App Model Training: MLOps Pipeline

Sujay S

May 1, 2025

1 Architecture Diagram

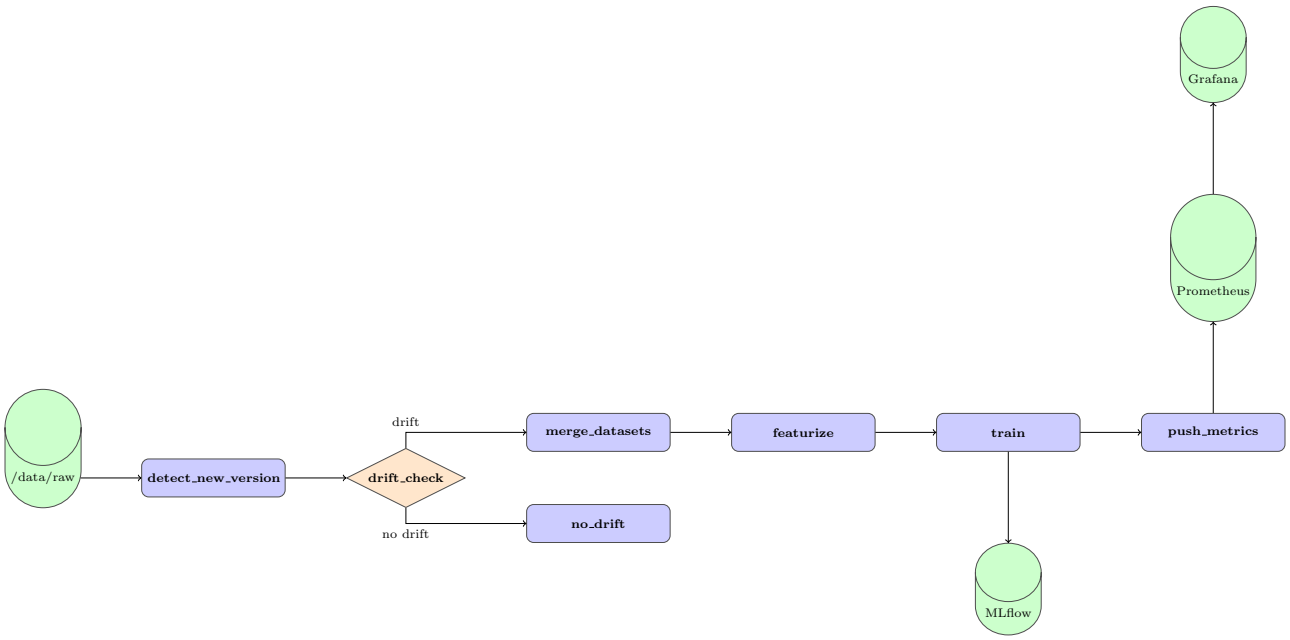

Figure 1: MLOps Pipeline Architecture

Explanation of Blocks

- **detect_new_version**: Checks the `/data/raw` directory and identifies the latest data version (e.g., v2 if v1, v2 exist).
- **drift_check**: Performs data drift detection between baseline and new datasets using Evidently.
- **merge_datasets**: Merges the baseline and new datasets upon drift detection.
- **featurize**: Extracts features from the merged dataset.
- **train**: Retrains the ML model and logs metrics to MLflow.
- **push_metrics**: Pushes pipeline and model metrics to Prometheus for visualization.
- **Grafana Dashboard**: Visualizes metrics like drift status, accuracy, and throughput.
- **MLflow**: Tracks model training, artifacts, and metrics.

2 High-Level Design Document

The pipeline leverages Apache Airflow to orchestrate data version management, drift detection, conditional dataset merging, feature extraction, model retraining, and metrics reporting. Prometheus and Grafana provide monitoring and visualization capabilities to track the pipeline performance and model metrics, ensuring transparency and ease of debugging. MLflow is utilized for effective model lifecycle management, logging, and artifact storage.

Design choices made:

- **Airflow**: Chosen for its powerful workflow orchestration, task dependencies management, and ease of extending pipelines.

- **Evidently**: Selected for accurate and comprehensive data drift detection.
- **MLflow**: Preferred for its seamless integration with training workflows, model registry, and metrics logging.
- **Prometheus and Grafana**: Picked for robust real-time monitoring, alerting, and easy visualization.

3 Low-Level Design Document

Endpoint Definitions and I/O Specifications

detect_new_version

Input: `/data/raw` directory path
Output: Latest data version identifier (string, e.g., "v2")

drift_check

Input: Baseline and new datasets (CSV/parquet)
Output: Decision ("merge_datasets" or "no_drift"), Drift Metrics (boolean, drift share)

merge_datasets

Input: Baseline and new datasets
Output: Merged dataset file (CSV/parquet)

featurize

Input: Merged dataset
Output: Feature-enriched dataset (CSV/parquet)

train

Input: Featurized dataset, Model hyperparameters (JSON)
Output: Trained model (MLflow artifact), Performance metrics (accuracy, AUC, F1-score)

push_metrics

Input: Pipeline metrics, model performance metrics
Output: Metrics pushed to Prometheus

Monitoring Endpoints

- **Grafana Dashboard**: `http://localhost:3001/`
- **MLflow**: `http://localhost:5001/`
- **Airflow**: `http://localhost:8080/`

4 Pipeline Monitoring

Pipeline metrics and model metrics can be monitored on Grafana dashboard as shown below.

Figure 2: Pipeline Monitoring Dashboard