Project Report - Ride Share optimization for taxi rides in Chicago - Proof of concept

Name of group members and CNetID:

1. Rahim Rasool - rahimrasool
2. Silky Agrawal - silkyagrawal
3. Tirumala Venkatesh Kaggundi - tiru

A brief overview of the final project:

For a person hailing a taxi in the city of Chicago (or in any given city), it is likely that the person might not mind sharing the cab/chartered bus with someone going to the same destination in the given time duration. However, there is no way for a person to know if such a possibility exists, and if yes, what are the compromises one has to make? For example, a shared cab may be available after 10 minutes, at a walking distance of 0.2 miles and which might dropoff at a point that is 0.1 miles away from the destination.

Uber (and similar ride aggregators) have tried ‘pool’ concept whereby a cab may pickup multiple passengers, and drop them off at different locations. However, the control in such cases lies with the aggregator. Once the passenger signs up for a ‘pool’, the time penalty might be severe if the detour is big/traffic laden for a specific passenger, and also due to the fact that the last mile connectivity comes with a penal cost for others in the pooled cab.

The public transport plies on identified roads and tracks. It doesn’t solve the last mile connectivity to most of those who are taking a taxi ride. In many instances, the taxi rides are taken in order to bridge the distance to the nearest metra/CTA stop. Thus there is an identified need, as per our data analysis which showed a very high frequency of small distance rides of less than 5 kilometers, to bridge this gap.

We have come up with an alternative to solve the above issue by analyzing the cab movement in the city for the last given period of time (say 3 weeks) and to find patterns which indicate the routes that are fit cases for clustering and launching pooled services. Regular and consistent cab movements between the same pickup and dropoff locations in a given period of time in the city indicate a pattern of movement that can be targeted for optimization through carpool arrangements or aggregated bus service. The data used for the study is the City of Chicago taxi ride that is available since 2013 to Feb 2021 and can be accessed using Google Big Query API. For our analysis, we extracted pre-COVID data for 3 weeks (with more than 800,000 rows) to find consistent patterns in the rides. The data was further pre-processed to deal with missing values and be ready for further analysis.

Seldom do two pickup and dropoff points match exactly, except if the point happens to be a popular destination. Therefore some means of clustering is required. We have tried various models (including some elementary ML approaches such as KMeans, Kmedoids, HDBSCAN etc) and have come up with our own simple, non ML based model, that bases the clustering on walkability to the clustered points at both pickup and dropoff for a passenger. More the walkability of people, greater the chance of finding a nearby cluster in the given period of time.

The routes are optimization using google ortools and the associated algorithms. Once the routes are found, the aggregation algorithms try to aggregate the routes for each pickup and dropoffs by way of arranging for an aggregated vehicle (cab/bus). This is followed by a visualisation of the routes with pickup and drops for each route using mapbox tool.

The issue of last mile connectivity is paramount, especially in a developing country context where there are challenges in terms of infrastructure for public transport, and we have tried to solve it by using aggregated service to the point of walkability of people from both pickup and dropoff locations.The concepts and model used here can be easily expanded and is easily transferable to any city.

The overall structure of the software (1-page maximum):

Data collection and cleaning part: We were able to access data from the Google Cloud Server using Big Query API. Given the limitations of our storage capacity (8GB RAM and 32-bit Python package, we were unable to download a full-3 month data that we had initially targeted but even 3-weeks pre-COVID data enabled us to download more than 800,000 rows which enabled us to identify consistent patterns in ride movements.

Clustering:

After trying various ML approaches, we have ended up with a simple data clustering method for our purpose, loosely based on KMedoid without using any ML package, but by using pandas alone. The user can input the acceptable walking distance, and based on this the clustering program would run to produce a citywide optimized clusters of routes that are available in any given hour based on the past taxi-movement data.

Route optimization and visualisation:

The route optimization is done using google ortools and the mapbox is use for distance api and mapping. The results are overlaid on a map for each route for a subset of data that shows the optimized route for the given hour in the chosen route.

Integration:

The parts of the program are integrated through an install.sh file that also take care of creating a virtual environment with packages to run the program as described in ReadMe.txt file.

What the project tried to accomplish and what it actually accomplished (200 words):

The project wished to find a complete solution to the transport aggregation and last mile connectivity issue for the city of Chicago and present an interactive visualisation alongwith details of individual cost savings and carbon savings for the society.

However, the lofty goals were tempered through the project once the development started, including the unfortunate departure of one of our colleagues who dropped the course, and finally we could achieve only part goals as shown below (which was a learning for us)

1. Data collection, cleaning and analysis of taxi movement in Chicago
2. Clustering of routes and optimized pickup and dropoff locations based on walkability and time
3. Route optimization between clusters
4. Approximate pooling possibilities
5. A command line interface
6. Visualization of routes on html
7. A complete route clustering and optimization proof of concept that can be scaled and transferred to other cities or larger contexts.

Limitations of the project/ Future potential developments:

1. We were unable to use a wider time frame than 3 weeks due to limits to our storage capacity. However, if we had time to work parallely, we could have explored the potential to use a larger time frame for our study.
2. We used list to access data from the Google API as our system did not support PyArrow that converts the query directly into pandas data frame. None of the PyArrow versions were supported on Python 32-bit which most of us used, and we were able to get 3 weeks of data using the list data structure. However, if we needed faster processing and had time, we could have explored if it were possible to run PyArrow on Python 64-bit more efficiently.
3. The optimization aspects may be further improved with more data and usage of some of the ML approaches in coming quarters.
4. The visualisation aspects shall be further improved going forward.
5. We can also calculate the number of cabs taken off the road and the carbon savings for the society using the route optimization.