TLC Trip Record Data Processing

Introduction

This script processes taxi trip record data from the TLC dataset. It utilizes PySpark for data manipulation and analysis.

Technologies Used

- PySpark
- Python
- Spark SQL

Libraries and Frameworks

- `pyspark.sql`: Provides SparkSession for interacting with Spark SQL and DataFrame operations.
- `pyspark.sql.functions`: Contains functions for DataFrame operations like aggregation, filtering, etc.
- `pyspark.sql.types`: Includes data types used in PySpark.

Initialization

- `SparkSession`: Creates a Spark session for Spark application.

Data Transformation

- `data_transformation`: Custom function to transform taxi trip data, applying necessary modifications and schema alterations.
- `merge_dataframes`: Function to merge additional location details with the trip data.

Constants

- `yellow_tripdata_files`: File paths for yellow taxi trip data.
- `green_tripdata_files`: File paths for green taxi trip data.

- `columns_to_drop`: Columns to be dropped from the DataFrame.

Processing Steps

1. **Yellow Taxi Data Processing**:

- Reads yellow taxi trip data files specified in 'yellow_tripdata_files'.
- Performs custom transformations on the data using **data_transformation()**.
- Removes duplicate rows.
- Displays a preview of the transformed data.

2. **Green Taxi Data Processing**:

- Reads green taxi trip data files specified in 'green_tripdata_files'.
- Performs custom transformations on the data using **data_transformation()**.
- Removes duplicate rows.
- Displays a preview of the transformed data.

3. **Loading Location Details**:

- Reads 'taxi_zone_lookup.csv' file into the Spark DataFrame named 'taxi_zone_lookup_df'.
- Removes duplicate rows from the location details DataFrame.

4. **Merging Location Details with Taxi Trip Data**:

- Merges location details into the yellow taxi trip data DataFrame ('yellow_tripdata_df') using merge_dataframes().
- Merges location details into the green taxi trip data DataFrame ('green_tripdata_df') using merge_dataframes().

5. **Displaying DataFrame Schemas:**

- Displays the schema of the yellow taxi trip data DataFrame ('yellow_tripdata_df').

- Displays the schema of the green taxi trip data DataFrame ('green_tripdata_df').

Conclusion:

This script efficiently processes yellow and green taxi trip data, incorporates location details, and prepares the DataFrames for subsequent analytical operations.

PART TWO

Taxi Trip Data Analysis using PySpark and Matplotlib

This documentation outlines the analysis and visualization of yellow and green taxi trip data, including counts, fare analysis, trip duration, pickups, and monthly trip counts. The analysis is conducted using PySpark for data processing and Matplotlib for visualization.

Data Analysis and Visualization:

Task 2A & 2B: Trips per Pickup Borough

- Calculates and displays counts of trips per Pickup Borough for Yellow and Green Taxis.
- Plots line graphs displaying counts of trips per Pickup Borough for each taxi type.
- The order of counts of trips per Pickup Borough for Yellow Taxis in descending order are Manhattan, Queens, Unknown, Brooklyn, Bronx, EWR and Staten Island.
- The order of counts of trips per Pickup Borough for Green Taxis in descending order are Manhattan, Queens, Brooklyn, Bronx, Unknown, Staten Island and EWR.

Task 2C & 2D: Trips per Dropoff Borough

- Calculates and displays counts of trips per Dropoff Borough for Yellow and Green Taxis.
- Plots line graphs displaying counts of trips per Dropoff Borough for each taxi type.
- The order of counts of trips per Dropoff Borough for Yellow Taxis in descending order are Manhattan, Queens, Brooklyn, Unknown, Bronx, EWR, Staten Island.
- The order of counts of trips per Dropoff Borough for Yellow Taxis in descending order are Manhattan, Queens, Brooklyn, Bronx, Unknown, Staten Island and EWR.

Task 4: Top 5 Pickup and Drop-off Boroughs

- Identifies and visualizes the top 5 Pickup and Drop-off Boroughs for Yellow and Green Taxis separately using bar plots.
- The top 5 Pickup Boroughs for Yellow Taxis are Manhattan, Queens, Unknown, Brooklyn, Bronx.
- The top 5 Pickup Boroughs for Green Taxis are Manhattan, Queens, Brooklyn, Bronx, Unknown.
- The top 5 Dropoff Boroughs for Yellow Taxis Manhattan, Queens, Brooklyn, Unknown, Bronx.
- The top 5 Dropoff Boroughs for Green Taxis are Manhattan, Queens, Brooklyn, Bronx, Unknown.

Task 6: Fare per Mile Analysis

- Filters data for March 2023, calculates fare per mile, identifies outliers, and visualizes fare per mile for Yellow Taxi trips.

Task 7: Percentage of Solo Trips

- Calculates and compares the percentage of solo trips for Yellow and Green Taxi datasets.
- Percentage of solo trips for Yellow Taxi dataset: 73.74%
- Percentage of solo trips for Green Taxi dataset: 79.60%

Task 8: Trip Duration and Distance Correlation

- Analyzes the correlation between trip duration and distance for Yellow Taxi trips in January 2023 using a scatter plot.
- There was a diverse array of trip duration to distance with some short distances having relatively long durations indicative of traffic or detours and some long distance trips having short durations indicative of lack of traffic on these roads.

Task 9: Taxi Pickups per Borough and Most Active Taxis

- Calculates and visualizes total pickups per borough for Yellow and Green Taxis, identifying the most active taxi type.

- The most active pickup boroughs for both taxi types are Manhattan, Queens, Unknown, Brooklyn, Bronx, EWR and Staten Island.

Task 10: Monthly Trip Counts

- Gathers month-wise trip counts for Yellow and Green Taxis and presents them in a bar plot to showcase monthly trends.
- The month with the most trips for both taxi types was January.

Challenges

Some challenges faces include:

- Merging the two datasets based on the LocationID of the boroughs
- Being able to extract data month-wise for plotting.
- Understanding the output of the graph

Conclusion

This was an enlightening analysis of trip data in the city of New York. Insight was gained on the trips made month-wise, distances covered, time duration, and things that could hinder the flow of traffic. This data can be used going forward in mapping the city, estimating traffic flow on road, and decongesting areas where traffic jams are prevalent.