DataEng: Project Assignment 3

Data Integration

Assignment date: February 16
Due date: February 28, 2021 @10pm PT
Submit: assignment submission form

Congratulations! By now you have a working, end-to-end data pipeline. Unfortunately, it does not have enough data to properly implement our Data Scientist's visualization. To fill out information such as "route ID" you need to access another source of data and build a new pipeline to integrate it with your initial pipeline. Here are your steps:

A. access the stop event data
B. build a new pipeline for the stop event data
C. integrate the stop event data with the bread crumb data
D. testing

A. Stop Event Data

Access C-Tran "Stop Event" data at this URL: http://rbi.ddns.net/getStopEvents As with the previous data source, this data set gives all C-Tran vehicle stop events for a single day of operation.

B. New Pipeline

Your job is to build a new pipeline that operates just like the previous one, including use of Kafka, automation, validation and loading.

C. Integrate Stop Events with Bread Crumbs

The two pipelines (BreadCrumb pipeline and StopEvent pipeline) must update the values in the Trip table such that all of the columns of both tables are filled correctly.

D. Visualization

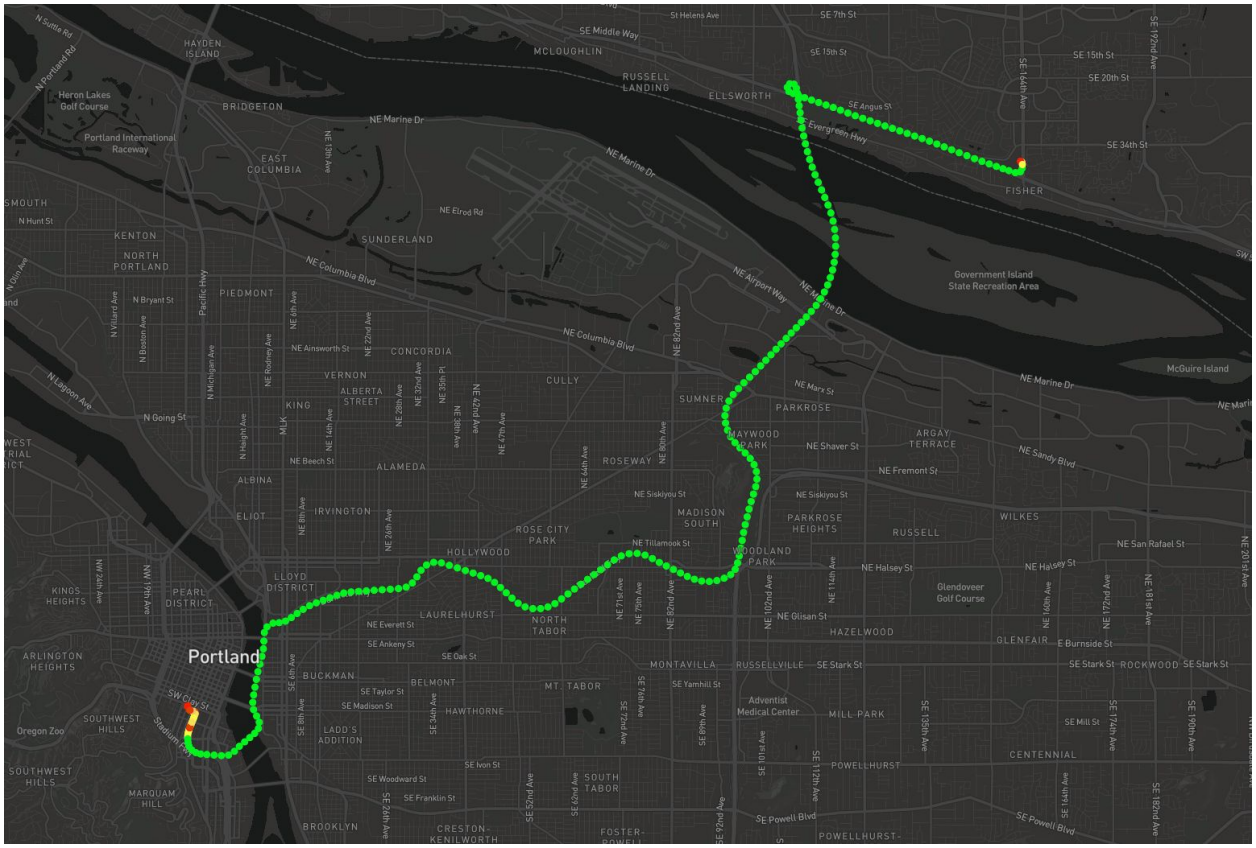
Aman developed a new visualization tool that allows you to view your bread crumb data and display it on a map. See Aman's description here. Your job is to integrate this tool with your database tables so that you can query the breadcrumb and trip data in your database server, transform to geoJSON format and display the resulting map visualization.

Submission

Make a copy of this document and update it to include the following visualizations. For each visualization extract from your database a list of {latitude, longitude, speed} tuples and then use the provided visualization code (see Section D above) to display bus speeds at all of the corresponding geographic coordinates. So, for example, if you are asked to visualize a “trip”, then you must query your database to find all of the {latitude, longitude, speed} tuples for that trip, and then display a map showing the recorded/calculated bus speed at each {latitude,longitude} location.

No need to produce software that neatly displays trips, routes, dates, times, etc. onto the visualization itself. Instead, just paste a screen capture of the map-based speed visualization into your submission document and then include a text description of the contents of the visualization. For example, text like this: “Bus Speeds for all outbound trips of route 65 between 9am and 11am on Sunday October 32, 2020.”

Visualization 1. A visualization of speeds for a single trip for any bus route that crosses the Glenn Jackson I-205 bridge. You choose the day, time and route for your selected trip. To find a trip that traverses this bridge, consider finding a trip that includes breadcrumb sensor points within this bounding box: [45.592404, -122.550711, 45.586158, -122.541270]. Any bus trip that includes breadcrumb points within that box either crosses the bridge or goes swimming in the Columbia river!
CODE:
SELECT latitude, longitude, speed FROM breadcrumb WHERE trip_id IN (select distinct trip_id from breadcrumb where latitude between 45.586158 and 45.592404 and longitude between -122.550711 and -122.541270 limit 1);



Visualization 2. All outbound trips that occurred on route 65 on any Friday (you choose which Friday) between the hours of 4pm and 6pm.

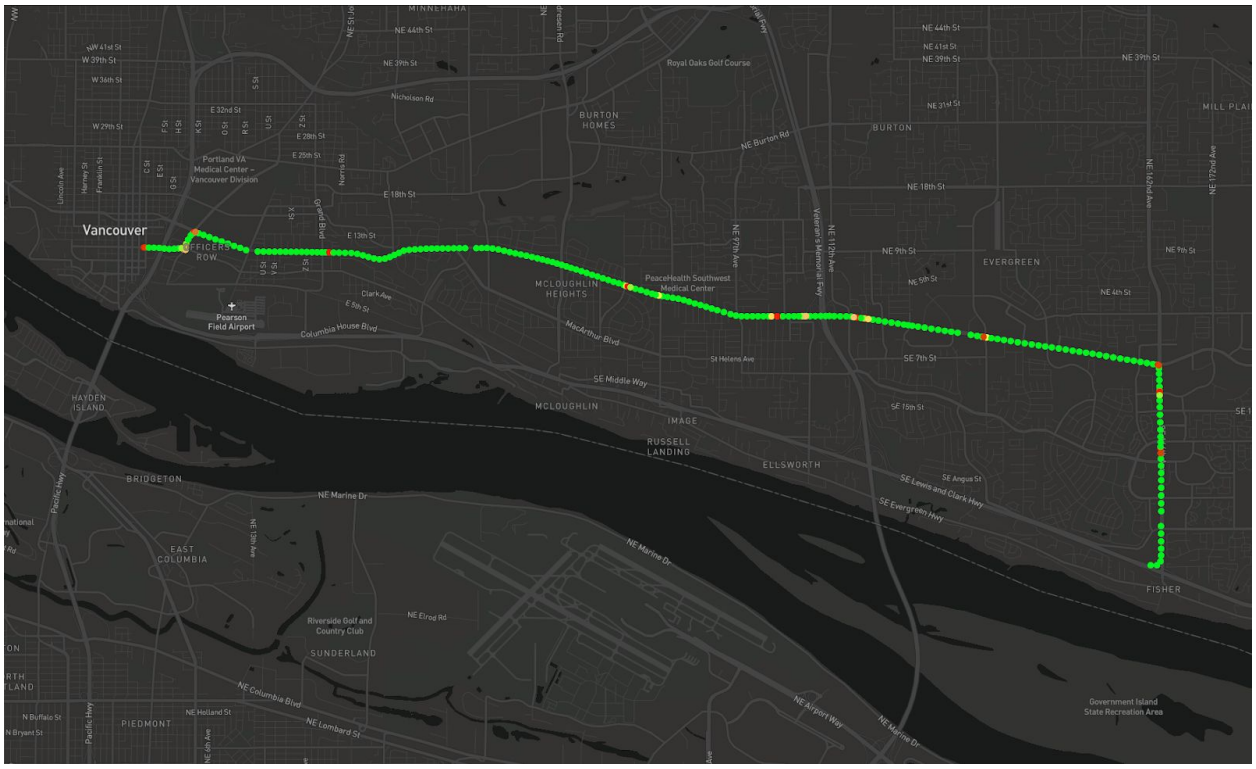
```
select b.latitude, b.longitude, b.speed from trip t inner join breadcrumb b on
b.trip_id = t.trip_id where b.tstamp
between '2020-10-16 16:00:00' and '2020-10-16 18:00:00' and route_id = 65;
```

Visualization 3. All outbound trips for route 65 on any Sunday morning (you choose which Sunday) between 9am and 11am.

```
select b.latitude, b.longitude, b.speed from trip t inner join breadcrumb b on
b.trip_id = t.trip_id where b.tstamp
between '2020-10-18 9:00:00' and '2020-10-18 11:00:00' and route_id = 65;
```

Visualization 4. The longest (as measured by time) trip in your entire data set. Indicate the date, route #, and trip ID of the trip along with a visualization showing the entire trip.

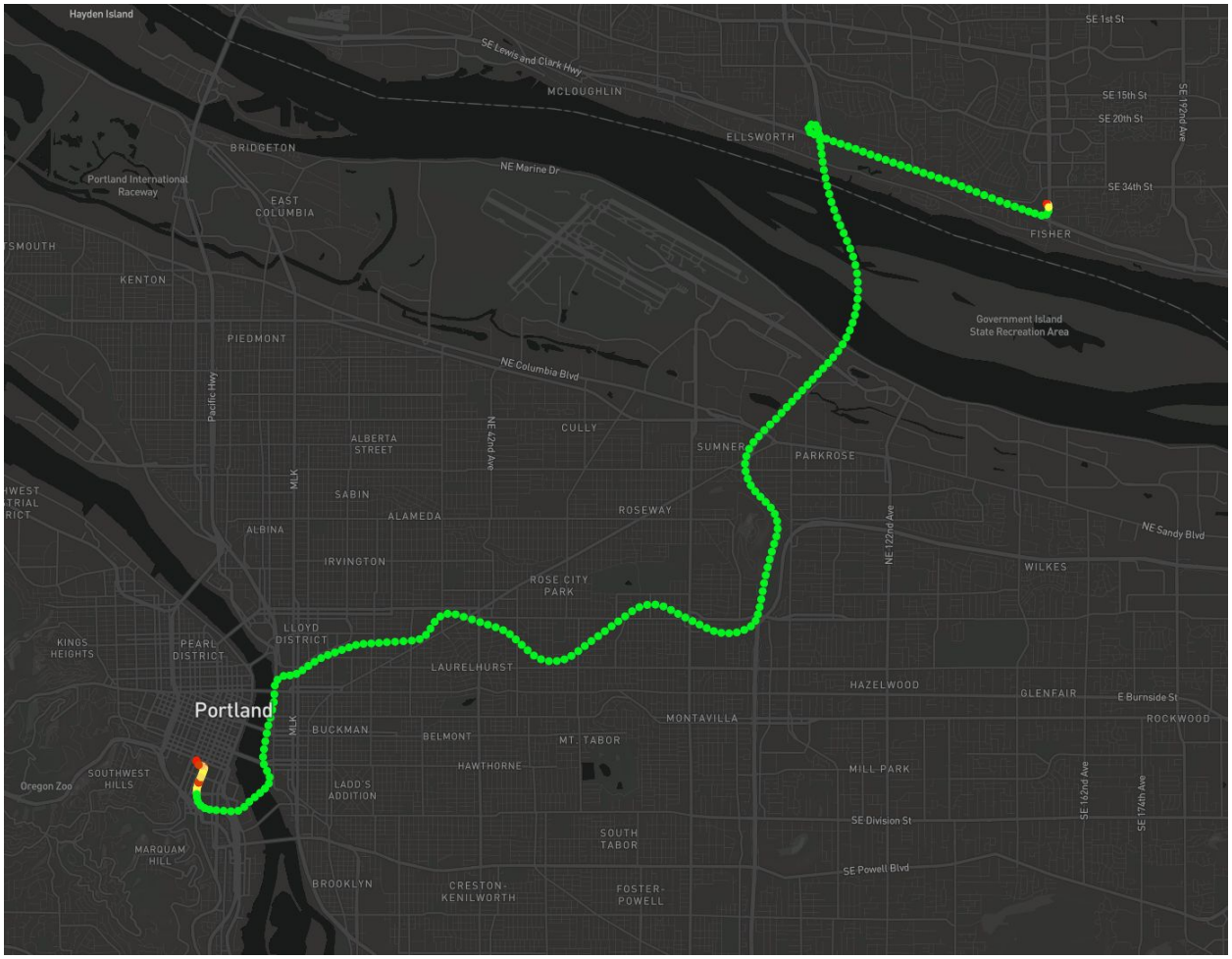
```
TRIP ID: 170129847
```



Visualization 5a, 5b, 5c, …. Three or more additional visualizations of your choice. Indicate why you chose each particular visualization.

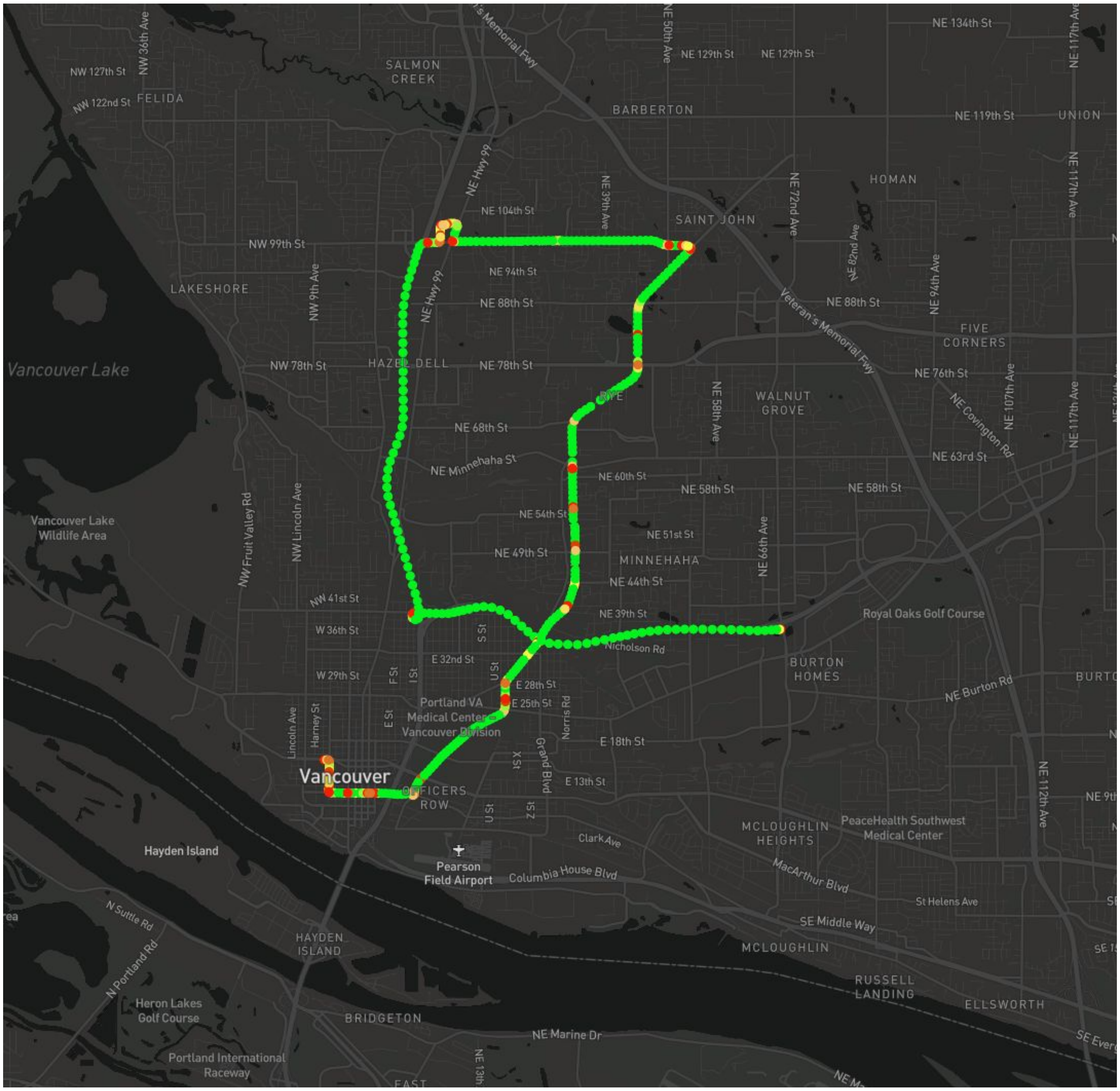
5a: What vehicle had the max speed and on which route

\copy (SELECT latitude, longitude, speed FROM breadcrumb WHERE trip_id IN (select distinct trip_id from breadcrumb where trip_id = 169050329)) TO 'visualizationa.tsv' with (format csv, header true, delimiter E'\t');



5b. What vehicle had the shortest route

```
\copy (SELECT latitude, longitude, speed FROM breadcrumb WHERE trip_id IN
(select distinct trip_id from breadcrumb
where trip_id = 169997629)) TO 'visualizationa.tsv' with (format csv, header
true, delimiter E'\t');
```


NE 134th St
NE 129th St
NE 129th St
NE 50th Ave
NW 127th St
NW 36th Ave
NW 122nd St
FELIDA
SALMON
CREEK
BARBERTON
NE 119th St
UNION
NE 117th Ave
NE Hwy 99
HOMAN
NE 104th St
NE 39th Ave
SAINT JOHN
NE 72nd Ave
NE 82nd Ave
NW 99th St
NE 94th St
LAKESHORE
NW 9th Ave
NE Hwy 99
NE 88th St
NE 88th St
NE 94th Ave
FIVE
CORNERS
Veteran's Memorial Fwy
Vancouver Lake
NW 78th St
HAZEL DELL
NE 78th St
NE 76th St
NE 107th Ave
WALNUT
GROVE
NE 58th Ave
NE Covington Rd
NE 68th St
NE 63rd St
NE Minnehaha St
NE 60th St
NE 58th St
NE 58th St
Vancouver Lake
Wildlife Area
NW Fruit Valley Rd
NW Lincoln Ave
NE 54th St
NE 51st St
NE 49th St
MINNEHAHA
NE 66th Ave
NE 44th St
NW 41st St
NE 39th St
Royal Oaks Golf Course
W 36th St
S St
Nicholson Rd
E 32nd St
W 29th St
F St
I St
U St
BURTON
HOMES
E 28th St
NE Burton Rd
Portland VA
Medical Center
Vancouver Division
E 25th St
Norris Rd
E St
Lincoln Ave
Harney St
E 18th St
X St
Grand Blvd
Vancouver
E 13th St
OFFICERS
ROW
NE 112th Ave
MCLOUGHLIN
HEIGHTS
PeaceHealth Southwest
Medical Center
Clark Ave
Hayden Island
Pearson
Field Airport
Columbia House Blvd
MacArthur Blvd
N Suttle Rd
St Helens Ave
SE Middle Way
HAYDEN
ISLAND
N Portland Rd
MCLOUGHLIN
RUSSELL
LANDING
Heron Lakes
Golf Course
ELLSWORTH
BRIDGETON
NE Marine Dr
Portland International
Raceway

Your Code

Provide a reference to the repository where you store your code. If you are keeping it private then share it with Bruce (bruce.irvin@gmail.com), David and Aman (github references TBD).

```
import geojson
import pandas as pd
import csv, json
from geojson import Feature, FeatureCollection, Point
import math


#!/usr/bin/python3
import csv, json
from geojson import Feature, FeatureCollection, Point
features = []
with open('visualizationa.tsv', newline='\n') as csvfile:
    reader = csv.reader(csvfile, delimiter='\t')
    data = csvfile.readlines()
    for line in data[1:]:
        row = line.split("\t")

        # Uncomment these lines
        lat = row[0]
        long = row[1]
        speed = row[2]

        speed = speed.strip()


        # skip the rows where speed is missing
        if speed is None or speed == "":
            continue

        try:
            latitude, longitude = map(float, (lat, long))
            features.append(
                Feature(
                    geometry = Point((longitude,latitude)),
                    properties = {
                        'speed': int(float(speed))
```

```
                }
            )
        )
    except ValueError:
        continue

collection = FeatureCollection(features)
with open("data.geojson", "w") as f:
    f.write('%s' % collection)
print(collection)
```