AI601: DATA ENGINEERING FOR AI SYSTEMS

Rubab Zahra Sarfraz

Projects

2025-04-05

PROJECT OVERVIEW

Objective:

- Implement a full data engineering lifecycle: data collection → storage → processing → basic model/analytics → deployment.
- If your dataset/problem is small or straightforward, go deeper in one specialized track (e.g., advanced data quality, real-time streaming, etc.).
- Teams: 3–4 members

CORE REQUIREMENTS

- **Data Collection & Ingestion**
    - Use one or more data sources (could be an API, open dataset, or web scraping).
    - Store data in a database/data lake with proper modeling.
- **Basic Data Transformation & Processing**
    - Clean the data (handle missing values, inconsistent formats).
    - Possibly do light feature engineering (if building an ML model).
- **Simple ML Model / Analytics**
    - E.g., a basic regression or classification OR a clear analytical insight.
    - Keep it simple but ensure it demonstrates an engineering pipeline.

CORE REQUIREMENTS

- **Deployment & Frontend**
    - Containerize with Docker or provide a minimal endpoint (FastAPI, Flask) OR an automated batch job (reference lab 7)
    - Show that your pipeline can run in a “production-like” environment.
    - Your app should have an interface (streamlit preferred)

- **Basic Logging & Monitoring**
    - Capture logs of your pipeline runs or requests (model inference logs, pipeline run logs, errors).
    - Show how you’d monitor or debug the solution if something goes wrong.

DEEP DIVE TRACKS

If your dataset/problem is **small/easy**, pick **one** area to explore more deeply:

1. **Advanced Data Quality & Governance**
2. **Real-time Streaming & Distributed Processing**
3. **Orchestration & Scheduling** (Airflow, Prefect)
4. **Production-grade Model Deployment** (versioning, drift detection, advanced monitoring)
5. **Scalable Batch Processing & Data Lakes** (Spark, partitioning, big data best practices)
6. **Agentic Workflows**

Note: If your data is already large/complex, the full end-to-end pipeline itself will likely be enough of a challenge.

DELIVERABLES

- In-class Presentation
- Github Repo with a proper README
- Live Demo/Video
- Top projects will get shoutouts from course staff and industry referrals!

SAMPLE PROJECTS 1

- **Medium Author Success Predictor:** Extract key metrics from Medium's top authors to forecast the popularity potential of new articles or authors.
- **Spotify Song Recommender:** Aggregate Spotify listening data to deliver personalized music recommendations based on user preferences.
- **Movie and TV Series Metadata Platform:** Consolidate and enrich movie and TV metadata for efficient browsing, search, and discovery.
- **Weather Data Aggregation and Historical Analysis:** Automate collection and processing of historical weather data for trend analysis and climate insights.

SAMPLE PROJECTS 2

- **Intelligent Voice Assistant for Customer Support:** Transcribe and analyze customer calls in real-time to identify intents and streamline support operations.
- **News Aggregation and Content Tagging Platform:** Collect, categorize, and tag news content automatically for enhanced readability and discovery.
- **Intelligent Data Lineage Tracker:** Develop a monitoring system that leverages LLMs to predict potential failures or bottlenecks in data pipelines based on historical performance data.

SAMPLE PROJECTS 3

- **Predictive Data Pipeline Monitoring System:** Monitor pipeline metrics proactively to anticipate failures and maintain pipeline health.
- **Data Compliance Audit Assistant:** Create an assistant that utilizes LLMs to automatically review datasets and data processing activities for compliance with regulations such as GDPR or HIPAA.

TIPS

- Shortlist a domain you are interested in.
- Pick any problem that needs some level of intelligence.
- Gather and list down data sources to collect data from.
- Map out each stage.
- Start implementation!
- Research based ideas are welcomed too!