Is Uber really cheaper than taxi?

Yahui Ke
kyahui3@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia

Xueyang Zhang
xzhang870@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia

Sikai Zhao
sikaizhao@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia

Hao Wang
hwang794@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia

Yaoxu Xiao
yxiao356@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia

ABSTRACT

Building and visualizing prediction model to analyze the price of Uber and Taxi is the main objective of this project. Detailed of proposed method and the upcoming experiments are provided in this progress report.

KEYWORDS

Uber, big data analysis, prediction model, visualization

1 INTRODUCTION

With the rapid development of Mobile Internet and booming of sharing economics, Uber has become one of the top choice for customers. However, the question of whether Uber is really cheaper than taxi or not needs more analysis to answer. This project is focus on building suitable model to predict the taxi price based on history data.

2 LITERATURE SURVEY

Apache Spark is one unified engine for diverse big data processing [1]. It was developed using Scala but can support multi-language APIs including Scala, Java, Python and R[2]. Spark uses Resilient Distributed Datasets (RDDs) to store data, which can be stored in memory between queries. This enables a large increase in data processing speed compared with MapReduce and DAG engines[3].

In chapter 10 of *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*[4], the author introduced a Python library which have calendar-related functionality. By using this datetime library, we can get week number by combine the string into date in Python. Also, we can use the build in function to convert between string and datetime.In Online Map Application Development Using Google Maps API, SQL Database, and ASP .NET, the author[5] shows how to get longitude and latitude using JavaScript which can be used to generate a location ID. In Point in polygon strategies[6], the author provides three strategies to decide whether a point is in a polygon or not, crossing test, angle summation and triangle fan. The crossing test is the simplest way which will be implemented in our algorithm to convert longitude and latitude into location ID.

A deep neural network was built by Simon[7] using six different features as input to predict the taxi fare using six different features such as time, distance. Rangapuram presents a forecasting method that parametrizes a particular linear State space models(SSM)[8] using a recurrent neural network (RNN) for time series problem. Ahmed provides a systematic review of the current time series forecasting model using neural networks[9]. Ahmed tries to define the most important theoretical contributions in the development of artificial neural network models for the forecasting of non-linear time series.

Cullen Schaffer used experiments to prove that cross-validation[10] leads to better performance that it guards against the chance of catastrophic performance. The popular method, 10-folder cross-validation, would be a suitable way to test the prediction model of our project with low variation. Vafeiadis also performed and evaluated five machine learning algorithms based on cross-validation and the Monte Carlo simulation[11] was used to determined the most efficient parameter combinations. Monte Carlo simulation can be used to find the influence of the parameters to the prediction results and model efficiency. Vora tested prediction models based on large scale dataset[12] and used Precision and Recall methods to evaluate the results. The parameters such as TP, FN can also be defined in our model to evade the performance of the algorithms. Traditional and modern clustering algorithms are evaluated by Xu[13] based on parameters of distance or similarity measurement and evaluation indicators.

Integrative Genomics Viewer (IGV) could efficiently interact data with different views[14]. Also, a zero-drift localization system[15] could help to localize and reuse the map in the already mapped areas with more accuracy. This zero-drift system will achieve in our NYC map with high precision in specific pick up and drop off location. It can also be used to match an object or a scene with obstacles and outliers from the data sets images. The approach of polygonal mesh reconstruction errors to obtain higher accuracy of this approach could apply in our visualization process[16].

3 METHODOLOGY

Data Cleaning

In our project, we choose to use Apache Spark for data cleaning and processing. The dataset we choose is the Yellow Taxi Trip Records in the year of 2018[17]. We will randomly divide our data up into 10 pieces and perform 10-fold cross-validation processing for training and testing. The original dataset contains useless extra columns, which include VenderID, Passenger_count, Store_and_fwd, Payment_type, MTA_tax, Improvement_surcharge and Tip_amount. Then we process the data cleaning by filtering out the invalid columns for our purposes. For example, we excluded the rows with RateCodeID == 5 as the fare for those trips are negotiated fare. We also excluded the rows with 0 or negative fare_amount/total_amount.Those data might indicates a refunded trip and thus is not useful for our purposes. Besides, to get the time information of each trip, we calculated the trip duration by substract the tpep_dropoff_datetime and tpep_pickup_datetime. Then we got the valid data we need as follow.

Sample Data:

tpep_pickup_datetime	tpep_dropoff_datetime	trip_duration	trip_distance	RatecodeID	PULocationID	DOLocationID	fare_amount	extra	tolls_amount	total_amount
2018-01-05 22:03:13	2018-01-05 22:11:16	485	90	1	148	211	7	0.5	0	9.3
2018-01-05 22:18:19	2018-01-05 22:35:46	1047	1.60	1	113	186	12	0.5	0	14.5
2018-01-05 22:39:21	2018-01-05 22:48:20	539	1.30	1	68	170	8	0.5	0	11.15
2018-01-06 22:55:12	2018-01-06 23:07:28	736	1.00	1	234	158	9.5	0.5	0	11.8
2018-01-05 22:05:07	2018-01-05 22:25:28	1221	3.30	1	161	148	15	0.5	0	16.3
2018-01-05 22:28:59	2018-01-05 22:46:26	1047	2.60	1	148	186	13	0.5	0	15.3
2018-01-06 22:47:57	2018-01-06 23:00:52	775	1.50	1	186	170	9.5	0.5	0	10.8
2018-01-06 22:20:29	2018-01-06 23:17:31	3422	19.85	1	132	255	65.5	0.5	0	66.8
2018-01-05 22:09:08	2018-01-05 22:25:16	968	2.80	1	48	164	12.5	0.5	0	15.3
2018-01-05 22:26:44	2018-01-05 22:37:15	631	1.40	1	164	230	8	0.5	0	11.3
2018-01-05 22:44:09	2018-01-05 22:48:53	284	.80	1	163	162	5	0.5	0	6.3
2018-01-05 22:59:23	2018-01-05 23:10:10	647	1.10	1	161	186	8.5	0.5	0	11.75
2018-01-05 22:11:50	2018-01-05 22:57:29	2739	16.09	1	132	129	49.5	0.5	0	50.8
2018-01-05 22:08:11	2018-01-05 22:15:21	430	1.00	1	163	143	6.5	0.5	0	10.14
2018-01-05 22:36:48	2018-01-05 22:50:21	813	.90	1	48	163	9.5	0.5	0	10.8
2018-01-05 22:54:43	2018-01-05 23:05:13	630	.96	1	230	161	8	0.5	0	9.3
2018-01-05 22:22:22	2018-01-05 22:36:36	854	2.33	1	186	237	11	0.5	0	15.38
2018-01-05 22:14:02	2018-01-05 22:16:26	144	.66	1	107	224	4	0.5	0	6.62
2018-01-05 22:32:57	2018-01-05 22:40:58	481	1.52	1	234	79	7.5	0.5	0	10.56
2018-01-05 22:31:58	2018-01-05 22:36:04	246	.90	1	152	166	5.5	0.5	0	8.15

Figure 1: Sample data after processing

Date&Time Clustering

Use python to import the modified data and use split, datetime function to generate a week id for training. Then extract the time and divide them into different groups by 15 minutes.

Model Building

1. Prediction model Features

In order to predict the taxi fare we need to use geometrical distance and travel time. According to NYC taxi fare model[18], the time of day, whether it's peak hour or it's a holiday also impact the fare. Also, rides to airport typically have different prices than normal. We introduce a feature called airport. The following describe all the features we will use for the taxi fare prediction:

Features Name	Description	Unit
Geometrical_distance	the geometrical distance from pickup location to drop off location	mile
Travel_time	total time takes for the ride	second
Week_Of_Year	Week of the year, value from 1 to 52	N/A
Time_Of_Time	Scale value to track time of the day, non busy hour: 1, busy peak hour: 2, night: 3	N/A
Airport_code	JFK 1, Newark 2	N/A
Pickup_Zone	NYC Taxi code	N/A
Drop_Off_Zone	NYC Taxi code	N/A

Figure 2: the features for taxi fare prediction

2. Training model
In this project, a specific Machine Learning library from Microsoft called LightGBM will be used to train the prediction model. The LightGBM has faster training speed and higher efficiency, lower memory usage, better accuracy. It also supports parallel and GPU learning which is capable of handling large-scale data[19]. The training will be happened in Microsoft Azure Cloud Computing Platform. Specifically, a LGBMRegressor will be used, and the following table shows the parameters need to be explored and its initial values when we are doing parameter turning for the prediction model.

Parameters	Initial value
learning_rate	0.05
num_leaves	5
n_estimators	800
max_bin	80
bagging_freq	5
bagging_seed	10
feature_fraction_seed	10
min_data_in_leaf	8
bagging_fraction	0.8

Figure 3: parameters and initial values

3. Prediction
User's input will be pickup and drop off location. The first step is to generate the prediction features for the model. It consists a few steps, such as converting input location into NYC Taxi Zone Id, getting the trip estimated time and geometrical distance using Here Map API. And then put all the input data into our trained model. The following figures show the workflow:

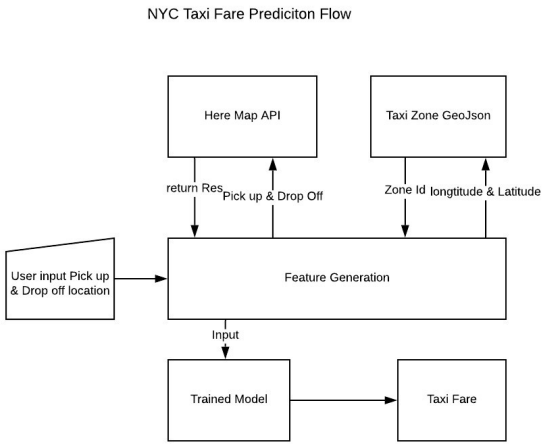

Figure 4: workflow

Testing

Considering the data amount and accuracy requirements, 10-folder validation will be used to test and evaluate the prediction model. At each round, 9 folder of data is used to train the module and the other used to test. Average of the 10 folder can indicate the performance of the module with low variance. Besides that, parameters of TP(true positive), TN(true negative), FP(false positive), FN(false negative) and other precision and recall parameters will be used to evaluate and describe the results of the experiments in a direct way.

Visualization

We will first visualization the different parameters based on the NYC map by using D3.js. We can get fundamental but meaningful information and use these outputs to further improve our prediction model. Furthermore, we will develop an interactive visualization application provides with customized details of rides and compared price of Uber and taxi.

4 UPCOMING EXPERIMENTS

In order to have a conclusion whether Uber taxi cost is cheaper than NYC Taxi cost. We have designed a three parts experiment to explore the result. The first part of experiment will generate 10000 trips in New York City by randomly choose pick up and drop off locations. Then each of the trip will using the trained prediction model to get an predicted price, and we will compare the costs of taxi and Uber by using these 10000 trips and Uber estimation API. We could generate two lines of the 10000 trip cost for Uber and taxi, it will clearly shows the cost difference. Meanwhile, a cost more ratio will calculated to give a straightforward result as follow.

Uber cost more ratio	Taxi cost more ratio
PLACE HOLDER	PLACE HOLDER

Figure 5: Uber and taxi cost radio

The second part of experiment will try to explore the impact of different months. For example, NYC will have heavy snow during winter months, which will influence the cost of Uber. In this experiment, we will randomly pick 1000 records of Uber 2018 open trips for each month. Then we will use the trip information and trained taxi fare prediction model to predict a taxi fare. Thus each months trips cost can be compared between Uber and Taxi in NYC as follow.

Moth	Uber cost more ratio	Taxi cost more ratio
Jan	PLACE HOLDER	PLACE HOLDER
Feb	PLACE HOLDER	PLACE HOLDER
Mar	PLACE HOLDER	PLACE HOLDER
Apr	PLACE HOLDER	PLACE HOLDER
May	PLACE HOLDER	PLACE HOLDER
Jun	PLACE HOLDER	PLACE HOLDER
Jul	PLACE HOLDER	PLACE HOLDER
Aug	PLACE HOLDER	PLACE HOLDER
Sep	PLACE HOLDER	PLACE HOLDER
Oct	PLACE HOLDER	PLACE HOLDER
Nov	PLACE HOLDER	PLACE HOLDER
Dec	PLACE HOLDER	PLACE HOLDER

Figure 6: different months uber/taxi cost radio

The third part of the experiment will consider the time of trip happened, as traffic has specific patterns, such as peak and non peak hours, weekend and workdays. In this experiment, we will randomly pick 1000 trips for different time and date, and then compare the Uber and taxi fares in these different time and date as follow.

Pick up Time	Uber cost more ratio	Taxi cost more ratio
Peak Hour		
Non Peak Hour		
Weekend/Holiday		
Workday		

Figure 7: different time uber/taxi cost radio

Innovation :

1. Compare fare cost in different months and different time.
2. Converting Pick up and Drop off longitude and latitude into Taxi Zone using NYC Taxi Zone GeoJson Data.

5 PLAN OF ACTIVITIES

All team members contribute similar amount of effort.

Xueyang Zhang: Data cleaning and processing
Hao Wang: Date&Time clustering & visualization
Yahui Ke: Draft prediction model building
Sikai Zhao: Model algorithm testing & visualization
Yaoxu Xiao: Model algorithm testing & visualization

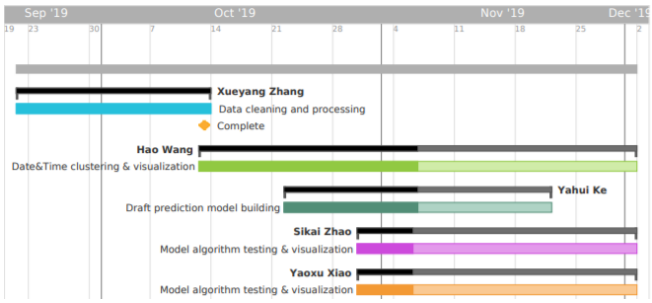

Figure 8: Project Progress

REFERENCES

[1] Zaharia, Matei, et al. "Apache spark: a unified engine for big data processing." Communications of the ACM 59.11 (2016): 56-65.

[2] Armbrust, Michael, et al. "Scaling spark in the real world: performance and usability." Proceedings of the VLDB Endowment 8.12 (2015): 1840-1843.

[3] Gopalani, Satish, and Rohan Arora. "Comparing apache spark and map reduce with performance analysis using k-means." International journal of computer applications 113.1 (2015).

[4] McKinney, Wes. *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython.* " O'Reilly Media, Inc.", 2012: chapter 10 Time Series

[5] Haines, Eric. "Point in polygon strategies." *Graphics gems IV* 994 (1994): 24-34.

[6] Hu, Shunfu, and Ting Dai. "Online Map Application Development Using Google Maps API, SQL Database, and ASP .NET." *International Journal of Information and Communication Technology Research* 3.3 (2013)

[7] Upadhyay, Rishabh Lui, Simon. (2017). Taxi Fare Rate Classification Using Deep Networks.

[8] Rangapuram, Syama Sundar and Seeger, Matthias W and Gasthaus, Jan and Stella, Lorenzo and Wang,

[9] Ahmed Tealab,Time series forecasting using artificial neural networks methodologies: A systematic review, Future Computing and Informatics Journal, Volume 3, Issue 2, 2018, Pages 334-340,ISSN 2314-7288.

[10] Schaffer, Cullen. "Selecting a classification method by cross-validation." Machine Learning 13.1 (1993): 135-143.

[11] Vafeiadis, Thanasis, et al. "A comparison of machine learn- ing techniques for customer churn prediction." Simulation Modelling Practice and Theory 55 (2015): 1-9.

[12] Vora, Deepali, and Kamatchi Iyer. "Evaluating the Effective- ness of Machine Learning Algorithms in Predictive Mod- elling." International Journal of Engineering Technology 7.3.4 (2018): 197-199.

[13] Xu, Dongkuan, and Yingjie Tian. "A comprehensive survey of clustering algorithms." Annals of Data Science 2.2 (2015): 165-193.

[14] Helga Thorvaldsdóttir, et al. "Integrative Genomics Viewer (IGV):high-performance genomics data." Briefings in Bioinformatics, Volume 14, NO 2.(2013):178-192.

[15] Raúl Mur-Artal, Juan D.Tardós. "Visual-Inertial Monocular SLAM With Map Reuse." IEEE Robotics and Automation Letters, Volume 2, Issue 2 (2017):796-803.

[16] Yasutaka Furukawa, Jean Ponce. "Accurate, Dense, and Robust Multiview Stereopsis." IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume: 32, Issue 8 (2010)

[17] <https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page>, accessed 10 Nov 2019

[18] <https://www1.nyc.gov/site/tlc/passengers/taxi-fare.page>, accessed 10 Nov 2019

[19] <https://github.com/microsoft/LightGBM>, accessed 10 Nov 2019