ISSN No:-2456-2165

A Knowledge Based Travel Time Prediction using Regression Technique

Pranjal Sharma
Dept of Information Technology
Galgotia's College of Engineering and Technology
Uttar Pradesh, India

Prashant Singh
Dept of Information Technology
Galgotia's College of Engineering and Technology
Uttar Pradesh, India

Plash Upreti
Dept of Information Technology
Galgotia's College of Engineering and Technology
Uttar Pradesh, India

Abstract:- Our Research paper proposes a model of travel time based prediction using regression techniques. The objective of our model is to predict the accurate trip duration of a taxi from one of the pickup location to another dropoff location. In today's fast-paced world, where everyone is short of time and is always in a hurry, everyone wants to know the exact duration to reach his/her destination to carry ahead of their plans. So, for their serenity, we already have million dollar startups such as Uber and Ola where we can track our trip duration. As a result of this, we proposed a technique in which every cab service provider can give exact trip duration to their customers taking into consideration the factors such as traffic, time and day of pickup. So, in our methodology, we propose a method to make predictions of trip duration, in which we have used several algorithms, tune the corresponding parameters of the algorithm by analyzing each parameter against RMSE and predict the trip duration. To make our prediction we used RandomForest Regressor, LinearSVR and LinearRegression. We improved the accuracy by tuning hyperparameters and RandomForest gave the best accuracy.

We also analyzed several data mining techniques to handle missing data, remove redundancy and resolve data conflicts. We used the NYC Limousine OpenData, and the travel details of the month of January in the year 2015 to carry ahead with feature extraction and prediction.

I. INTRODUCTION

Travel time prediction plays an important role in transportation. It provides road users with travel time information to understand current traffic conditions and accurate travel time estimation could help to reduce transportation costs by avoiding congested links. Travel time prediction is essential to the development of advanced traveler information systems. Travel-time prediction refers to predicting future travel-time. In this busy world, the goal of every person is to get the exact duration of the distance the person wants to travel, taking into consideration the factors such as traffic, time etc. The different algorithms used are: Linear Regression, Random Forest Regression.

A. LINEAR REGRESSION

It is a linear model that establishes the relationship between a dependent variable *y(Target)*, and one or more independent variables denoted *X(Inputs)*.

Linear regression has been studied at great length, and there is a lot of literature on how your data must be structured to make best use of the model.

$$y = k + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$$

B. RANDOM FOREST

The random forest approach is a bagging method where deep trees, fitted on bootstrap samples, are combined to produce an output with lower variance. However, random forests also use another trick to make the multiple fitted trees a bit less correlated with each other: when growing each tree, instead of only sampling over the observations in the dataset to generate a bootstrap sample, we also sample over features and keep only a random subset of them to build the tree. Sampling over features has indeed the effect that all trees do not look at the exact same information to make their decisions and, so, it reduces the correlation between the different returned outputs. Thus, Random forest algorithm combines the concepts of bagging and random feature subspace selection to create more robust models.

II. LITERATURE SURVEY

In this section, we will briefly review the previous research carried out on travel time prediction using several classic prediction models. Especially papers by S. Kato et al [2] and R. Bera et al [1] which form a good starting point for our research. The goal of this research is to try to develop and test a potential research methodology to promote the efficiency and accuracy of travel-time prediction for its further applications and development, which would be more efficient than the previously proposed methods. In these previous algorithms ([3,4,7]), inputs for linear regression, live data of traveling times and current states of toll checks were used. However, the authors for these papers did not

consider other important factors, such as which day of the week it was or which season of the year they were experiencing in the region. By considering additional important factors and using more efficient Regression methods, we expect to improve the accuracy of predictions made.

The authors of paper [1] proposed to utilise the incoming data from live GPS systems and data streams but did not consider the application of their proposed solution in legacy devices, which are still unequipped with modern GPS systems and have restricted internet access.

Also, our method can run predictions based on a much larger and broader dataset, since the data cleaning and gathering methods that are utilised in our proposed method are efficient enough to overcome the challenges with data that were faced by the previous proposals. Unlike in paper [5], where it was observed that the proposed solution could not process large amounts of raw data.

The ulterior motive of this research is to evaluate the performance of machine learning models, namingly, Linear Regressor and Random Forest Regressor on the trip data extracted from NYC Taxi Limousine OpenData for the year 2015 in the month of January which is also the experimental data for this thesis and estimate the difference in accuracy between both of the methods.

Each algorithm performs differently depending on the dataset and the parameter selection. For overall methodology, Decision tree technique, Linear regression, Lasso regression and Rigid regression have given impressive results[7]. But the Random Forest technique has turned out to be the most suited technique for travel time prediction using our proposed methodology.

III. REQUIREMENT ANALYSIS

Dataset

We used NYC Taxi Limousine OpenData for the year 2015 in the month of January. The link for sample dataset is https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page

Software and Hardware Requirements

Python based Computer Vision and Deep Learning libraries will be exploited for the development and experimentation of the project. Tools such as Anaconda Python, Jupyter Notebook and libraries such as Numpy, Matplotlib, Scipy, Pandas, Sklearn, Pickle, Svm will be utilized for this process. Training will be conducted on NVIDIA GPUs for training the Deep Q-learning technique.

IV. MATERIALS AND METHODOLOGY

Materials that we have used include: Python software for coding and NYC Taxi Limousine OpenData. Our methodology involves the use of machine learning techniques such as: Linear Regression, Random Forest Regression.

A. Dataset

We used NYC Taxi Limousine OpenData for the year 2015 in the month of January. We selected the following features: Trip Distance: Distance is an important factor for predicting the duration of a trip, as Distance = Speed/Time. Day of the week: Weekdays experience slow speed because of the daily routine of schools and offices, henceforth the need for this feature. Time of the day: Peak hours of offices and school start and end such as Morning 8 - 12 and evening 4 to 7 experience high traffic. Pick up and dropoff cluster: Route being travelled that is from one cluster to another is important to predict and identify that particular trip.

The different features are as shown:

TABLE I. FEATURES USED

1	tpep_pickup_datetime	9	dropoff_hrs
2	tpep_dropoff_datetime	10	day_week
3	pickup_longitude	11	tpep_pickup_timestamp
4	pickup_latitude	12	tpep_dropoff_timestamp
5	dropoff_longitude	13	duration
6	dropoff_latitude	14	speed
7	trip_distance		
8	pickup_hrs		

B. Methodology (Proposed Method)

We analyzed several data mining techniques to handle missing data, remove redundancy and resolve data conflicts. We used the NYC Limousine OpenData. We used the travel details of the month of January in the year 2015 to carry ahead with feature extraction and prediction.

The Data Mining techniques are used to handle missing data. In order to handle missing data,we did use a method of binning, but after analysing we found that the missing data was very less in number (i.e) only a few tuples 1-2 percent of total tuples were missing so we decided to remove missing data. So, In order to get rid of redundant data, we perform correlation analysis with the help of plots to check if the attributes are positively or negatively correlated if not redundant. To resolve data conflicts which we encountered for attributes such as time of pickup and dropoff, we converted the time to epoch format and worked on this epoch format to get our features. We used KMeans algorithm to cluster pickup and dropoff location as KMmeans tries to group based solely on euclidean distance between objects we will get back clusters of locations that are close to each other, also as locations are not spread across the world and confined NYC KMeans does a decent job here. To train the model we used RandomForestRegressor because let us consider an example, Let's suppose you are trying to predict income. The predictor variables that are available are education, age, and city. Now in a linear regression model, you have an equation with these three attributes. Fine. You'd expect higher degrees of education, higher "age" and larger cities to be associated with higher income. But what about a PhD who is 40 years old and living in Scranton Pennsylvania? Is he likely to earn more than a BS holder who is 35 and living

in Upper West SIde NYC? Maybe not. Maybe education totally loses its predictive power in a city like Scranton? Maybe age is a very ineffective, weak variable in a city like NYC? This is where decision trees are handy. The tree can be split by city and you get to use a different set of variables for each city. Maybe Age will be a strong second-level split variable in Scranton, but it might not feature at all in the NYC branch of the tree. Education may be a stronger variable in NYC. Applying a similar analogy to our data set where we need to split based on day of the week and pickup , RandomForest comes handy.

V. ANALYSIS

In order to undergo the analysis part we examined several algorithms on regression and concluded that Random Forest is the well-suited Regression technique for our respective proposed model.

To begin with, we first examined the missing data, which was much less compared to the whole data so we decided to exclude the missing data.

By Exploratory Data Analysis we observed that the average speed is more from 0:00hrs to 5:00hrs and average speed is less dusing 16:00hrs to 20:00hrs in the evening.

We did correlational analysis to check for relation between two attributes which helped us to find the redundant data.

In our research, we performed the analysis with the help of KMeans algorithm, in order to cluster the pickup and dropoff location, as KMeans is based on euclidean distance between the objects which in result, we get back clusters of locations that are close to each other, possible only because the dataset was of NYC city, which depicts that KMeans was doing a great job.

We trained our model using Linear Regressor which gave us an accuracy of 75-78% and then we compared our model with a Random Forest Regressor and found that Random Forest was giving us more accurate results of 81-83%.

In comparison to older techniques like Linear Regression our model gave a more accurate result by 6-7%.

Further, to improve the confidence we tuned the hyperparameters such as number of trees and maximum depth for the Random Forest Algorithm.

With an increasing number of trees we observed that the Root Mean Square Error(RMSE) decreased rapidly to a healthy level.

VI. RESULTS

The proposed hierarchy of the workflow model was Loading the data, Cleaning the data, Training the model, Making Predictions, Tuning the hyper Parameters to increase Confidence.

A. Cleaning the data

Cleaning the data involves eliminating the outliers and taking attributes required for feature extraction post Exploratory Data Analysis(EDA). To remove outliers some of the issues occured are to make sure duration is greater than zero, ensure speed needs to be realistic (i.e) speed needs to be between 6 and 140 mph, to make sure pickup and drop off locations are not random and belong to clusters close-by without loss of generality.

B. Exploratory Data Analysis(EDA)

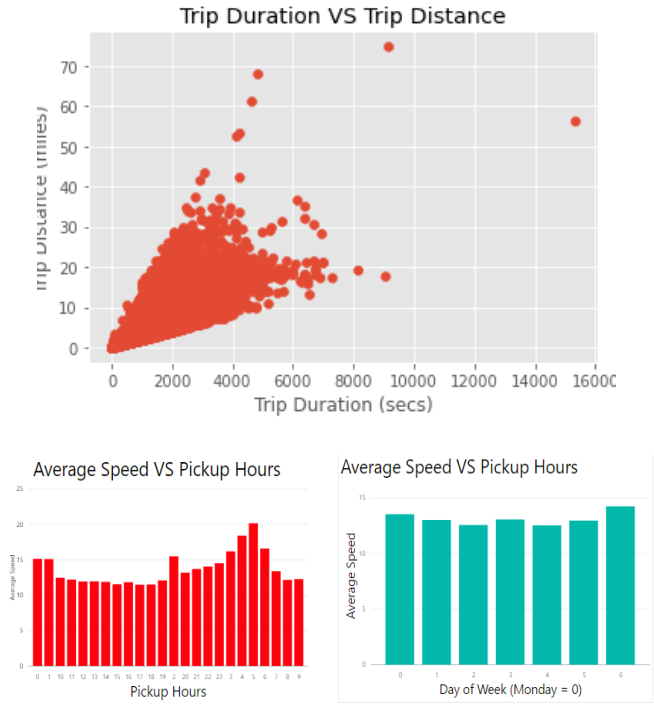

Fig 2. Exploratory Data Analysis(EDA)

EDA as shown in Figure 2 made us draw the following conclusions: The average speed is more during the time 00hrs - 05hrs in the morning, The average speed is less during the time 16hrs - 20hrs in the evening, There exists a positive covariance/correlation between trip distance and trip duration.

ISSN No:-2456-2165

C. KMeans Clustering

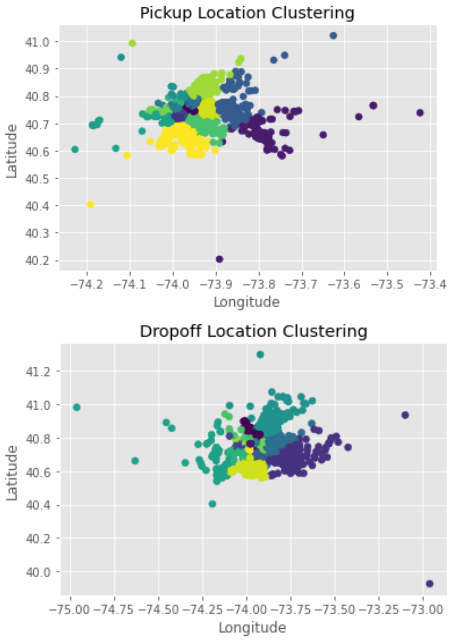

Fig 3. .Pickup & Dropoff clusters using KMeans

Getting features involved clustering the pick up and drop off locations as shown in Figure 3 and having the one hot encoding of pickup and dropoff clusters as features along with one hot encoding of other features such as time and day of pickup wherein these were selected as features after EDA done above.

D. Training the model

To train the model we used Linear Regression and Random Forest Regression algorithm with 80-20 split of dataset for training and testing respectively. It gave an accuracy of 76-78 and 82-83 percent respectively.To improve the accuracy, tuning of several hyper-parameters such as number of trees and maximum depth for a random forest algorithm.

E. Tuning the hyperparameters to improve confidence

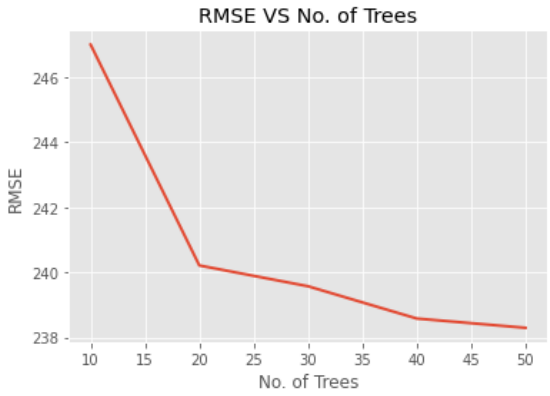

Fig 4. .Tuning of hyper-parameters

With respect to Figure 4 increasing the number of trees as observed above the RMSE decreased to good levels until it reached the elbow point of value greater than 20.

VII. CONCLUSION AND FUTURE SCOPE

Compared to all the other algorithms such as Linear regression(accuracy: 78 percent) and its variants, Random Forest(accuracy: 83 percent) gives the best result. However, a more realistic approach to solve the problem statement would be to get dynamic data or real data via the Cab service provider's API. This would help us get the traffic at that time and will provide accurate results. We aim to carry this work ahead using dynamic data sets via API's, getting real data and using other algorithms such as Stochastic gradient descent to train the model and make predictions.

REFERENCES

[1]. J.K. Jammula, R. Bera, K. V. R. Ravishankar, "Travel Time Prediction Modelling in mixed traffic conditions" International Journal for Traffic and Transport Engineering, 2018 8(1): 135 - 147
[2]. T. IdŽ and S. Kato "Travel-Time prediction using Gaussian process regression: a trajectory-based approach". SIAM Intl. Conf. Data Mining 2009.
[3]. J.M. Kwon and K. Petty, "A Travel Time Prediction Algorithm Scalable to Freeway Networks With Many Nodes with Arbitrary Travel Routes," Transportation Research Record 2005.
[4]. Vanajakshi, L.; Subramanian. S.C.; Sivanandan. R. "Travel time prediction under heterogeneous traffic conditions using global positioning system data from buses", IET Intelligent Transport Systems 2008 3(1): 1-9
[5]. J. Rice and E. van Zwet, "A simple and effective method for predicting travel times on freeways," in Proc. IEEE 4th Int. Conf. Transportation Systems, 2001, pp. 227–232
[6]. S. Arhin, E. Noel, M.F. Anderson, L. Williams, A. Ribisso, R. Stinson, "Optimization of transit total bus stop time models", Journal of traffic and transportation engineering,2016. (English edition) 3(2): 146-153
[7]. S. Ding, H. Zhao, Y. Zhang, X. Xu, and R. Nie, "Extreme learning machine: algorithm, theory and applications,"Artificial Intelligence Review, 2013