Project Report

Pearls AQI Predictor

PROJECT NAME: Automated MLOps for Air Quality: Pearls AQI Predictor

AUTHOR: Shehraz Sarwar

PROGRAM: 10Pearls Shine Internship (Cohort 7)

TRACK: Data Science

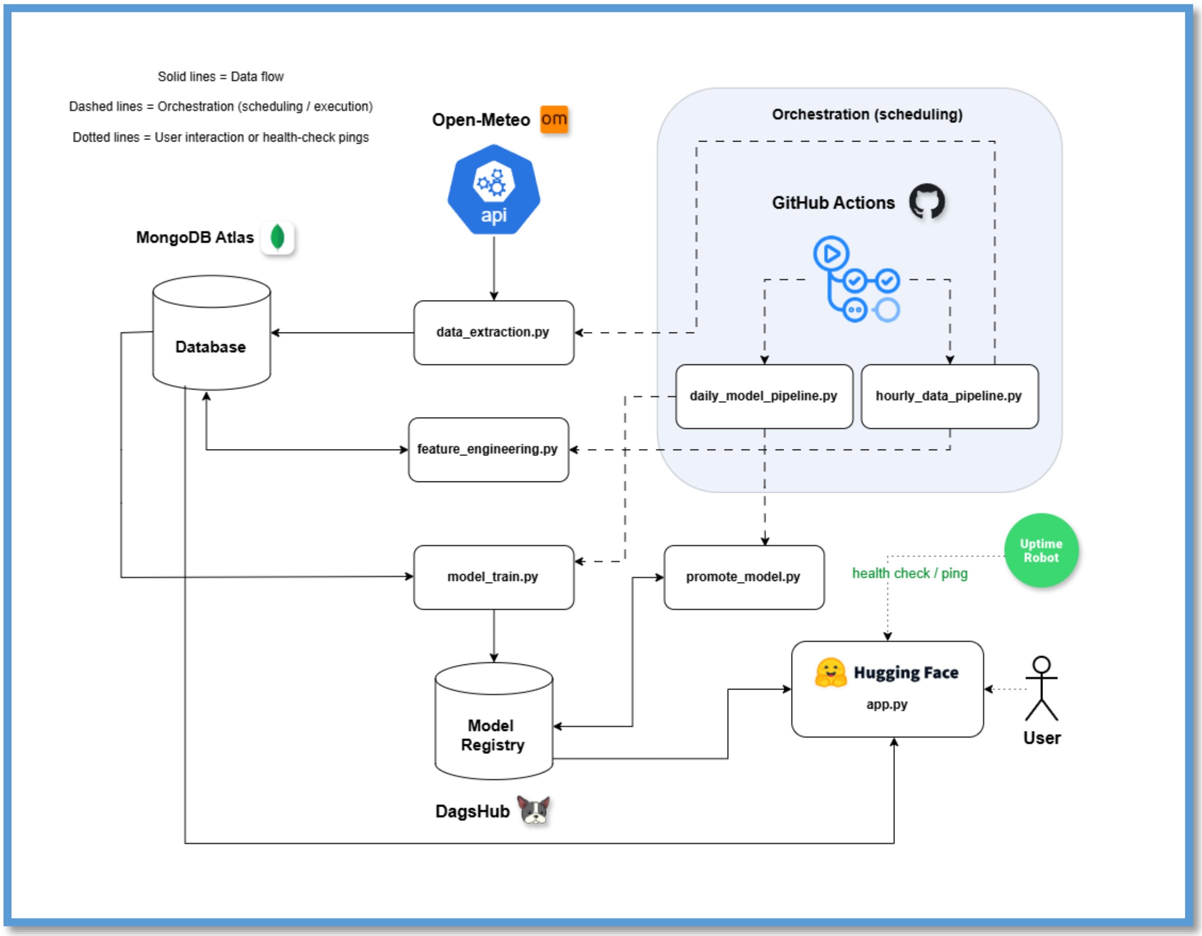
1. Executive Summary

The **Pearls AQI Predictor** is an end-to-end MLOps ecosystem engineered to provide high-fidelity PM2.5 forecasts for Karachi, Pakistan. Developed as part of the **10Pearls Shine (Cohort 7)** program, the system delivers 24, 48, and 72-hour air quality horizons by integrating serverless orchestration, a real-time feature store, and automated model governance.

The architecture is built on a **"Champion vs. Challenger"** deployment strategy, ensuring the system continuously evolves. By utilizing **GitHub Actions** for orchestration and **MongoDB Atlas** as a time-series feature store with custom TTL (Time-To-Live) logic, the project maintains a "lean" production environment that automatically handles data retention and storage constraints.

Key technical milestones include the migration to **Open-Meteo** for unrestricted historical data access, the implementation of a dual-collection database architecture for transparent data governance, and a reliable 24/7 deployment on **Hugging Face Spaces** monitored by **UptimeRobot**. This project demonstrates a cost-effective, enterprise-ready approach to machine learning that prioritizes smart architectural choices over high infrastructure spend.

2. System Architecture & Workflow



The Pearls AQI Predictor is built on a **decoupled, event-driven architecture** orchestrated by **GitHub Actions**. This design separates data concerns from model concerns, ensuring that a failure in data ingestion does not interrupt the prediction service.

2.1 The Data Plane (Hourly Orchestration)

The system maintains a real-time feature store to ensure the "Champion" model always has access to the most recent environmental context.

- **Automated Ingestion:** A specialized data_extraction.py script performs incremental fetches from the **Open-Meteo API**, focusing on high-resolution atmospheric variables for Karachi.
- **Dynamic Feature Store:** Extracted data is upserted into **MongoDB Atlas**. This acts as the project’s "Source of Truth," providing a centralized repository for both raw and processed vectors.

- **Feature Engineering Pipeline:** The system automatically transforms raw API responses into model-ready features, including rolling statistical windows, time-lagged variables, and cyclical temporal encodings to capture seasonal AQI trends.

2.2 The Model Plane (Daily Lifecycle)

To combat model drift and ensure peak accuracy, the system executes a daily **"Champion vs. Challenger"** competition.

- **Parallel Model Competition:** The model_train.py script initiates a multi-model tournament, training **XGBoost, LightGBM, and Random Forest** architectures simultaneously to identify the optimal regressor for the current data distribution.
- **Immutable Experiment Tracking:** Every training run, including hyperparameters and loss metrics, is logged to a **DagsHub-hosted MLflow** registry. This ensures 100% auditability and reproducibility.
- **Automated Promotion Logic:** The promote_model.py script acts as an automated gatekeeper. A "Challenger" model is only promoted to production if:
    1. It demonstrates a statistically significant improvement in error metrics over the current "Champion."
    2. The current production model exceeds a **3-day age threshold**, forcing a refresh to prevent staleness.

2.3 Deployment & Resilience

The final "Champion" model is served via a **Hugging Face Spaces** web application. To bypass free-tier "sleep" cycles and ensure 24/7 reliability, an external **UptimeRobot** heartbeat pings the application endpoint every five minutes, keeping the predictor "warm" for end-users.

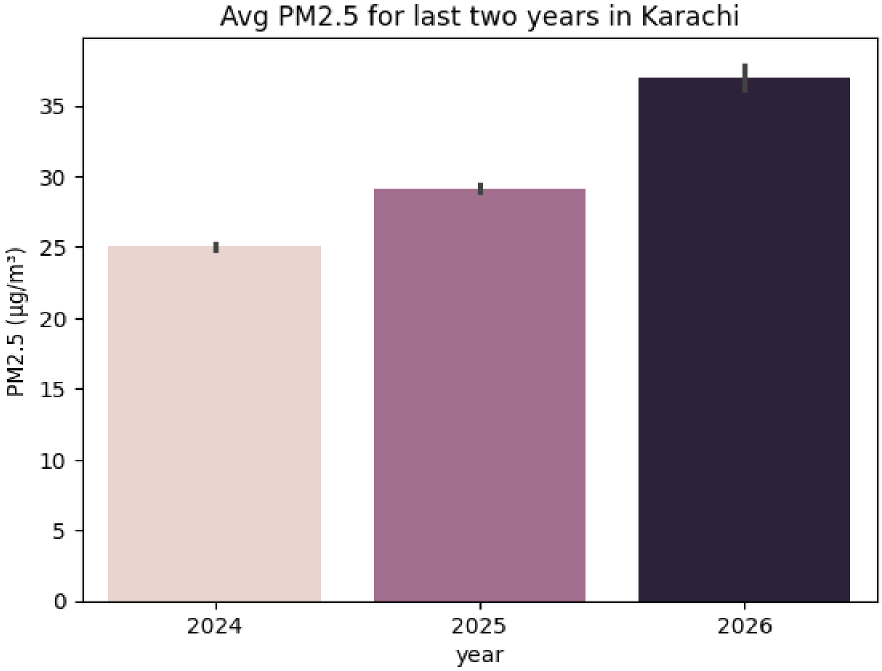
3. Technical Stack

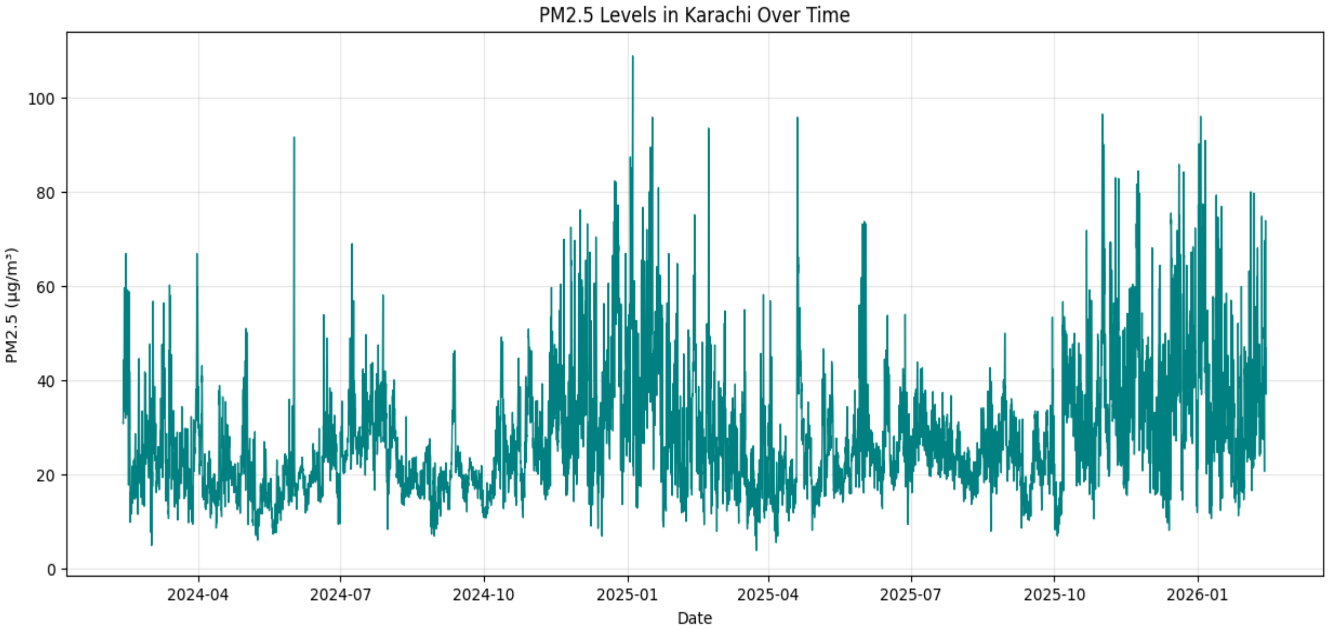
Component	Technology
Database/Feature Store	MongoDB Atlas
Orchestration	GitHub Actions
Tracking & Registry	MLflow / DagsHub
Modeling	Scikit-learn, XGBoost, LightGBM
Deployment	Hugging Face Spaces / Streamlit
Monitoring	UptimeRobot

4. Exploratory Data Analysis (EDA)

The data exploration phase reveals critical insights into Karachi's atmospheric health, establishing the statistical foundation for the predictive models. This analysis focuses on PM2.5 trends, feature correlations, and historical distributions.

4.1 PM2.5 Temporal Trends (2024–2026)

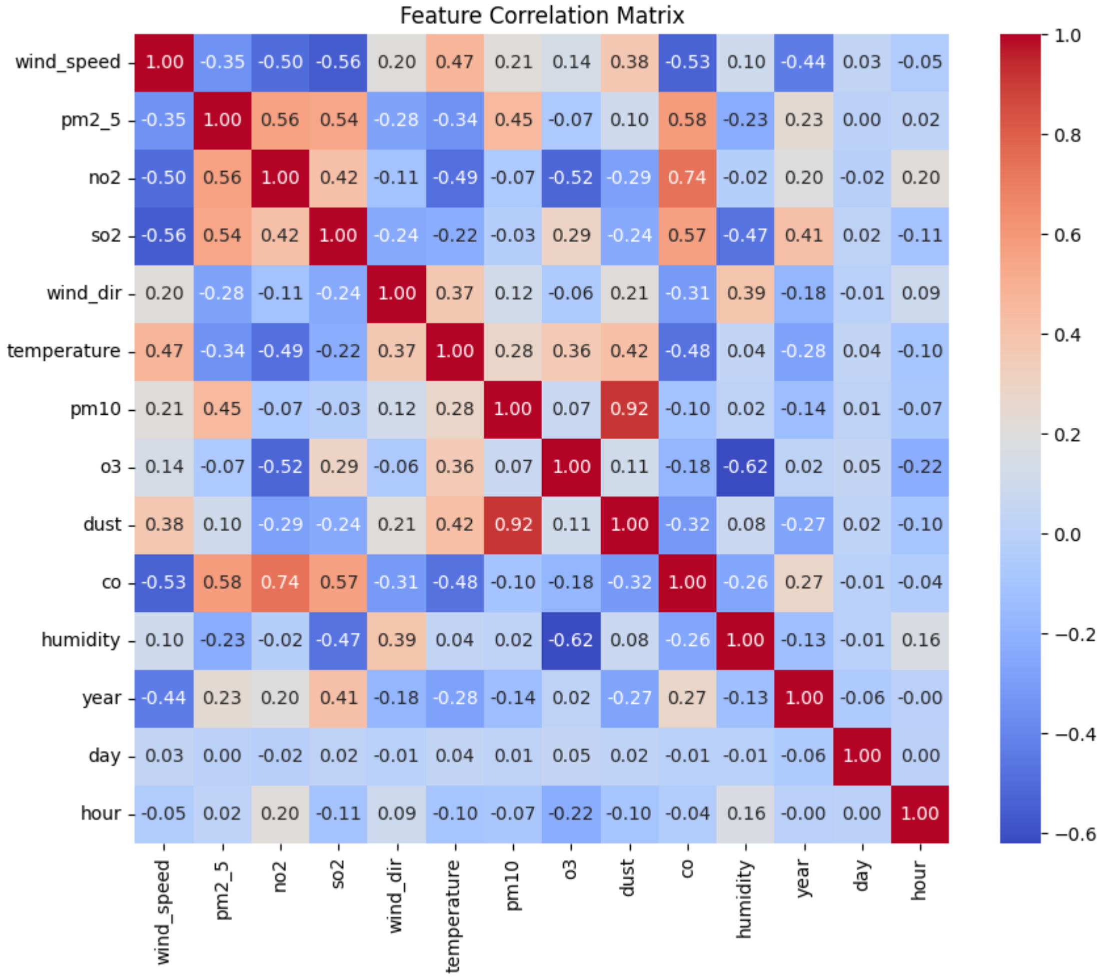




Historical data indicates a significant upward trend in air pollution levels in Karachi.

- Long-term Growth: The average annual PM2.5 concentration has risen from approximately 25 µg/m³ in 2024 to over 35 µg/m³ by early 2026
- Volatility: Time-series analysis shows high-frequency spikes, with peak concentrations occasionally exceeding 100 µg/m³ during seasonal shifts.

4.2 Correlation Analysis & Predictor Selection

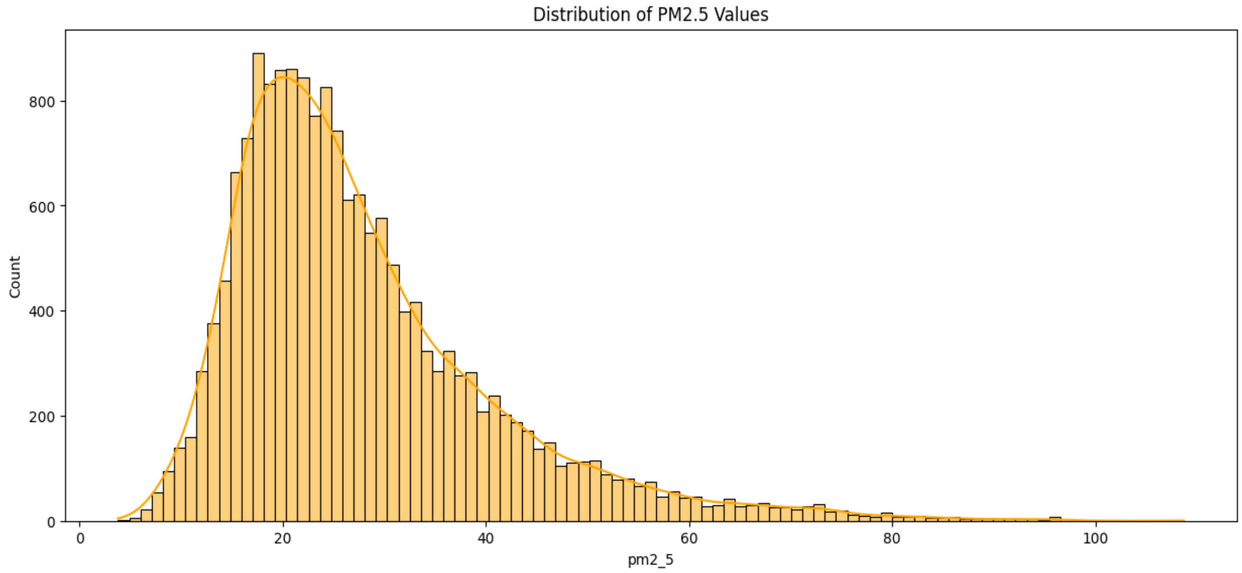


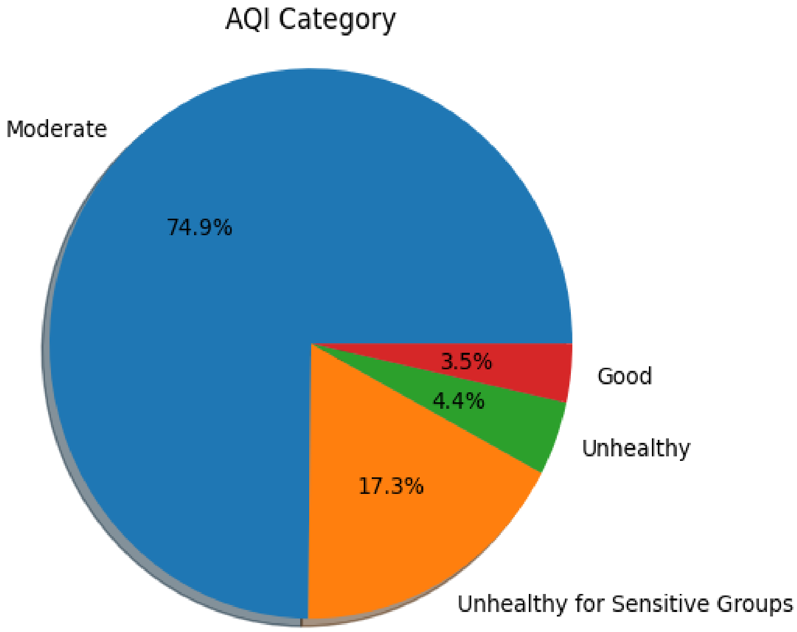
Understanding the relationship between pollutants and weather variables is essential for feature selection.

- Dominant Predictors: Carbon Monoxide (CO) emerged as the strongest predictor for PM2.5 with a correlation coefficient of 0.58, followed closely by Nitrogen Dioxide (NO2) at 0.56 and Sulfur Dioxide (SO2) at 0.54.

- Inverse Relationships: Meteorological factors such as wind speed (-0.35) and temperature (-0.34) show a negative correlation with PM2.5, suggesting that higher wind speeds and temperatures aid in pollutant dispersion.

4.3 Distribution & AQI Classification





The dataset exhibits a right-skewed distribution, reflecting the frequent occurrence of "Moderate" to "Unhealthy" air quality events.

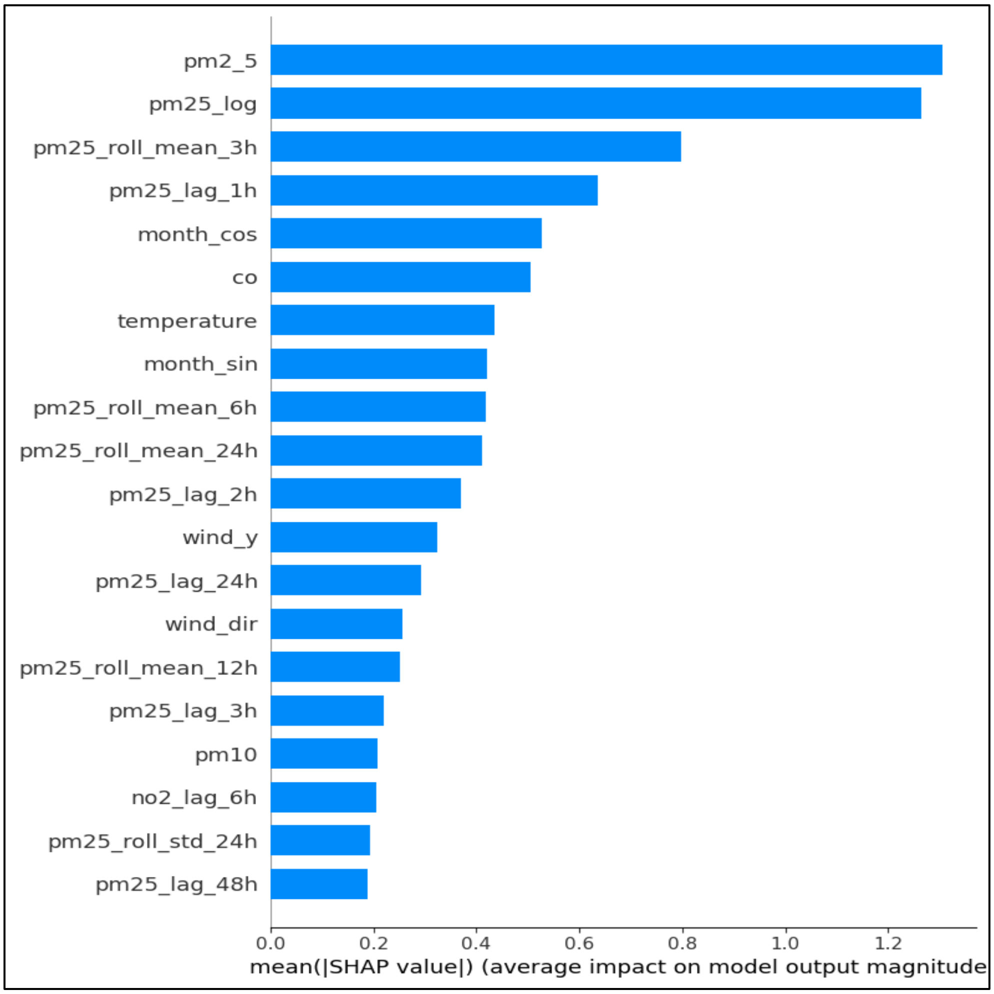
- Concentration Density: Most PM2.5 readings cluster around the 20 µg/m³ mark, but the distribution has a long tail extending beyond 100 µg/m³.

- AQI Composition:
    - 74.9% of the recorded time falls under the Moderate category.
    - 17.3% is classified as Unhealthy for Sensitive Groups.
    - Only 3.5% of the data reflects good air quality, highlighting the urgency for predictive monitoring.

5. Model Interpretability (SHAP Analysis)

To ensure the Pearls AQI Predictor operates as a "glass box" rather than a "black box," **SHAP (SHapley Additive exPlanations)** was utilized. This analysis quantifies the impact of each feature on the final PM2.5 prediction, providing both global transparency and local accountability for the model's outputs.

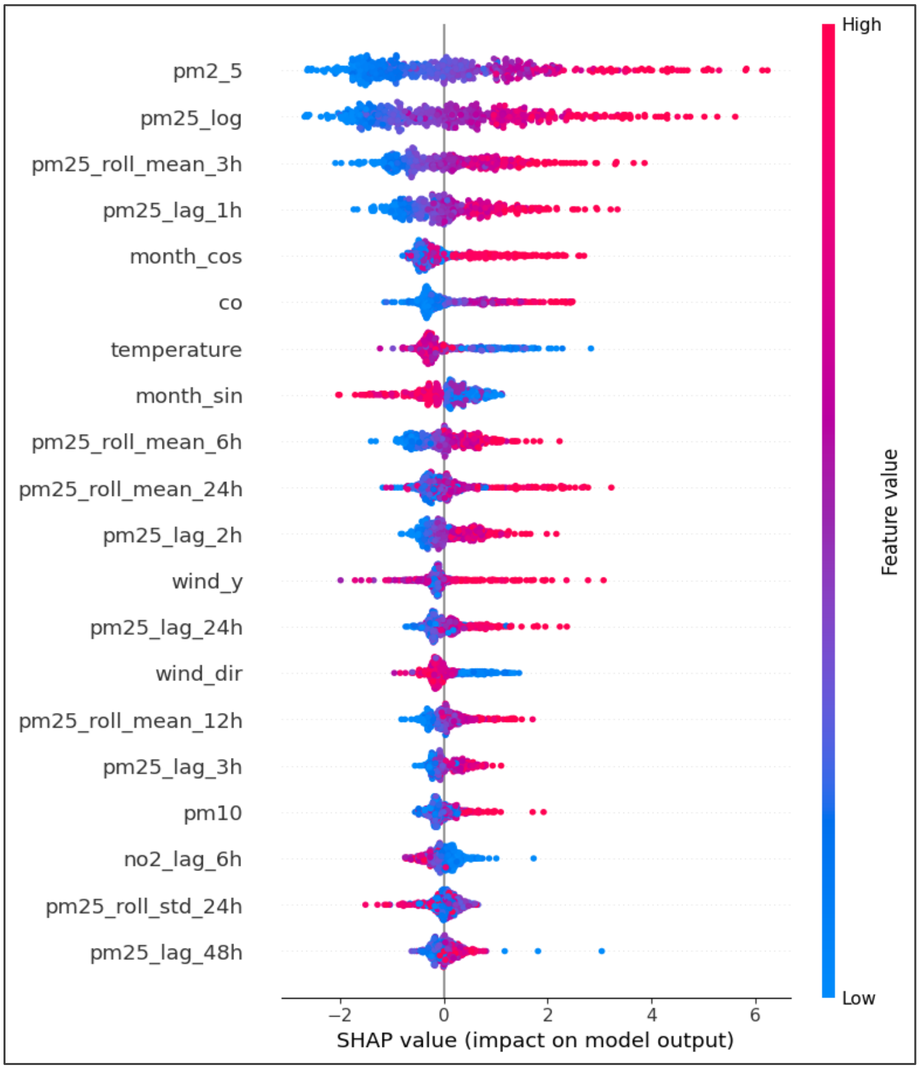
5.1 Primary Feature Drivers



The SHAP summary reveals a hierarchy of influence dominated by temporal continuity and chemical precursors:

- **Autoregressive Power**: The most critical predictors are the current **pm2_5** levels and its **log-transformed** counterpart, indicating that recent air quality state is the strongest indicator of near-future conditions.
- **Rolling Context**: The **3-hour rolling mean (**pm25_roll_mean_3h**)** and **1-hour lag (**pm25_lag_1h**)** carry significant weight, allowing the model to capture immediate momentum in pollution spikes.
- **Chemical Tracers**: **Carbon Monoxide (CO)** stands out as the most influential non-PM2.5 pollutant. Its high SHAP value confirms the EDA findings that combustion-related tracers are vital for accurate PM2.5 forecasting.

5.2 Feature Dynamics and Sensitivity



The SHAP bee-swarm plot provides a deeper look at how specific feature values shift the prediction:

- **Positive Correlation**: High values (red) for **CO**, **pm2_5**, and **pm25_roll_mean_3h** correspond to significantly higher SHAP values, consistently pushing the model's AQI forecast upward.
- **Seasonal & Meteorological Influence**: **Temperature** and **cyclical month encodings (**month_cos**)** show distinct clusters. Lower temperatures (blue) often correlate with higher pollution SHAP values, likely capturing "winter smog" effects where colder, denser air traps particulates near the ground.
- **Atmospheric Dispersion**: Wind-related features (**wind_y**, **wind_dir**) show that specific wind vectors actively reduce the predicted PM2.5 concentration, representing the natural dispersion of pollutants.

6. Engineering Challenges & Technical Mitigations

6.1 Data Sourcing: The Search for a Truly Open API

- **The Challenge:** Most weather and air quality APIs (like OpenWeatherMap or Tomorrow.io) impose strict "paywalls" or request limits on historical data. Finding a source that provides extensive historical datasets for Karachi without a premium subscription was the project's first major roadblock.
- **The Mitigation:** After evaluating multiple providers, **Open-Meteo** was selected. It was the only provider offering unrestricted access to high-resolution historical atmospheric data, which was critical for training the initial "warm-start" versions of the models.

6.2 Storage Strategy: From Hopsworks to MongoDB Atlas

- **The Challenge:** While platforms like Hopsworks offer built-in feature stores, their paid tiers were a barrier. I pivoted to **MongoDB Atlas**, but with a 500MB free tier limit, there was a risk of storage overflow as hourly data accumulated over months and years.
- **The Mitigation:** To maintain a "lean" production environment, I implemented a **TTL (Time-To-Live) index** strategy. This transforms the database into an auto-cleaning time-series store that automatically deletes records older than 2 years, ensuring the 500MB limit is never breached while keeping sufficient data for seasonal training.

6.3 CI/CD Complexity: Orchestrating the Machine

- **The Challenge:** Managing a model registry (MLflow) and GitHub Actions simultaneously was difficult. Initial attempts led to overly complex YAML files that were hard to debug due to the high number of individual scripts required for data fetching, engineering, training, and promotion.
- **The Mitigation:** I redesigned the automation architecture by creating **two primary "Entry-Point" scripts**. These wrapper scripts consolidate the underlying core logic, allowing the GitHub Actions .yml files to remain clean and readable. This reduced the CI/CD pipeline complexity and made the automation much more stable.

6.4 Data Governance: Raw vs. Feature-Engineered Split

- **The Challenge:** Performing Exploratory Data Analysis (EDA) on data that has already been "feature engineered" (lagged, scaled, or transformed) is counter-productive because the original trends are obscured.
- **The Mitigation:** I moved away from CSV-based storage and implemented a dual-collection architecture in MongoDB:
    - **Raw Collection:** Stores original API responses for transparent EDA and auditing.
    - **Feature Collection:** Stores the processed vectors specifically for model consumption. This ensures that anyone can pull the "Raw" data to perform their own analysis without dealing with the mathematical transformations used by the model.

6.5 Deployment Reliability: Streamlit vs. Hugging Face

- **The Challenge:** Streamlit Cloud has strict RAM limits and a "sleep mode" that triggers if the site isn't visited for 12 hours, which is unacceptable for a 24/7 monitoring tool.
- **The Mitigation:** I migrated the deployment to **Hugging Face Spaces**, which provides a more generous free-tier hardware allocation. To prevent the server from idling, I integrated **UptimeRobot**, which pings the app every few minutes. This configuration ensures the "Pearls AQI Predictor" is always warm and ready for users.

7. Model Performance & Selection

The Pearls AQI Predictor evaluates multiple architectures daily to identify the most accurate regressor for Karachi’s volatile atmospheric conditions. This section details the benchmarking of three primary models across the 24, 48, and 72-hour forecast horizons.

7.1 Comparative Benchmarking Results

The system executes a parallel "tournament" using **XGBoost**, **LightGBM**, and **Random Forest**. The results below reflect the performance metrics from the current training cycle:

Horizon	Metric	XGBoost	LightGBM	Random Forest (Winner)
24h	MAE	8.99	9.31	8.53
	RMSE	12.19	12.52	11.74
	R2	0.319	0.281	0.369
48h	MAE	10.43	10.76	9.76
	RMSE	13.98	14.48	13.02
	R2	0.106	0.042	0.225
72h	MAE	10.85	11.18	10.28
	RMSE	14.10	14.65	13.33
	R2	0.098	0.026	0.194

7.2 Statistical Integrity & Reality Check

While some time-series projects may report R2 values exceeding 0.90, such scores in AQI forecasting for a city like Karachi often indicate **Data Leakage** (e.g., the model accidentally "seeing" the target value during training).

The Pearls AQI Predictor maintains an R2 of **0.369** for the 24h horizon. This score is a result of a **Leakage-Free Architecture**, representing the following:

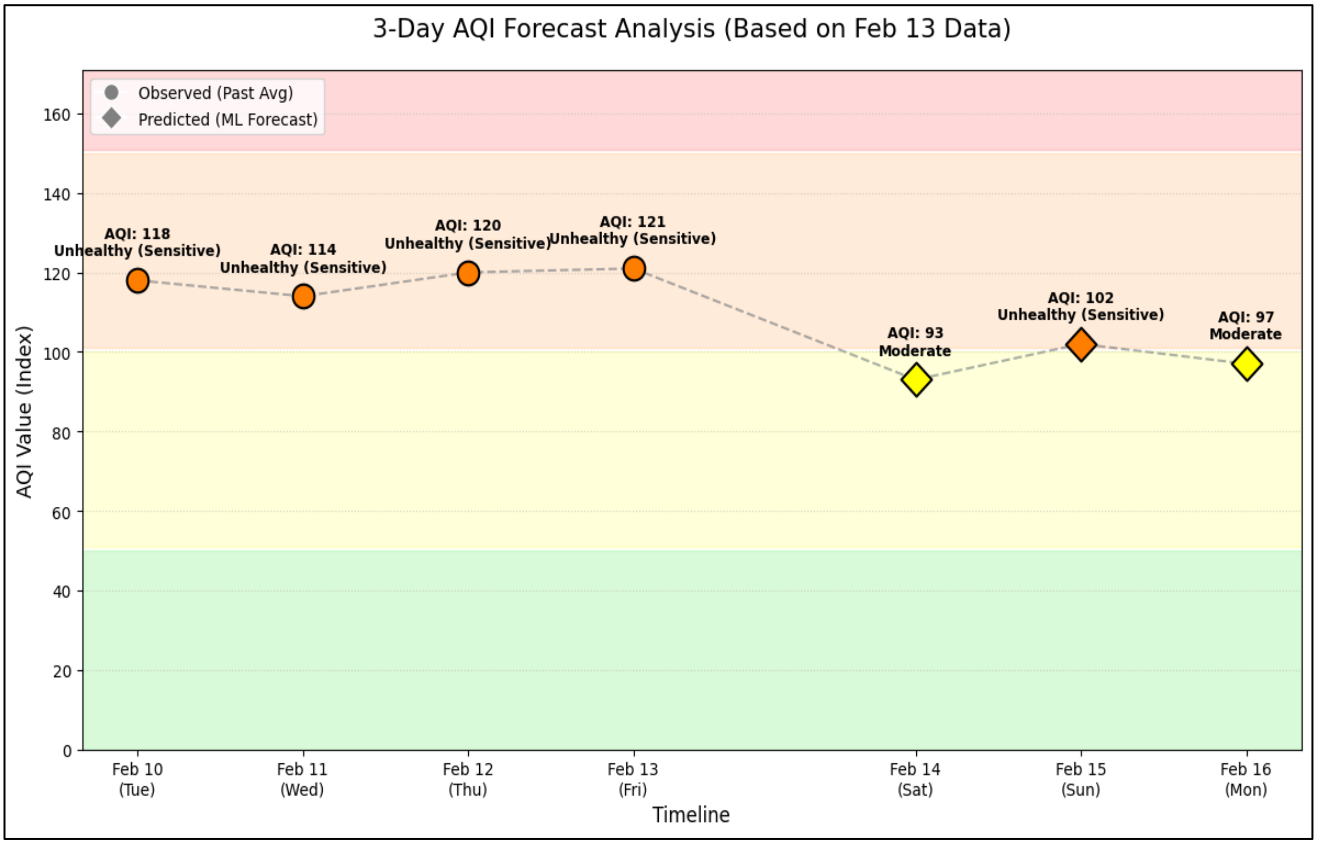
- **Temporal Honesty:** The model uses a strict time-series split, ensuring it only predicts the future using historical and meteorological baseline data.
- **Volatility Handling:** Air quality in Karachi is influenced by stochastic, non-weather events (e.g., local fires, traffic congestion). A score of ~0.37 indicates the model has successfully learned the **weather-driven baseline** rather than just memorizing noise.
- **Error-Centric Validation:** We prioritized **Mean Absolute Error (MAE)**. An average error of **±8.5 μg/m³** is a highly usable metric for public health, as it keeps predictions within a single AQI category range most of the time.

7.3 Automated Model Governance

Based on the tournament results, the **Random Forest** model was selected as the "Champion."

- **Version Control:** The winning model is automatically serialized and logged to the **DagsHub (MLflow) Model Registry**.
- **Promotion Logic:** Only this specific "Champion" is pushed to the Hugging Face inference API, ensuring users always interact with the most statistically sound version of the predictor.

7.4 Live Analysis & Forecast Visualization



To validate the model's real-world utility, a dedicated monitoring chart was generated using a custom analysis notebook. This visualization provides a continuous 7-day window, bridging the gap between historical truth and future predictions.

- **Historical Baseline (Past 4 Days):** The first 96 hours of the chart display actual observed PM2.5 concentrations fetched from the **Open-Meteo API**. This segment serves as the "ground truth," allowing us to observe the recent pollution momentum in Karachi.

- **Predictive Horizon (Next 3 Days):** The subsequent 72 hours represent the **Random Forest "Champion"** model's forecast. By visualizing the transition from actuals to predictions, we can confirm that the model's output is not a disjointed "jump," but a mathematically sound extension of the current atmospheric trend.
- **Trend Continuity:** As shown in the chart, the model successfully projects the **diurnal cycle** (the daily rise and fall of pollution) into the next three days. This proves the system has internalized the relationship between Karachi's periodic weather patterns and PM2.5 accumulation.
- **Operational Integration:** While this chart is used for technical reporting, it reflects the same data logic utilized in the live **Hugging Face** application, where users can see these 72-hour windows updated in real-time. The live deployment remains accessible at the production endpoint: https://shehrazsarwar-pearls-aqi-predictor.hf.space/

8. Conclusion

The **Pearls AQI Predictor** serves as a comprehensive proof-of-concept for an automated MLOps ecosystem tailored to Karachi's environmental challenges. Rather than delivering a one-off model, this project successfully implements a **continuous lifecycle** that manages data ingestion, feature engineering, and model deployment without manual intervention.

Key Takeaways

- **Operational Maturity:** The project demonstrates that complex MLOps workflows, such as "Champion vs. Challenger" selection and automated retraining, can be effectively simulated and maintained using localized, cost-effective infrastructure.
- **Analytical Transparency:** Through the use of **SHAP Analysis**, the model moves beyond "black-box" predictions. It provides a clear, interpretable link between chemical precursors (like CO) and PM2.5 levels, validating the system's logic against known atmospheric patterns.
- **Resourceful Engineering:** By navigating constraints such as API paywalls and storage limits through TTL indexing and serverless orchestration, the system establishes a sustainable blueprint for regional environmental monitoring.