NYC Taxi Rides: Fare and Duration Prediction

Himanshu Jaiwal
A53216658
hjaiswal@ucsd.edu

Tushar Bansal
A53211130
tbansal@ucsd.edu

Prateek Jakate
A53214137
pjakate@ucsd.edu

Tejas Saxena
A53202646
tsaxena@ucsd.edu

***Abstract*—This report presents an evaluation study of various models and features to predict the duration and fare of a taxi trip in New York City. We investigate the optimal model and hyper-parameters for this problem. We show that Decision Trees based models, especially Gradient Boosting are effective in learning the predictive patterns from the given features. We examine the effective ways to represent the features and also study the importance of various features in our predictive models. We achieved lowest RMSE of 4.87 minutes for duration prediction using the Gradient Boosting model and RMSE of $2.49 for the fare prediction problem using the Ensemble of Random Forest and Gradient Boosting model. Looking at results, we can claim that though the model was not completely successful in predicting the exact duration or fare, it can still be used as an effective tool to give an approximate estimate.**

I. Introduction

In this analysis, we try to address some of the long standing problems in the transportation industry i.e. to predict the duration and fare at the beginning of the trip. With the advent of technology based cab services, this problem has become particularly important for the drivers and the customers. A good prediction mechanism can be instrumental for drivers in optimizing their returns, while also saving the customers from the uncertainties attached to a trip.

In our analysis, we attempt to examine and understand the provided features to create a prediction model for this problem. We study the impact of various features and also attempt to find the best possible ways to leverage those features. We review the performance of various models and discuss the effectiveness and shortcomings for these models.

In this report, we discuss four models: Ridge Regression, Random Forest, Gradient Boosting and an ensemble of Random Forest & Gradient Boosting. We evaluate these models based on the Root Mean Square Error (RMSE). We also discuss the importance of various features in our prediction algorithms.

II. Data

The data provides the details of yellow taxi rides in the New York City from Jan 2016 to June 2016. This data is provided by the NYC Taxi and Limousine Commission (downloaded from Kaggle).The data for each month is about 1.8GB and consists of roughly over 10 million trips. Each trip records fields: pick-up and drop-off dates/times, pick-up and drop-off coordinates, trip distances, itemized fares (fare, toll, tax and tip amounts), payment types, and driver-reported passenger counts.

From a preliminary analysis on this combined 6 month data, it was evident that there were few discrepancies in the dataset. For some cases the duration and fare were zero or negative; for some the ratio of duration by distance was unreasonably high (22hrs/0.01 mile) or low (5mins/20mile). A lower bound of speed (two mile per hour) and an upper bound (60 miles per hour) were used in our analysis to make our models less vulnerable to these outliers. We also remove the cases where the trips with payment type 4/6 as these trips were disputed/voided. After all this data cleaning, we randomly select 300k trips to create a new dataset for our analysis.

Fig. 1 demonstrates the pick-up (blue) and drop-off (red) locations of 2500 trips from our data.

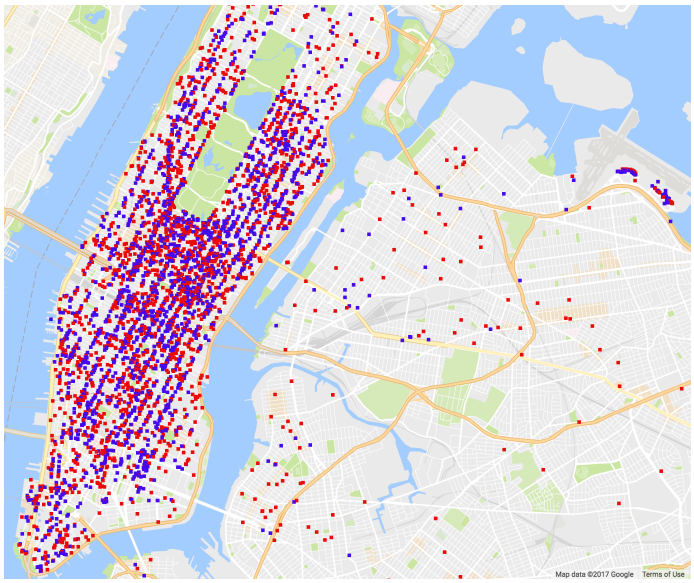

Fig. 1: Mapping of pick-up and drop-off locations

A. Exploratory Analysis

To better understand the problem at hand and the features, we perform an exploratory analysis on the data. The purpose of this analysis is to get insights on how the prediction variables behave with various features and how can we leverage these features in our models to achieve best possible results.

1) Distance: Fig. 2 demonstrates a positive correlation between trip distance and duration. An interesting finding is that the variance of duration increases, as the trip distance increases.

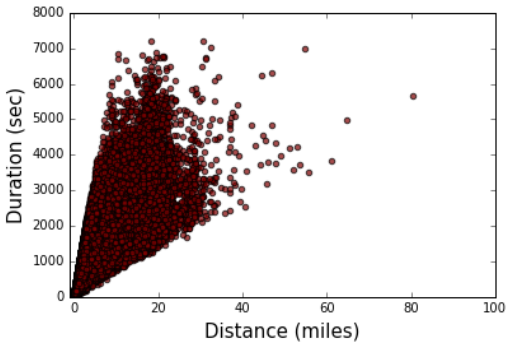

Fig. 2: Distance vs Duration

From Fig. 3, we can observe that the ratio of duration/distance, pace, goes down with increasing distance. This can be attributed to the fact that longer distances usually involve trip rides to/from areas outside the central city, thus involving travel on less congested routes. therefore increasing the average speed. In the analysis ahead, we'll see that trip distance is a very important feature for our models.

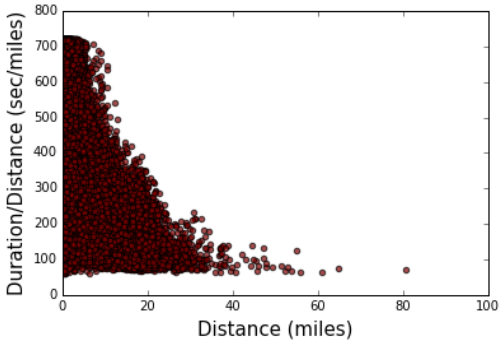

Fig. 3: Mean of Duration/Distance ratio vs Trip Distance

2) Time of day: The time of the day is an important factor to determine the duration a trip can take. The same trip during rush hours can take much longer compared to non-rush hours. In Fig. 4, we plot the mean Duration/Distance ratio with the pick-up hour of the day. We observe that the average travel speed is slowest from 11am to 4pm in the day and fastest from 12am to 5am. The impact of this feature can be particularly significant for longer trips.

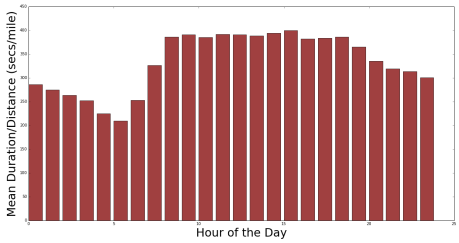

Fig. 4: Mean of Duration/Distance ratio vs Pick-up hour of the day

3) Day of week: The day of the week can have a significant impact on the predictions because we expect the weekdays to be more congested than the weekends especially during the day time. From Fig. 6, we can clearly see that the average speed is higher on the weekend than the weekdays.

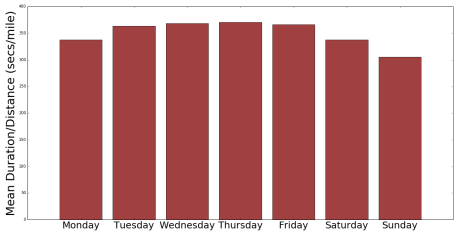

Fig. 5: Mean of Duration/Distance ratio vs Day of week

4) Location: An important aspect of this prediction task is the pick-up and drop-off locations. As discussed above, the distance factor is indeed import to determine the duration but it is also important to consider the exact route the taxi is taking. Though we might not have the exact routes of the trip but the pick-up and drop-off coordinates can act as a proxy for this feature.

An important aspect of this problem was to use the pick-up and drop-off coordinates effectively. Directly feeding them to the model would have made less sense as the coordinates could have potentially infinite possible values and with the limited number of cases given in the training set, it would have been difficult to extract relevant patterns. Also, it can be argued that trying to fit a linear/quadratic regressors for the coordinates wouldn't be a good idea as the traffic/demographic hardly change that way. To tackle this problem, we use k-means clustering to create 40 clusters using the pick-up and drop-off coordinates. This is covered in more detail in the features section.

In Fig. 5, we plot the mean of Duration/Distance ratio for trips starting in each cluster. We can clearly observe that the ratio varies drastically for different clusters, which cements our hypothesis that coordinates can be an effective feature.

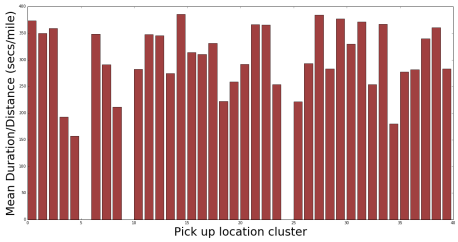

Fig. 6: Mean of Duration/Distance ratio vs Cluster

B. Fare

As expected, we can see from Fig. 7 that the fare has a strong positive correlation with the trip distance. In real life

situations too, the fare of a trip is a linear function of the duration and the distance parameter. So given the distance of a trip, if we can create a good model for predicting the duration, it is likely that the same model will be effective in predicting the fare of the trip too.

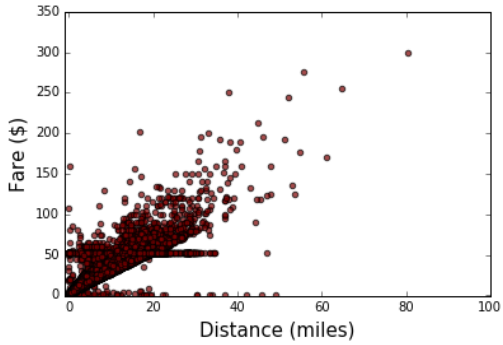

Fig. 7: Fare vs Trip Distance

III. Predictive Task

We are predicting the duration of a NYC Yellow Taxi ride and its fare amount (without taxes, toll and tip amount) that user will have to pay at the end of a ride. Since predicting the duration directly is a lot more complicated because of its huge variance, we chose to predict the ratio of duration and distance in our model. This ratio, called pace, was later multiplied with distance to predict the duration. Once, the duration was predicted, it was used along with the other features to predict the fare of a ride. For the purpose of training models, tuning hyperparameters and reporting our results, we split the data into three sets, training set (60%), validation set (20%) and test set (20%). We defined the baseline for predicting pace as the average of all pace values in the training set. This average pace value is then multiplied with distance to predict the duration of a ride with baseline model. Similarly, for fare prediction, average value of fare/distance ratio was calculated over the training set as a baseline model and this value can be multiplied with distance to predict the fare with baseline model. Root Mean Square Error (RMSE) was used as a metric to measure of our model's accuracy. It had the advantage of being convex and physically interpretable (it has the same unit as time for duration prediction and money for fare prediction). To validate the usefulness of our model, RMSE values for baseline model and our models were calculated on the test set and compared. We also compared our model's error on the test set with prior work of the same kind on the same data set.

IV. Features

Only the features which are available before a trip starts were used to predict the duration and fare. We incorporated the distance, pick-up date, time and location, drop location and toll amount (on path from source to destination). From the exploratory analysis done above, we can infer few valuable insights about the feature representations:

- Distance should be inverted before training a linear regression model. This can be inferred from the fact that we try to predict the factor duration/distance in our model, which is directly proportional to the 1/distance feature. So directly using the distance feature might not as effective in the model as the 1/distance feature.
- Features like month, weekday and hour of day do not display a well defined relationship to pace and should therefore be represented in the form of one hot encoded vectors. As all these features are categorical in nature, representing them in their absolute form could be a problem of models like linear regression. One hot encoding provides an efficient method to capture patterns corresponding to all these categorical features.
- Representing the pick-up and drop-off coordinate data is an important aspect of feature representations. We can improve this representation significantly using a number of ways (like mapping to zip codes), which are also discussed in detail in the literature section. We opted for of creating clusters on the given pick-up and drop-off coordinates. This helped us significantly especially in linear models as the information that couldn't be captured in a linear way could now be easily associated with different categories. We tried using this cluster representation in two ways: (i) Directly use the pick-up and drop-off clusters. (ii) Input the combination of pickup-dropoff cluster as one hot encoded vector (|C|*|C|).

To represent the pickup and drop location, the latitude, longitude pairs were first partitioned into clusters and a one-hot vector was then used to represent the cluster Id. During, the experimentation, it was discovered that just representing the pickup and drop locations as separate features was not good enough. To capture conditions such as traffic between different locales, one hot encoding was used to represent each combination of pickup and drop cluster Ids. The choice for the number of clusters used for this task depends on the trade-off between reconstruction error and feature size (as the feature size grows with number of clusters in a square fashion). A graph was plotted between number of clusters and reconstruction error to reach the final value 40.

For fare prediction, duration of the ride was added as an additional feature to the existing set of features.

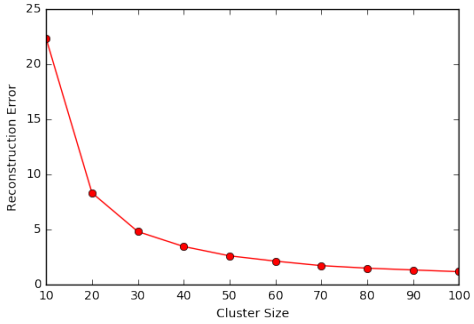

Fig. 8: Reconstruction error vs Number of Clusters

V. Models

A. Baseline

The RMSE for the Baseline model on the test set was obtained as 14.89 minutes for duration prediction and $12.917 for fare prediction.

We then use regression models to predict the duration and the fare and beat the baseline. This is because we want to estimate values which are real numbers. Moreover, since the problem statement involves a prediction and not a classification, it is more suited to a Regression model.

B. Ridge Regressor

We started with using the most basic model which is linear regression, but it was overfitting on the training set leading to very high values of weights. To overcome this overfitting, we decided to use the Ridge Regressor. The Ridge Regressor solves a regression model where the loss function is defined as the linear least squares function. In order to avoid overfitting to the training set, regularization is performed using the l2-norm.

$$Loss function : \|A\mathbf{x} - \mathbf{b}\|^2 + \lambda\|\mathbf{x}\|^2$$

For both the duration and fare prediction, the hyper-parameter λ was optimized. This was performed by plotting the validation RMSE against the increasing values of λ and the value for which the validation RMSE was lowest was chosen. The optimal value of λ was chosen as 0.3 for both the fare and duration prediction.

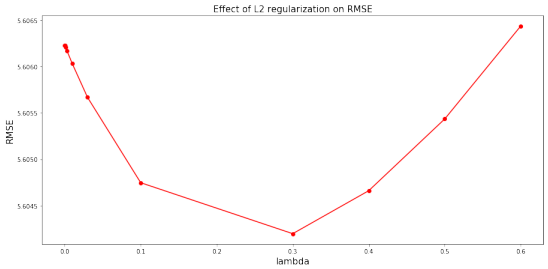

Fig. 9: Validation RMSE vs Lambda (Duration Prediction)

C. Random Forest Regressor

The features being used apart from distance and the toll amount were categorical in nature. That's why we perform one hot encoding on the pickup hour, day, month, year as well as on the coordinate clusters. As an effective way to deal with these categorical features, we have implement a random forest regressor for prediction. A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

This model prevents overfitting by itself, but there is an issue of scaling. We know that the validation RMSE will tend to reduce as the training set size is increased. However, we need to choose a reasonable size of the training set beyond which the validation RMSE does not significantly reduce. To see this, we plot the validation RMSE vs the training set size.

Fig. 10: Validation RMSE vs Training Set Size (Duration Prediction)

In order to set the hyper-parameters which are the number of trees and the max-depth of a tree, we plotted the variation of validation RMSE with the number of trees and the max-depth. A value of 10 was used for both the number of trees and the max-depth. Increasing the values for these parameters would have reduced the RMSE further but it was noticed that the reduction was not significant to warrant the scaling. This is shown in the plots of the validation RMSE vs these hyper-parameters.

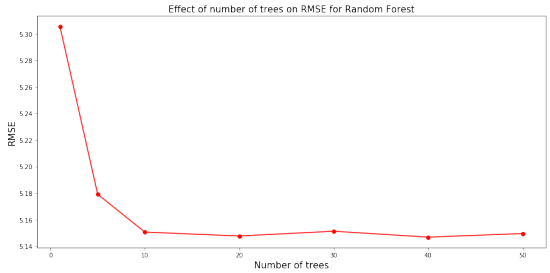

Fig. 11: Validation RMSE vs Number of Trees (Duration Prediction)

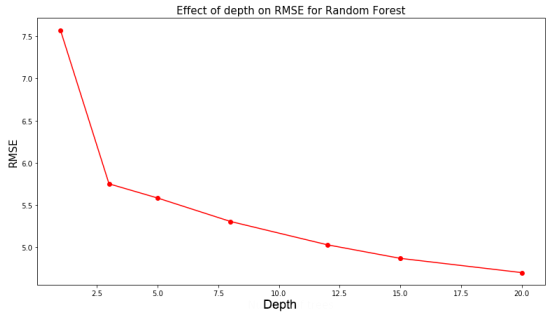

Fig. 12: Validation RMSE vs Max Depth (Duration Prediction)

D. Gradient Boosting Regressor

Gradient Boosting builds an additive model in a forward stage-wise fashion. It allows for the optimization of arbitrary differentiable loss functions. In each stage a regression tree is fit on the negative gradient of the least squares regression loss function.

Similar to the Random Forest Regressor, we need to optimize the values of number of estimators and the max-depth. We select their values as 30 and 10 respectively using random search such that the reduction in RMSE with further increase in hyper-parameters is not significant.

E. Ensemble of Random Forest and Gradient Boosting Regressor

In this model, we take the mean of the predicted values that we obtain using the Random Forest Regressor and the Gradient Boosting Regressor. We then use these mean predicted values to compute the RMSE for the test set.

F. Comparison of Feature Representations

In order to identify which features were most effective for the duration prediction we have removed each feature one by one from the set of features and observed the change in the validation RMSE in minutes for Ridge Regressor and Random Forest Regressor. This has been shown in the table for duration prediction.

Feature Removed	Ridge	Random Forest
Day of the Month	5.655	5.1528
Month	5.6925	5.1613
Hour	5.7481	5.4289
Day of the Week	5.6751	5.2078
Toll Amount	5.696	5.173
Distance	5.676	**9.932**
Pickup and Drop Clusters	**11.99**	5.167

We also observed that taking the combination of pickup and drop locations as a feature instead of taking them separately resulted in significant improvement in performance. This could be physically interpreted as follows: when we use the pickup and drop location separately, we are trying to model the duration of the ride based upon how these affect individually. It is like saying that every ride that originates at a location A and every ride that terminates at a location B are related to the duration in some manner. This may capture the traffic at these individual locations but might not be effective in capturing the relation between them. On the other hand, when we use the combination of pickup and drop locations as a feature, we are modeling the traffic between location A and location B as a whole, resulting in more information capture and better results. Though, this information comes at the price of increased feature size.

VI. Literature

The dataset we used has been taken from the kaggle website [7]. Dataset has data of taxi rides with taxis of NYC Taxi and Limousine Commission. Other works based on the same dataset that we used for survey and comparative analysis are as follows:

[1] uses one month of NYC taxi rides data. The model infers the possible paths for each trip and then estimates the link travel times by minimizing least squared difference between expected path travel times and the observed path travel times. In addition, the time and day of the rides were used as the most important features in the prediction model. The model is evaluated using a test network from Midtown Manhattan. They constructed an optimization problem using these computed features to reduce the prediction RMSE. The optimization problem is solved using Levenberg–Marquardt method. The RMSE is compared for different days and time combinations and the lowest RMSE achieved was 1.034 minutes. Similar to their model, we implemented the day of the week and time of the day features which were very effective in our models too. The state-of-the art feature estimation by predicting paths and using them to compute the travel time was not incorporated into our solution and can be considered for further improvement. It is evident from this analysis that the location and path plays a very important role in this prediction problem. However, due to time and resource constraints, we opted for a much simpler representation (clusters) to incorporate these factor.

[2] proposes a model that allows users to visually query taxi trips. Besides standard analytics queries, the model supports origin-destination queries that enable the study of mobility across the city. This model is able to express a wide range of spatio-temporal queries, and it is also flexible in that not only can queries be composed but also different aggregations and visual representations can be applied, allowing users to explore and compare results.

[3] uses Linear Regression and a Random Forest Regressor to predict the duration and fare for rides. Most of the streets and avenues in Manhattan are aligned in a grid structure because of this they have used co-ordinate transformation. We have additional features of pickup and drop clusters which were not included in their study. Note that this report only uses 1 month of the NYC taxi data. Comparison of our RMSE with this report is given as:

Model	**Proposed RMSE Duration (mins)**	**RMSE Duration (mins) [3]**
Linear Regression	5.656	6.51
Random Forest	**5.152**	5.24

Model	**Our Model Fare RMSE ($)**	**RMSE Fare ($) [3]**
Linear Regression	2.5086	3.52
Random Forest	2.496	**2.28**

VII. Observations and Results

We obtained very interesting insight when we plotted our cluster centroids over the map of New York. As cluster centroids are representative of the areas from where most of the pick ups and drop offs of the rides occur. We got cluster centroids mapped to airports, posh areas and city parks viz. John F. Kennedy International Airport, LaGuardia Airport,

Times Square, Columbia University, Barclays Center, Whitney Museum of American Art etc. as can be seen in Fig 13.

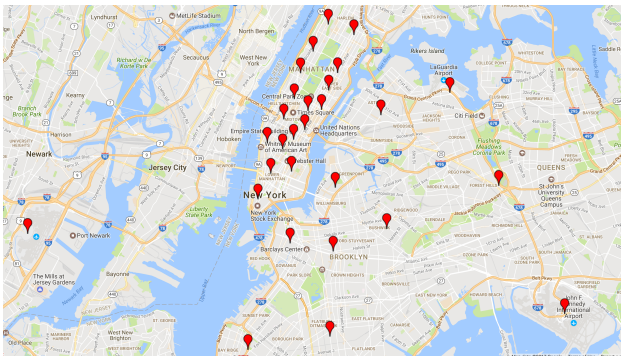

Fig. 13: Cluster Centers in New York

Most important features observed for Duration prediction are distance, day of the week, hour of the day, cluster features and toll amount. From Random forest, we obtained the importance of features as follows:

Feature	**Random Forest Feature Importance**
Distance	0.759788
Week day(Sunday)	0.038566
Week day(Saturday)	0.013711
Hour of the day(6am)	0.013423
Hour of the day(5am)	0.012210
Hour of the day(4am)	0.012113
Hour of the day(3am)	0.011951
Hour of the day(2am)	0.010415

RMSE values with different models for duration prediction and fare prediction calculated with most important features are as follows:

Model	**RMSE Duration (mins)**	**RMSE Fare ($)**
Baseline	14.89	12.917
Ridge Regression	5.656	2.5086
Random Forest	5.152	2.496
Gradient Boosting	**4.87**	2.5749
Ensemble	4.9494	**2.49**

Thus, we are getting lowest RMSE of 4.87 with Gradient Boosting Regessor for prediction of duration of the ride. For fare prediction, we are getting lowest RMSE of 2.49 with Ensemble of Random Forest and Gradient Boosting Regressor.

VIII. Conclusion

Ridge Regression, Random Forest, Gradient Boosting and an ensemble method (combination of gradient boosting and random forest) were trained to predict ride duration and fare. The results show that Random Forest and Gradient Boosting performed better than Ridge Regression, which was expected. An interesting insight gained was the most important feature in different models. While Distance turned out to be the most influential feature in Random Forest model, pickup and drop cluster proved to be most important in Ridge Regression. This helps in understanding the underlying difference between these two approaches to model the prediction variable. While in ridge regression, a lot of preprocessing upon the feature was required to get a feature which was linearly related to the prediction variable (combination of pickup and drop cluster ID), random forest did that on its own. Since pickup and drop locations are related to the distance of the ride, random forest learns this relation from distance itself making the other one obsolete.

References

[1] Urban link travel time estimation using large-scale taxi data with partial information. Xianyuan Zhana, Samiul Hasanb, Satish V. Ukkusuria, Camille Kamgac. ScienceDirect 2004.

[2] Visual Exploration of Big Spatio-Temporal Urban Data: A Study of New York City Taxi Trips. Nivan Ferreira ; Jorge Poco ; Huy T. Vo ; Juliana Freire ; Cláudio T. Silva. IEEE Transactions on Visualization and Computer Graphics (Volume: 19, Issue: 12, Dec. 2013)

[3] Fare and Duration Prediction: A Study of New York City Taxi Rides Christophoros Antoniades, Delara Fadavi, Antoine Foba Amon Jr. December 16, 2016

[4] Travel time prediction under heterogeneous traffic conditions using global positioning system data from buses. Vanajakshi, L., S. C. Subramanian, and R. Sivanandan. IET intelligent transport systems 3.1 (2009): 1-9.

[5] Easytracker: automatic transit tracking, mapping, and arrival time prediction using smartphones. Biagioni, James, et al. Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems. ACM, 2011.

[6] Travel-time prediction with support vector regression. Wu, Chun-Hsin, Jan-Ming Ho, and Der-Tsai Lee. IEEE transactions on intelligent transportation systems 5.4 (2004): 276-281.

[7] Source of dataset: https://www.kaggle.com/nyctaxi/yellow-taxis