Assignment - 2 (CS5830)

Setup a Data Engineering Pipeline

-Salem Aslam (BE20B027)

Introduction

This report presents two Directed Acyclic Graphs (DAGs) designed to automate the extraction and processing of weather data of NCEI. The DAGs are built using Apache Airflow.

1. Weather Data Extraction DAG

1.1. Overview

- DAG Name: weather_data_extraction_1
- Description: Fetches and processes weather data from the National Centers for Environmental Information (NCEI) website.
- Start Date: April 14, 2024

1.2. Workflow Steps

1. Fetch HTML Page (`fetch_page`):
   - Downloads the HTML page containing CSV links from the NCEI website for the specified year (2023).
   - Executes a curl command using the `BashOperator`.

2. Select Random Files (`select_files`):
   - Parses the HTML page, extracts CSV file links, and randomly selects a subset of files.
   - Utilizes BeautifulSoup for HTML parsing and the random module for file selection.

3. Fetch CSV Files (`fetch_files`):
   - Downloads selected CSV files using curl based on the links extracted in the previous step.
   - Accesses the list of selected files from the XCom system.

4. Zip Files (`zip_files`):
   - Zips the downloaded CSV files into a single zip archive.
   - Uses Python's `zipfile` module to create the zip file.
   - Removes original CSV files after zipping.

5. Move Zip File (`move_zip_file`):
   - Moves the zip archive to a specified directory (`/tmp/new_data_dir`).
   - Creates the target directory if it doesn't exist.

- Utilises the `BashOperator` for file manipulation.

2. Weather Data Processing DAG

2.1. Overview

- DAG Name: weather_data_pipeline
- Description: Processes weather data extracted by the first DAG.
- Start Date: April 14, 2024

2.2. Workflow Steps

1. Wait for Archive (`wait_for_archive`):
   - Waits for the archive file (`2020_data.zip`) to appear in the specified directory (`/tmp/new_data_dir`).
   - Utilises the `FileSensor` to monitor the file's existence.

2. Unzip Archive (`unzip_archive`):
   - Creates a directory for extracted files and unzips the archive into it.
   - Uses the `BashOperator` to execute shell commands.

3. Extract and Filter Data (`extract_and_filter_data`):
   - Extracts and filters data from CSV files using Apache Beam.
   - Filters columns with names starting with 'Hourly' and keeps essential columns like 'DATE', 'LATITUDE', and 'LONGITUDE'.

4. Compute Monthly Averages (`compute_averages`):
   - Computes monthly averages of weather data using Apache Beam.
   - Groups data by date and calculates the mean of numeric columns.

5. Combine Data (`Comb_data_loc`):
   - Combines data from multiple CSV files into one DataFrame.
   - Identifies common columns and selects a specific month for merging.

6. Generate Geomaps (`Geo_map`):
   - Generates geomaps based on the combined data using Apache Beam.
   - Plots data on a world map using GeoPandas and Matplotlib.

Conclusion

The Weather Data Extraction and Processing DAGs automate the retrieval, filtering, and analysis of weather data, enabling efficient data-driven decision-making in various domains. These DAGs offer scalability, reliability, and reproducibility, making them valuable tools for weather data management and analysis.