2024 S2C COMP5339

Assignment 2 Report

Group ID: COMP5339HD

SID: 500174041
SID: 500000463
SID: 530744122
SID: 490348738
SID: 510052735

1. Overview

1.1 Objective

The objective of this project is to build a comprehensive data pipeline for processing and analyzing electrocardiogram (ECG) data linked with clinical ground truth from hospital records. This data pipeline is necessary because the ECG dataset is large and complex, requiring an efficient pipeline to handle data ingestion, cleaning, and transformation. In addition, a well-designed pipeline ensures that the process is reproducible and can scale with additional data or new requirements.

The target users of this data pipeline could be healthcare professionals and data scientists who are interested in insights derived from ECG data linked with clinical outcomes. Our data pipeline could contribute to providing clean, structured data ready for analysis, streamlining data processing to reduce manual effort, and enabling advanced analytics.

1.2 Data Description

The dataset used in this project is the MIMIC-IV-ECG-ICD dataset, which links ECG recordings from MIMIC-IV-ECG with clinical ground truth from the MIMIC-IV database. This derived dataset could be used to foster further research on ECG-based prediction models with clinical ground truth and build a resource for benchmarking clinical ECG prediction models.

The detailed description of each column in the dataset can be found at https://canvas.sydney.edu.au/courses/59629/pages/assignment-2-eicu-data-description-optional

2. Solution

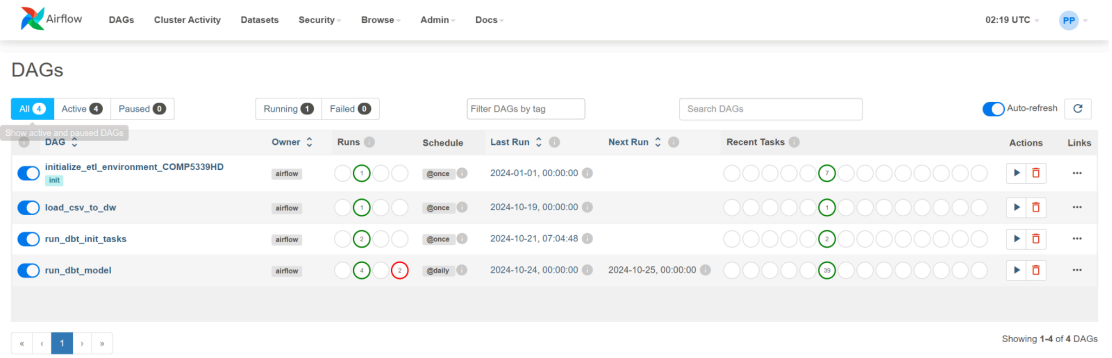
2.1 Airflow DAGs

- initialize_etl_environment_COMP5339HD: This dag initializes the ETL environment by creating multiple tables including patients, ecg_recordings, hospitals, diagnoses, ecg_metadata, and fold_assignment for MIMIC-IV-ECG data.
- load_csv_to_dw: This dag loads all the tables and corresponding CSV data to our data warehouse.
- run_dbt_init_tasks: This dag initializes the process of our dbt models.
- run_dbt_model: This dag runs all the SQL tasks in our dbt models including all the files in the folders src, staging, dim, and fact.

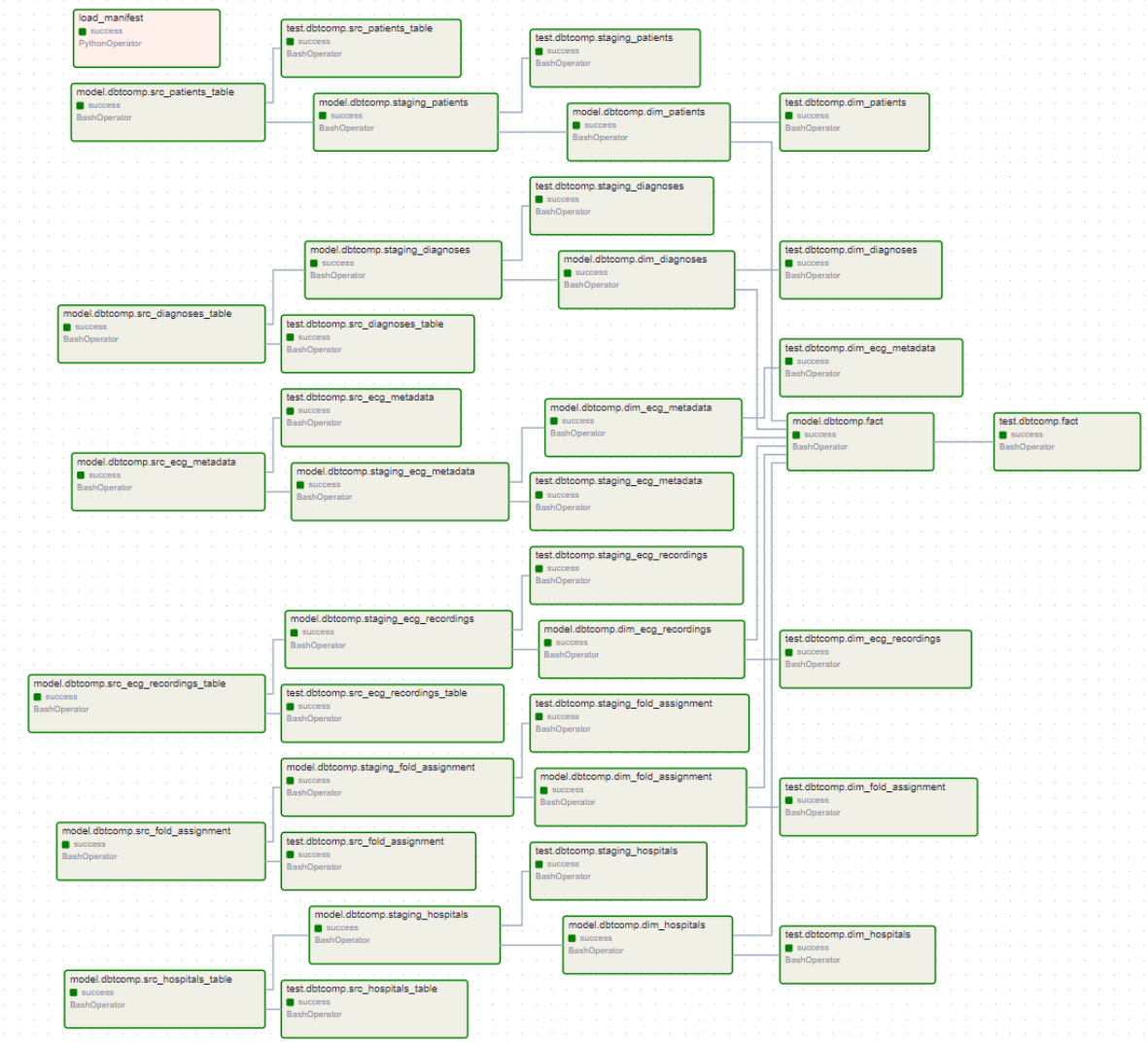
2.2 DBT

- src/: Create source models for each table. These models read directly from the source tables and are used as the base for staging models.
- staging/: Create staging models that build upon the source models, performing data cleaning.
- dim/: Create dimension models that represent the entities in the data model.
- fact/: Create the fact table that aggregates data for analysis.

2.3 Architecture



2.4 ETL Pipeline

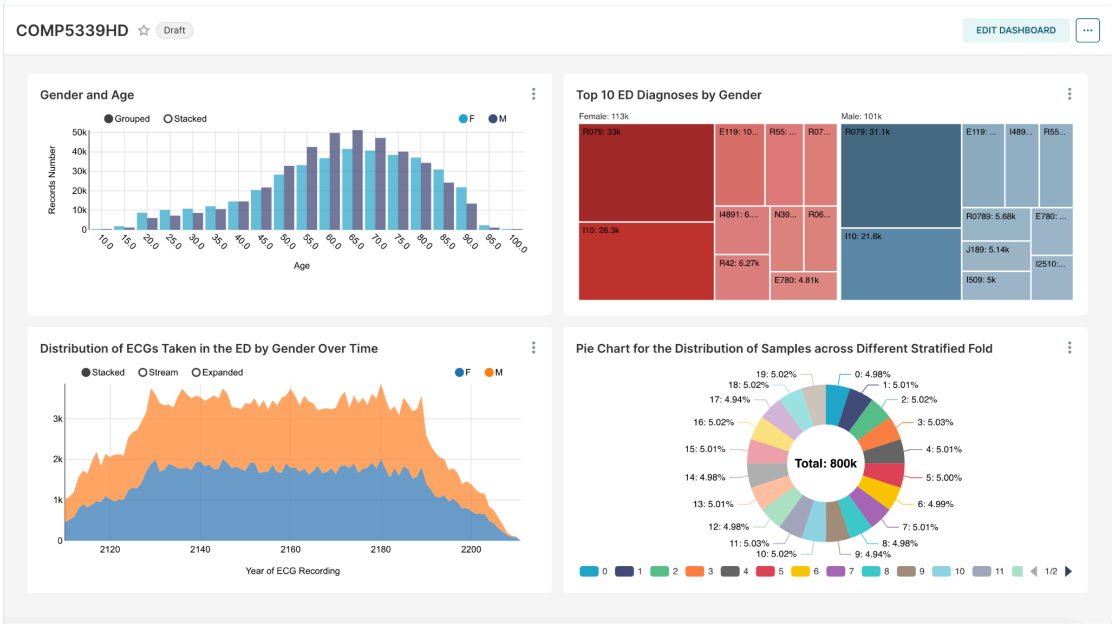


3. Dashboard

3.1 Superset Code Modifications

After building our ETL pipeline in Airflow, we create visualizations using Superset. We begin by writing SQL queries in the editor for each plot. Then, we save the queries and customize the chart based on our specific topics. Finally, we insert the charts into the dashboard.

3.2 Superset Visualization & Data Insights



The dashboard provides a thorough overview of the MIMIC data, including visualizations related to patient demographics, emergency department (ED) diagnoses, ECG recordings, and data stratification. It is intended for healthcare analysts and researchers.

- Bar chart: The patients aged between 60 and 70 account for the largest proportion, suggesting that people in this age group are more prone to illness.
- Treemap: R079 (chest pain) and I110 (hypertensive heart disease with heart failure) are common ED diagnoses for both males and females.
- Area chart: The period with the most frequent ECG recordings ranges from 2130 to 2190, where males take more ECGs than females.
- Pie chart: The stratified folds tend to be evenly distributed among the dataset.

4. Setup Process

4.1 Detailed Directory Structure (Only Main Files)

dbt/

- **dbt_comp5339/:** Main dbt project directory.
  - **models/:** Contains SQL models for staging, dimension, and fact tables.
    - **staging/:** SQL scripts for staging tables.

- **dim/:** SQL scripts for dimension tables.
- **fact/:** SQL scripts for fact tables.
- **sources.yml:** Source configuration file.
- **src/:** Source SQL scripts.

airflow/

- **dags/:** Contains DAGs and related scripts for Airflow.
  - **initialize_etl_environment_COMP5339HD.py:** Create the necessary database objects.
  - **load_csv_to_dw.py:** Import data from data.csv to the Data Warehouse.
  - **run_dbt_init_tasks.py:** Initialize dbt models.
  - **run_dbt_model.py:** Run all the SQL tasks.

superset/

- **assets/:** Assets for Superset, such as dashboards.
  - **dashboard.json:** JSON file defining a Superset dashboard.

4.2 System Setup

(1) Setup and Data Generation

- **Setup the Environment:**
  - Install Docker on your system.
  - Download the project repository and navigate to the root folder.
- **Build and Start the Docker Containers:**
  - Run the following command to build and start the Docker containers: *docker compose up --build*

(2) ETL Orchestration with Airflow

- **Initialize ETL Environment:**
  - Access the Airflow GUI at http://localhost:8080 .
  - Use the default credentials (user: airflow, password: airflow) to log in.
  - Trigger the initialize_etl_environment_COMP5339HD DAG.
- **Data Import:**
  - Create a dataset folder in the root directory. Put data.csv in that folder.
  - Trigger the load_csv_to_dw DAG.

(3) Data Transformation with dbt

- **Initialize dbt:**
  - Access the Airflow GUI and trigger the run_dbt_init_tasks DAG.
- **Run dbt Models:**
  - Trigger the run_dbt_model DAG.

(4) Data Analysis and Visualization with Superset

- **Explore Data with Superset:**
  - Access the Superset GUI at http://localhost:8088 .
  - Use the default credentials (user: admin, password: admin) to log in.
  - Create visualizations to analyze the data in the Data Warehouse.

5. Limitations

While the implemented data pipeline successfully integrates ECG data with clinical ground truth, there are several limitations in the current approach that need to be addressed:

- **Limited Error Handling:** The current pipeline lacks comprehensive data validation checks during the ingestion and transformation stages. This can lead to undetected errors propagating through the pipeline.
- **Scalability Concerns:** As the volume of data grows, the pipeline may face performance issues, particularly during heavy transformation tasks in dbt or data loading processes in Airflow.

6. Improvements

To enhance the current data pipeline and address its limitations, several potential improvements are recommended for future development:

- **Implement Comprehensive Data Cleaning Procedures:** Develop robust data cleaning scripts within dbt models to handle missing values, standardize formats, and correct inconsistencies across all relevant fields.
- **Improve Scalability:** Refine dbt models by optimizing SQL queries and using incremental models where appropriate. Leveraging distributed computing resources could also contribute to handling larger workloads.