Rohit Ravishankar
rr9105@rit.edu

Project 1

CSCI - 720 Big Data Analytics

Collaborators: None

5.

a. Background section (three or four paragraphs), with references. Discuss the background of GPS: When did GPS come about? How many satellites are there? What other relevant details should we know about GPS? What else did you find out that is interesting about GPS?

Ans. GPS or Global Positioning System was initially owned by the US government and operated by the US Air Force. It provided geolocation & time information to any location on Earth.

It was launched in 1973 by the DoD(Department of Defense) and was initially for the period when scientist were able track movements in the satellite through radio signals, termed as doppler effect. This system became fully operational in the year 1995.

The number of active satellites (government & private) are 1,100. About 60% of the active satellites are used in communication with the US has the largest number of active satellites (502), followed by Russia (118) & China (116).

There are 3 main parts to any GPS system:-

- Satellites
- Receivers
- Ground Stations

The ground stations have control segment uses measurements collected by the monitor stations to predict the behavior of each satellite's orbit and atomic clocks. The prediction data is linked up to the satellites for transmission to users. The control segment also ensures that GPS satellite orbits remain within limits and that the satellites do not drift too far from nominal orbits. GPS satellites orbit in circular orbits at 17,440 km altitude, each orbit lasting 12 hours. The satellites are powered by solar cells to continually orientate themselves to point the solar panels towards the Sun and the antennas towards the Earth. Each satellite contains four atomic clocks. When you buy a GPS, you are actually buying only the GPS receiver and get free use of the other two main components.

Some interesting facts about GPS:-

- To determine an exact view of Earth, a group of 24 satellites work together.
- A GPS can also be used to determine the exact time to the fraction of a second since they use highly accurate atomic clocks.

References:

https://www.nasa.gov/directorates/heo/scan/communications/policy/GPS_History.html
https://talkingpointsmemo.com/idealab/satellites-earth-orbit
http://www.socialstudiesforkids.com/articles/geography/gps.htm
https://dpipwe.tas.gov.au/Documents/Worksheet%203%20-%20Applications%20of%20GPS.pdf
https://www.streetdirectory.com/travel_guide/12293/gps_vehicle_tracking/

b. Describe how you wrote your program. What program pattern did you use? Did you hold all the information in memory at once? Did you parse all of the files at once?

Ans. The steps I took to write the program:-

1. I created a list of all the input file names and iterated over the list of input file names. For each of the files, I performed the following tasks:-

   1.1. The first step, was to create the header of the output file (KML file) by calling the *emit_header()* method. The *emit_header()* created the output file and wrote the KML header to the output file and returned the output file pointer.

   1.2. The *convert_to_kml()* reads the input file name and the output file pointer created in the *emit_header()* method to emit the coordinates (latitude and longitude) to the output, viz., the KML file.

   1.3. Some steps are followed to pre-process the data. This includes, method calls to *convert_to_degrees_mins()* method to convert the degrees and minutes for the latitude and longitude values to only degrees. Another step in pre-processing the data was to read in only *"GPRMC"* data and hence, I ignored the data points which was not GPRMC. Subsequently, I deduplicated data from the list of all coordinates my using the *remove_duplicates()* function.

2. In the next step I read all the input data into the a list-of-lists which included data about the speed, time, tracking angle, latitude and longitude. The key difference between step 2 and what I followed in step 1.2 was that I read in a larger subset of data. I wanted to ensure separation-of-concerns and hence, refrained from creating one function to perform the task of writing into the KML and reading from the file. I used the entire data set to calculate the position of the stop signs and the left turns.
3. In order to calculate the position of the stop signs, I used temporal sequential analysis. The algorithm I built was based off of the algorithm mentioned on Prof. Kinsman's slides.
4. In order to calculate all the left turns, I modified my algorithm to find the stop signs to also incorporate the tracking angle. This modification helped me track the angle of the vehicle at any given point and hence deduce if the vehicle was making a left turn or a right turn.
5. The next step was be to call the *emit_trailer()* which closes all the tags that were opened in the header.

The programming pattern I used, was along the lines of the concept in HomeWork 5, a program that writes another program. I created the KML file and wrote all the coordinates and the stop sign placemarks as I was processing the data. The variation I used was 'Variation 1' as mentioned on Dr. Kinsman's slides, i.e., loosely coupled to replicate the *emit_header()*, *emit_body()* and *emit_trailer()* function.

I did not hold all the information in memory at once. I read each file sequentially and processed file-by-file and wrote the output file for one file before moving on to the next one.

c. How did you define a left-hand turn, and then detect it in the data? What problems did you find with the approach? Did you have to do any noise removal or signal processing?

Ans. I defined a left turn as one where tracking angle of the turn is greater than 50 degrees and less than 170 degrees, the speed of the turn is greater than 1 mph and the distance between last position in the list of left turns and the current left turn is less than 0.001degrees. An interesting point to note is that we use degrees to calculate the distance since, that is more universal than converting it into miles. This is because the distance between longitudes is different corresponding to different latitudes of earth.

I defined left turns as where the tracking angle changes by 50 to 170 degrees because left turns are wide turns while right hand turns are sharper. The problem I found was that there are often times false positive for right turns. This is because while getting on to ramps an an interstate, the ramps are designed in such a way that the tracking angle is wider and hence it gets caught with the window size set. No, I did not do any specific noise removal or signal processing. I had worked on removing noise when eliminating the the *GPGGA* records.

The problem I faced was deciding the window size and the tracking angle to consider for left turns so as to minimize the false positives.

d. How did you define a stop, and then detect it in the data? What problems did you need to overcome?

Ans. I defined a stop sign as any location where the car is traveling at a speed less than **7** miles per hour, the time difference between the previous stop sign and the current stop sign is greater than a minimum of **1s** apart and last stop sign was at least **0.01** degrees away. Again as mentioned previously, an interesting point to note is that we use degrees to calculate the distance since, that is more universal than converting it into miles. This is because the distance between longitudes is different corresponding to different latitudes of earth.

The problem I faced was deciding an appropriate window size to use to calculate the stop sign. The number of points I was considering to calculate the stop sign and the threshold values determined the number of stop signs I displayed along the way. I figured out the right values for the threshold by means of heuristics.

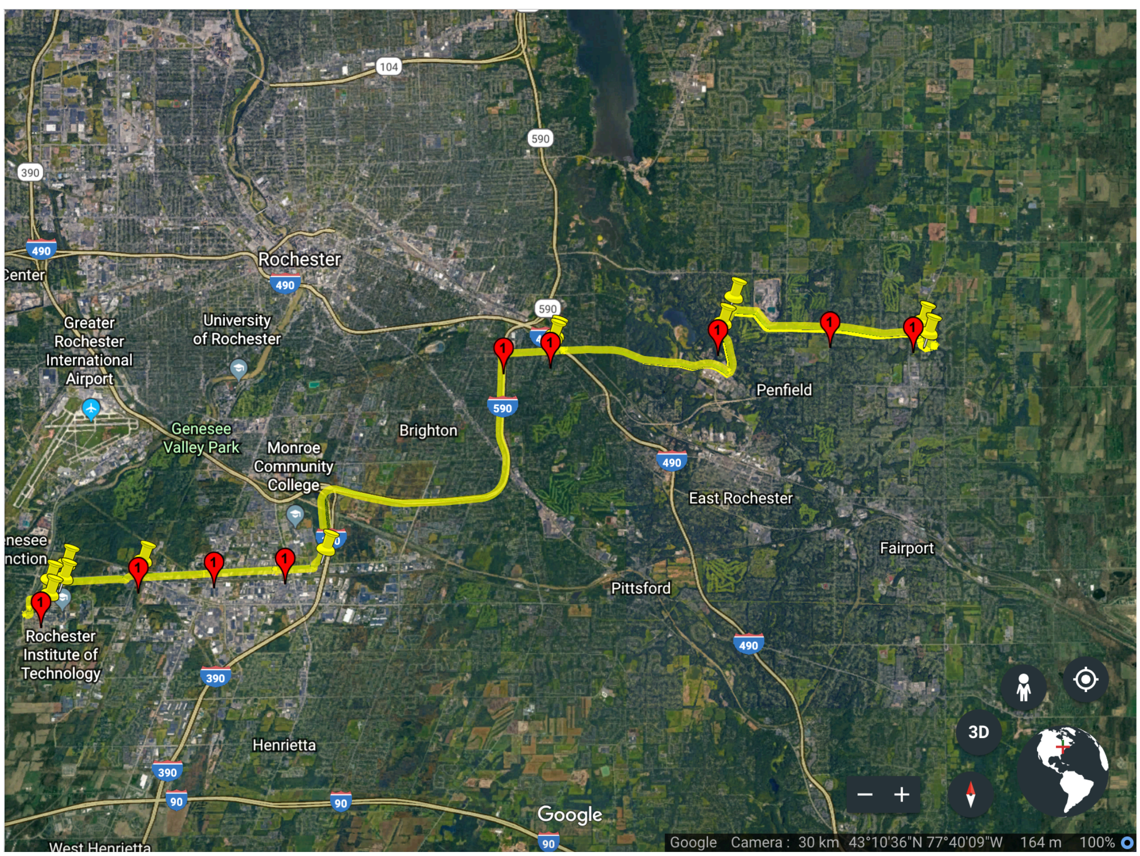
e. How did you define a stop, and then detect it in the data? What problems did you need to overcome?

Ans. *"ZIAA_CTU_2018_10_10_1255.txt"* with a cost function of *1.5615602222222245.*

I did not face any significant problems to calculate the cost function as I had pre-processed my data to convert the time from *hhmmss* to *s.*

f. A screen capture of image of the best track or trip to work, with annotations if you did them.

Ans.



g. A summary conclusion of what you learned overall, and how this might be useful for some commercial application.

Ans. I learnt how to confidently write a program to write another program. I learnt about how to pre-process my data better, especially for time and date data, since they may possibly be in different formats. I learnt about GPS, how to work with GPS data and how do you detect stop signs and left turns given speed, coordinates and turning angle metrics.

A useful commercial application this knowledge could be used for is in delivery logistics akin to the example Dr. Kinsman used in class about UPS drivers not being allowed to take left turns. I believe such data analysis would help reduce fuel cost and time for a lot of large scale retailers such as Walmart and Target and delivery agents such as UPS, Fedex etc.