DESB GRAND CHALLENGE 2015

Problem

The 2015 DEBS Grand Challenge competition aims to evaluate event-based systems in the context of real-time analytics over high-volume geospatial data streams. The underlying scenario addresses the analysis of taxi trips based on a stream of trip reports from New York City. Specifically, the 2015 Grand Challenge targets the following problems: identifying recent frequent routes and identifying high-profit regions. The corresponding queries demand the analysis of taxi location data along with other trip meta-data.

Data

The provided data consists of reports of taxi trips, including starting points, drop-off points, corresponding timestamps, and information related to the payment. Data are reported at the end of the trip, i.e., upon arrival in the order of the drop-off timestamps.

The specific attributes are listed below:

medallion	an md5sum of the identifier of the taxi - vehicle bound
hack_license	an md5sum of the identifier for the taxi license
pickup_datetime	time when the passenger(s) were picked up
dropoff_datetime	time when the passenger(s) were dropped off
trip_time_in_secs	duration of the trip
trip_distance	trip distance in miles
pickup_longitude	longitude coordinate of the pickup location
pickup_latitude	latitude coordinate of the pickup location
dropoff_longitude	longitude coordinate of the drop-off location
dropoff_latitude	latitude coordinate of the drop-off location
payment_type	the payment method - credit card or cash
fare_amount	fare amount in dollars
surcharge	surcharge in dollars
mta_tax	tax in dollars
tip_amount	tip in dollars
tolls_amount	bridge and tunnel tolls in dollars

Data at https://drive.google.com/file/d/0B4zFfvIVhcMzcWV5SEQtSUdtMWc/view?usp=sharing

Query 0: Data Cleansing and Setup

Load data in Apache Spark. The total size is about 12GB, you can use about 1GB of that for the project. The goal of this task is to remove the malformed data from the dataset, i.e., null or 0.0 columns, unknown licenses or drivers.

Query 1: Frequent Routes

Part 1

The goal of the query is to find the top 10 most frequent routes during the last 30 minutes. A route is represented by starting and ending grid cells. All routes completed within the last 30 minutes are considered for the query.

The output query results must be

```
start_cell, end_cell, Number of Rides
```

start_cell_id is the starting cell of the route, end_cell_id is the ending cell of the route.

Show only the 10 most frequent routes.

Part 2

Update the query above to conform to the following additional requirement:

- The query results must be updated whenever any of the 10 most frequent routes change.
- The output format for the result stream is:
- `pickup_datetime, dropoff_datetime, start_cell_id_1, end_cell_id_1, ... , start_cell_id_10, end_cell_id_10, delay`

Where pickup_datetime and dropoff_datetime are the timestamps of the trip report that resulted in an update of the result stream, start_cell_id_X the starting cell of the Xth-most frequent route, end_cell_id_X the ending cell of the Xth-most frequent route.

If less than 10 routes can be identified within the last 30 min, then NULL is to be output for all routes that lack data.

The attribute “delay” captures the time delay between reading the input event that triggered the output and the time when the output is produced. Participants must determine the delay using the current system time right after reading the input and right before writing the output. This attribute will be used in the evaluation of the submission.

The cells for this query are squares of 500 m X 500 m. The cell grid starts with cell 1.1, located at 41.474937, -74.913585 (in Barryville). The coordinate 41.474937, -74.913585

marks the center of the first cell. Cell numbers increase towards the east and south, with the shift to east being the first and the shift to south the second component of the cell, i.e., cell 3.7 is 2 cells east and 6 cells south of cell 1.1. The overall grid expands 150km south and 150km east from cell 1.1 with the cell 300.300 being the last cell in the grid. All trips starting or ending outside this area are treated as outliers and must not be considered in the result computation.

Query 2: Profitable Areas

This query aims to identify areas that are currently most profitable for taxi drivers. The profitability of an area is determined by dividing the area's profit by the number of empty taxis in that area within the last 15 minutes. The profit originating from an area is computed by calculating the median fare + tip for trips that started in the area and ended within the last 15 minutes. The number of empty taxis in an area is the sum of taxis with a drop-off location less than 30 minutes ago and no following pickup.

Part 1

The result stream of the query must be

```
pickup_datetime, dropoff_datetime, profitable_cell_id_1,
empty_taxies_in_cell_id, median_profit_in_cell_id, profitability_of_cell,
```

Report only the 10 most profitable areas

Part 2

The resulting stream of the query must provide the 10 most profitable areas in the subsequent format:

```
pickup_datetime, dropoff_datetime, profitable_cell_id_1,
empty_taxies_in_cell_id_1, median_profit_in_cell_id_1,
profitability_of_cell_1, ... , profitable_cell_id_10,
empty_taxies_in_cell_id_10, median_profit_in_cell_id_10,
profitability_of_cell_10, delay
```

With attribute names containing cell_id_1 corresponding to the most profitable cell and attribute containing cell_id_10 corresponding to the 10th most profitable cell. If less than 10 cells can be identified within the last 30 min, then NULL is to be returned for all cells that lack data. Query results must be updated whenever the 10 most profitable areas change. The pickup_datetime dropoff_datetime in the output are the timestamps of the trip report that triggered the change.

The attribute “delay” captures the time delay between reading the input event that triggered the output and the time when the output is produced. Participants must determine the delay using the current system time right after reading the input and right before writing the output. This attribute will be used in the evaluation of the submission.

Note: We use the same numbering scheme as for query 1 but with a different resolution. In query two we assume a cell size of 250m X 250m, i.e., the area to be considered spans from cell 1.1 to cell 600.600.

FAQ : http://www.debs2015.org/call-grand-challenge.html

Grading

Task	Points	Design Choice
Query 0:	5 (+1) points	Time Model, File Partitioning (+1 if using kafka)
Query 1 Part 1	5	
Query 1 Part 2	2.5	
Query 2 Part 1	5	
Query 2 Part 2	2.5	