Automate Data Pipelines for Real-Time Analytics using Prefect.io

Project Overview

This project is part of the **DSI321: BIG DATA INFRASTRUCTURE** course.It focuses on designing and deploying an automated data pipeline for real-time analytics using Prefect.io. The data collected relates to PM2.5 air quality levels in Thailand. The system is containerized with Docker, orchestrated by Prefect, visualized with Streamlit, and further enhanced by integrating a large language model—Typhoon LLM—to provide intelligent summarization and user-friendly insights.

Introduction

In recent years, air pollution—especially fine particulate matter (PM2.5)—has become a critical environmental issue in Thailand . Accurate and timely monitoring of air quality is essential for public awareness, health decision-making, and long-term policy planning. To support this need, this project focuses on building an automated data pipeline that continuously collects, processes, and analyzes air quality data in near real-time.

The pipeline utilizes the **Air Quality API**, which provides hourly average PM2.5 concentration data from automatic air quality monitoring stations operated by the **Air4Thai** network across the country. This API is officially maintained by the **Pollution Control Department of Thailand**, offering reliable access to high-quality environmental data.

To ensure automation, scalability, and reproducibility, the project leverages **Prefect.io** for workflow orchestration and **Docker** for containerization. Insights from the processed data are visualized using a web-based dashboard built with Streamlit.

To make the insights even more accessible and easier to interpret, the project integrates **Typhoon LLM** to generate natural language summaries of air quality trends. The language model helps interpret PM2.5 data by identifying significant changes, highlighting anomalies, and providing region-specific recommendations in both Thai and English. Users can interactively explore real-time data and receive AI-powered explanations via the dashboard—bridging the gap between complex analytics and everyday understanding.

Getting Started

To run it locally:

1. **Clone the Repository**:

```
$ git clone <this-repo-url>

$ cd <this-repo-folder>
```

2. **Start Docker Services**: Launch Prefect server and Streamlit:

```
docker-compose up -d --build
```

After successful deployment, you can access:
Prefect Dashboard : http://localhost:4200
JupyterLab : http://localhost:8888
LakeFS : http://localhost:8001 (changed from default 8000)
Stramlit : http://localhost:8501

> [!IMPORTANT]
> Before executing `deploy.py`, you must first **create a repository named `air-quality-data` in LakeFS**.

> [!TIP]
> You can do this via the LakeFS web interface or using the CLI command:
>
> ```
> lakectl repo create lakefs://air-quality-data
> ```

3. **Deploy Prefect Flow**: Deploy the pipeline with a start of the hour (minute 40).:

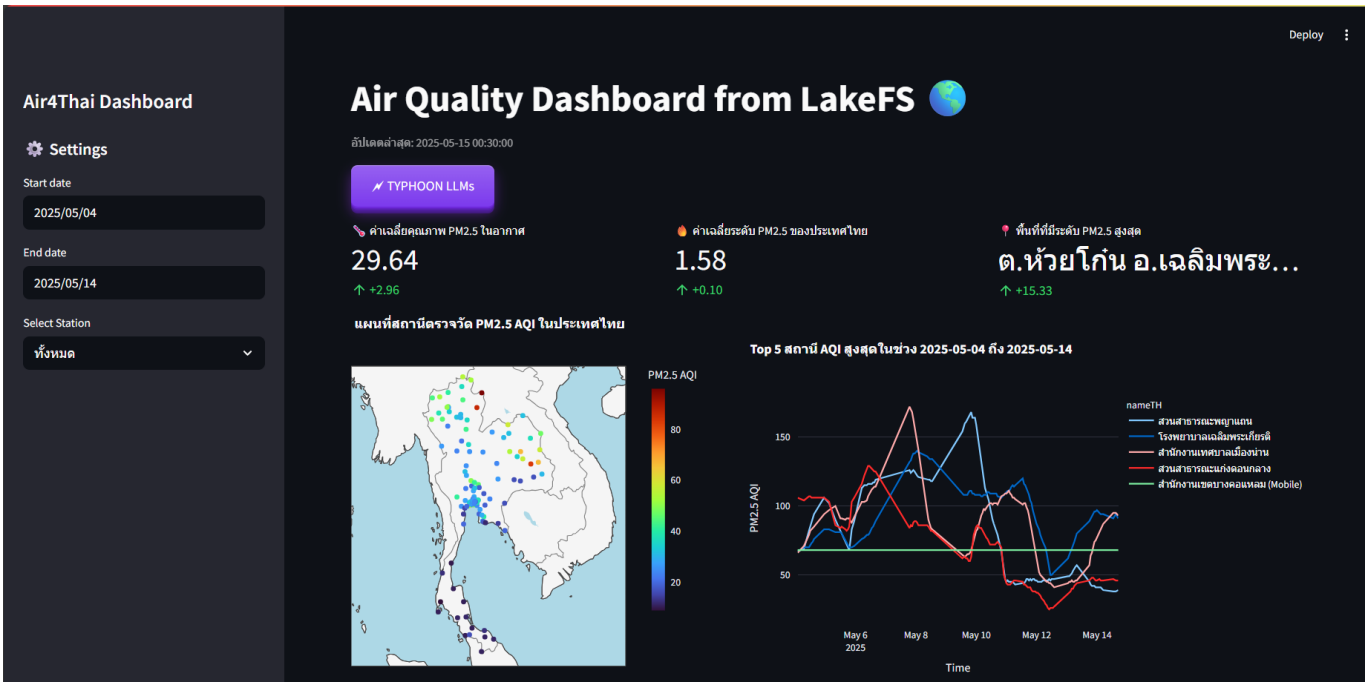
```
python src/pipeline.py deploy

# OR via JupyterLab at http://localhost:8888
## Start new terminal session

python deploy.py
```

This creates a deployment named `data-pipeline` in the `default-agent-pool` work pool, scheduled to run at minute 40 of every hour. (`cron="40 * * * *"`).

📊 **Streamlit Interface** : Real-Time Air Quality Dashboard



Key Dashboard Features:

- **Real-time PM2.5 visualization** for various provinces
- **Time-series trend graphs** for selected stations
- **Interactive map** with station markers and AQI color coding
- **Typhoon LLM-powered summary insights** per region/time

🤖 **Typhoon LLM Integration:** Summarization & Insight Generation

To enhance the real-time air quality dashboard, we integrated Typhoon LLM, a Thai language large language model, to automatically generate concise summaries and actionable insights from air quality data.

Why Typhoon LLM?

- **Thai Language Support**
  Typhoon LLM is trained on extensive Thai text data, making it ideal for interpreting and generating insights in Thai for local users.
- **Powerful for Summarization**
  It excels at summarizing trends and highlighting anomalies, such as identifying provinces with unusually high PM2.5 levels or suggesting areas that may require attention.
- **Real-Time Integration**
  Combined with Prefect and Streamlit, Typhoon LLM enables real-time insight generation as new data flows in.

Techniques Used

- **Prompt Engineering**
  We designed clear and structured prompts, e.g.: “Summarize the air quality situation in Thailand for

{time range} and highlight provinces with high PM2.5 levels.”

- **Workflow Integration via Prefect**
  A task was added in the Prefect pipeline to send cleaned and updated data to Typhoon LLM for automatic summarization.

- **Insight Cards in Streamlit**
  The model's output is displayed as user-friendly “insight cards” on the dashboard, helping users quickly understand key information without analyzing charts in detail.

Data Schema

The data schema is defined in `src/SCHEMA.md`. For this air quality data example:

```
{
  "columns": [
    "timestamp", "stationID", "nameTH", "nameEN", "areaTH",
    "areaEN", "stationType", "lat", "long", "PM25.color_id",
    "PM25.aqi", "year", "month", "day", "hour"
  ],
  "types": [
    "datetime64[ns]", "string", "string", "string", "string",
    "string", "string", "float64", "float64", "int64",
    "float64", "int64", "int64", "int32", "int32"
    ],
  "key_columns": [
    "timestamp", "stationID", "nameTH", "nameEN", "areaTH",
    "areaEN", "stationType", "lat", "long", "PM25.color_id",
    "year", "month", "day", "hour"
  ]
}
```

- **timestamp**: ISO format timestamp of data collection.
- **stationID**: Station ID code.
- **nameTH**: Station name in Thai.
- **nameEN**: Station name in English.
- **areaTH**: Area name in Thai.
- **areaEN**: Area name in English.
- **stationType**: Type of the station (e.g., roadside, general area).
- **lat**: Latitude of the station.
- **long**: Longitude of the station.
- **PM25.color_id**: Color ID for visualization based on PM2.5 level.
- **PM25.aqi**: PM2.5 Air Quality Index (AQI).
- **year**: Year of data record.
- **month**: Month of data record.
- **day**: Day of data record.
- **hour**: Hour of data record.

Key columns are used for data quality checks (no missing values allowed).

🛠️ Technologies Used

- **Prefect.io** - For workflow orchestration and scheduling
- **Docker** - For containerizing the application and ensuring environment consistency
- **LakeFS** - For data versioning and management
- **Streamlit** - For interactive data visualization
- **Typhoon LLM** - For natural language summarization of air quality data
- **Air Quality API** - For real-time PM2.5 data from the Air4Thai network

⚖️ Support & Contact

If you have any questions, suggestions, or need assistance, please open an issue or contact the developer directly.🤝

DSI321: BIG DATA INFRASTRUCTURE | KorNxHaidar | 2025