning & CV:** Models were tuned using optimized parameters (e.g., restricted max_depth, subsample, colsample, and regularization for tree models) and evaluated using **TimeSeriesSplit (n_splits=5)** cross-validation on the training data. The model with the **lowest CV-MAE** was chosen as the best_model.

1. Training Performance and Overfitting Analysis

Multiple models were trained and evaluated using cross-validation (CV), as well as train, validation, and test splits. The training results show:

Model	Train MAE	Val MAE	Test MAE	Train R^2	Val R^2	Test R^2
XGBoost	1.121	2.925	3.446	1.000	0.953	0.975
LightGBM	0.966	3.146	3.646	1.000	0.942	0.975
CatBoost	1.947	3.718	4.114	0.999	0.929	0.968
RandomForest	1.162	5.779	5.632	0.999	0.893	0.952
GradientBoosting	1.400	3.126	4.288	0.999	0.934	0.956
Ridge	8.999	7.070	10.252	0.986	0.901	0.857
Linear	8.676	6.678	8.850	0.987	0.907	0.887

- Tree-based models (XGBoost, LightGBM, CatBoost, RandomForest, GradientBoosting) achieved very high training accuracy ($R^2 \approx 0.999–1.000$) but maintained strong performance on validation and test sets.
- Linear models (Ridge, Linear Regression) performed worse in terms of error metrics and R^2 values, indicating they were underfitting the non-linear patterns in the dataset.
- Despite extremely high train accuracy for tree-based models, the difference between train and validation/test MAE is modest, suggesting **no severe overfitting**. The models generalize well on unseen data.

2. Data Drift Consideration

The historical best cross-validation MAE (2.054 for XGBoost) was slightly better than the current run (2.060 for XGBoost). The small difference in CV MAE indicates that the dataset has not drifted significantly since the previous training runs.

- **Historical Best Test MAE:** 3.882
- **Current Run Test MAE:** 3.446

The current run achieved better test MAE than the historical best, further supporting that the models are robust and data drift is minimal.

3. Model Selection

- Based on cross-validation MAE and test set performance, **XGBoost** was selected as the primary model.
- Gradient Boosting and LightGBM also showed strong performance but were slightly less optimal compared to XGBoost in terms of CV MAE.
- Tree-based ensemble models consistently outperformed linear models, which confirms the non-linear nature of AQI prediction.

4. Best Checkpoint Model

- The **best checkpoint model** (historical best) is **XGBoost** with:
    - CV MAE: 2.054
    - Test MAE: 3.882
- The current run did not surpass this historical best in CV MAE but achieved slightly better test MAE (3.446), indicating that the checkpoint is still valid for deployment.

5. Forecasting Performance

For the 96-hour AQI forecast:

- XGBoost produced MAE = 1.615 and R^2 = 0.833 on forecasted values.
- Gradient Boosting achieved the best R^2 (0.927) but slightly higher MAE for forecast evaluation.
- Overall, tree-based models provide robust predictions with low errors and are suitable for short-term AQI forecasting.

6. Continuous Data Shift Management
Environmental time-series data can change due to factors like weather anomalies or policy shifts. The system mitigates this by:

- **Daily Retraining:** Automated daily training updates the model with the latest data, ensuring adaptation to new patterns.
- **Short Forecast Horizon:** Limiting predictions to 96 hours keeps forecasts within a period where data distribution is relatively stable.

Automated CI/CD Pipeline (GitHub Actions)

The MLOps workflow is fully automated via three chained GitHub Actions:

1. **fetch_data.yml:** Hourly/manual trigger; runs fetch_data.py, pulls data, applies US-EPA conversion, commits data, and dispatches data-updated event.
2. **eda.yml:** Triggered by data-updated or manually; runs eda.py for cleaning, feature engineering, selection, uploads features to Hopsworks Feature Store, commits artifacts, and dispatches eda-completed event.
3. **train.yml:** Hourly/manual trigger; runs training.py to retrain models, checkpoint the best model, generate 96-hour forecasts, commit models, metrics, forecasts, comparison plots, and upload artifacts.

Web Application Dashboard (Streamlit Cloud) AQI Prediction Dashboard · Streamlit

A dynamic Streamlit dashboard provides transparent access to system outputs.

- **Deployment:** app.py is deployed on Streamlit Cloud, linked to GitHub for automatic containerization and hourly updates via CI/CD commits.
- **Key Features:**
    - **Future Predictions:** Interactive selection of models (default: best checkpoint) with 96-hour Actual vs. Predicted AQI line charts.
    - **Historical Data:** Zoomable 30-day time-series plots for AQI and pollutants for EDA.
    - **Forecast Comparison:** Bar charts of model metrics (MAE, R^2) for evaluating model drift and performance shifts.