# Predicting Taxi Demand Using Machine Learning

**Suhas A Bhyratae[1], Arvind R[2], Bhuvan Kumar S [3] , Aishwarya R Pillai[4]**

*[1]Assistant Professor, Dept of ISE, Atria Institute of Technology,Banglore, Karnataka [2,3,4]Students, Dept of ISE, Atria Institute of Technology,Banglore, Karnataka*

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract -** *Taxi service is imbalanced in big cities. Taxi drivers have to decide where to wait for passengers to pick up someone as soon as possible. Passengers always prefer a quick taxi service whenever needed. The busy area to be concentrated can be decided by the control centre. The sensors that are installed in these vehicles help in automatically discovering new facts. This data is already being used by transporting systems to find time-saving routes, taxi dispatching and other such aspects. By organizing the availability of the taxi, more customers can be served in a short time. In this paper we are using six different algorithms along with the streaming data to increase the performance of demand prediction and distribution of taxi-passenger in a short term time horizon. We evaluate our method on the dataset of New York City. We do this by dividing the city into smaller areas and then analyzing and predicting the demand in each area. The data set includes around nineteen different features with properties like the GPS location, pickup points, drop-off location, etc. This model can be used to predict the demand in the different areas of the city at a particular time and we show which the algorithm that gives the best results*

***Key Words***: **Taxi Demand Prediction, Baseline Models, Regression Models, Time Series Data**

## 1.INTRODUCTION

Taxi drivers need to choose someplace to wait for the passengers so that they can pick someone fast. Likewise, passengers also need to find their cabs quickly. Dispatching the taxi resourcefully helps both the customers and drivers and also helps to reduce waiting time for customers, as well as drivers. In this system, a real-time taxi demand prediction is proposed and streaming data is used to predict the future demand for taxis in a particular area at a particular time. The few real-time objectives include managing many numbers of taxis in a crowded area, utilization of resources effectively to lessen waiting time, organizing the available taxi to serve more customers in a short time. Our system uses features like GPS location and other properties of the taxi like pickup point, drop point etc. to predict taxi demand.

Our work focuses on the real-time choice problem about going to the best taxi stand after a passenger drop-off (i.e. where a quicker pickup of a passenger can be got). The network reliability for both companies and clients can be improved with a smart approach regarding this issue: a clever allotment of vehicles throughout stands will reduce the average waiting time to pick-up a passenger whereas the distance travelled will be profitable. Passengers will also experience a lower waiting time to get a taxi which will be automatically dispatched or directly picked-up at a stand.

## 2. PROPOSED METHODOLOGY

Prediction of taxi demand is a time series analysis problem. The different steps involved are; cleaning the data, clustering, Fourier Transform and making predictions using machine learning models. In the system a minimum Pentium 2.266 MHz processor and Python language is used. 1GB RAM and 250mb disk space is required. A collection of libraries such as dask, folium, numpy, pandas, matplotlib, etc are also used. The input data has been collected from New York City Taxi and Limousine Commission's website. The collected data was around 7000 examples which had to be cleaned and it was brought up to 92% accuracy. The dataset is cleaned in the preprocessing. Redundant data was removed depending on factors like if the pickup point was outside the city, if the trip lasted for more than 24hrs and also removing records which are incomplete. Once the cleaned data set is available it is then clustered using the K-means algorithm. All the time series data will be then converted into frequency domain to get frequency and amplitude using Fourier transforms. This is further on given as input to various baseline models and regression models for which output will be the accuracy. The model with best accuracy will be selected for prediction. The different baseline models used in this system are Simple Moving Average, Weighted Moving Average and Exponential Moving Average and the different regression models used include Linear Regression, Random Forest and xg Boost.

The data points analyzed by creating a series of averages of various subsets of the complete data set is the moving average. It can also be called as moving mean or rolling mean and it is a type of finite impulse response filter. The different types are: simple, weighted and exponential. A simple moving average (SMA) can be defined as an arithmetic moving average which is calculated by taking the sum of all the recent values and then dividing that by the number of values. Short-term averages react quickly to changes in the values of the underlying, while long-term averages are slower to react. The Formula for SMA is: SMA={A_1 + A_2 + ... + A_n}/ n

An exponentially weighted moving average (EWMA) which is also known as exponential moving average (EMA), is a first-order infinite impulse response filter that applies weighting factors which decrease exponentially. There is an exponential decrement in weighting for each older datum, never reaching zero. The graph at right shows an example of the weight decrease. Fig. 1 shows the simple and exponential moving average indicators.

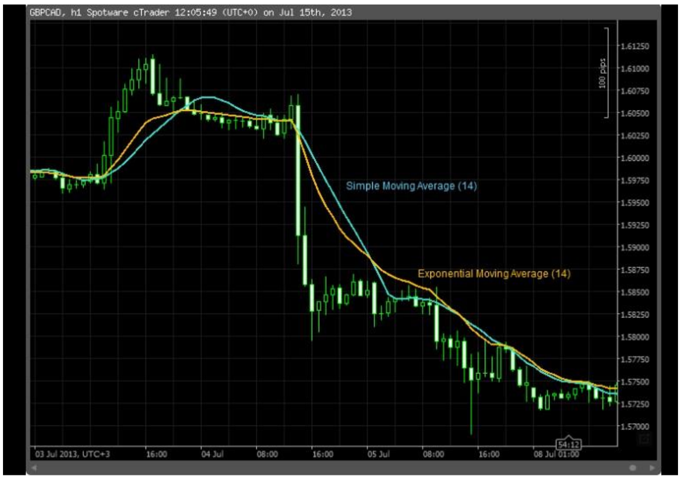

**Fig -1**: Simple and Exponential Moving Average

A weighted moving average (WMA) is an average that has multiplying factors to give different weights to data at different positions in the sample window. Mathematically, the weighted moving average is the convolution of the datum points with a fixed weighting function.The Fig. 2 shows how the weights decrease, from highest weight for the most recent datum points, down to zero. In the exponential moving average which follows, it can be compared to the weights.

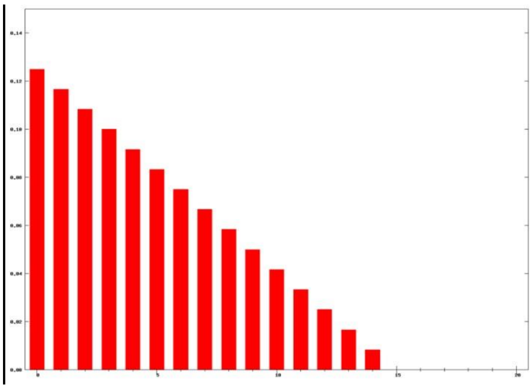

**Fig -2:** Weighted Moving Average

Linear regression is a type of Regression model. It is a linear approach to modelling the relationship between a response that is scalar and one or more independent variables. The case where one independent variable is present is called a simple linear regression.

For cases with more than one independent variable is called multiple linear regression. Here, multiple correlated dependent variables are predicted, rather than a single scalar variable. In linear regression, the linear predictor functions are used to model the relationships whose unknown parameters are estimated from the data. These models are called linear models. Given the value of the predictors, linear regression focuses on the conditional probability distribution of the response, instead of the joint probability distribution of all of these variables. The linear regression model has an extensive use because these models depend linearly on their unknown parameters and are easier to fit than the models which are non-linearly related to their parameters. Fig. 3 shows how the values are divided in a simple linear regression.

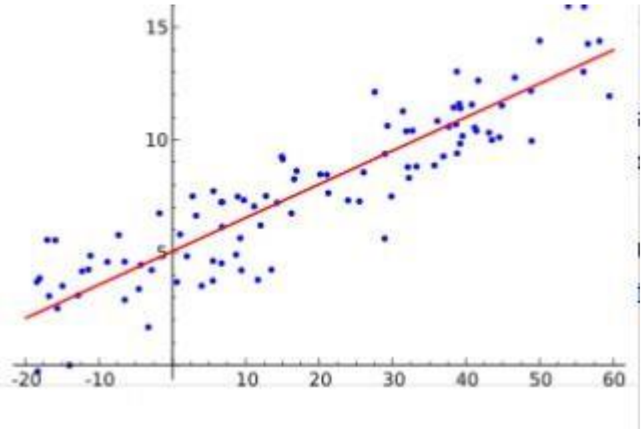

**Fig -3**: Linear Regression

Random forest, also known as random decision forests is a method for regression, classification and also tasks which work by constructing a collection of decision trees during the training time and giving the mode of the classes (classification) or a mean prediction (regression) of the individual trees as output. Fig. 4 gives the structure of random forest.

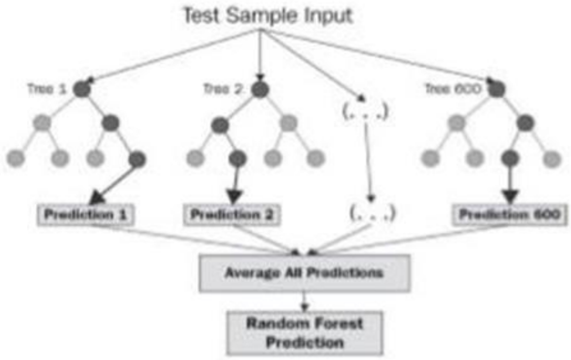

**Fig -4:** Random Forest Structure

XGBoost is a tool that is highly flexible and versatile which can work through most of the regression, classification and other ranking problems. It can be easily accessed and used through different platforms. XGBoost stands for eXtreme Gradient Boosting. This algorithm was developed to reduce the processing time of a computer and to allocate the usage of memory resources. Handling the missing values, support parallelization in the construction of a tree, etc are some of the important features. The fig. 5 shows an example structure of XGBoost.

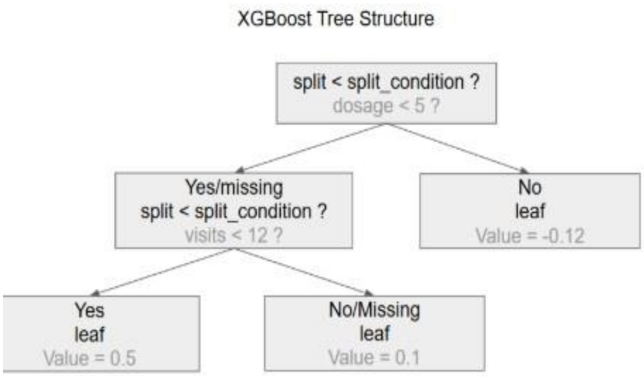

**Fig -5:** Example tree structure for XGBoost

Our system produces predictions and the algorithm which gives the best accuracy will be selected through this process. We see that the Random Forest, Regression Tree and the XGBoost are most suitable

## 3. GENERAL STRUCTURE

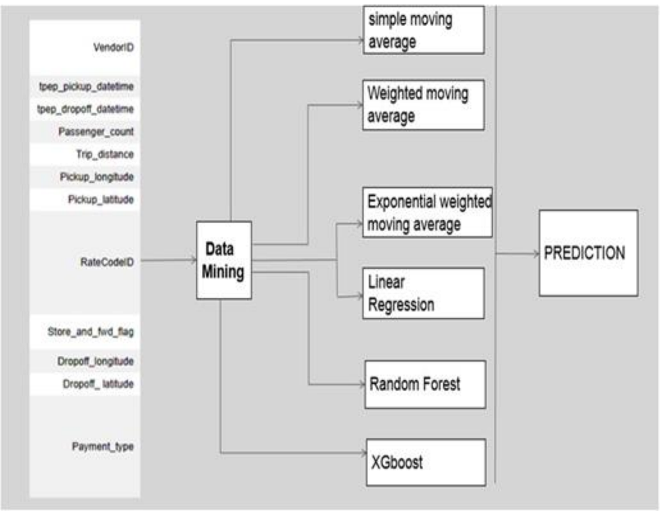

**Fig -6:** General Structure of the System

The above figure shows the general structure of the model. It shows how the data first undergoes data mining where only the features required are extracted. After data mining we perform clustering using k-means algorithm. Then later the data is passed as input to the different models and the predictions are obtained as output. The model which gives the most accurate prediction is obtained.

## 4. CONCLUSIONS

Random Forest Regression seems to be best model where MAPE of train value decrease below 12% there is not any sign of overfitting or under fitting but other models seems little bit overfitting. All models have test MAPE in range of 12.6 to 13.6%.

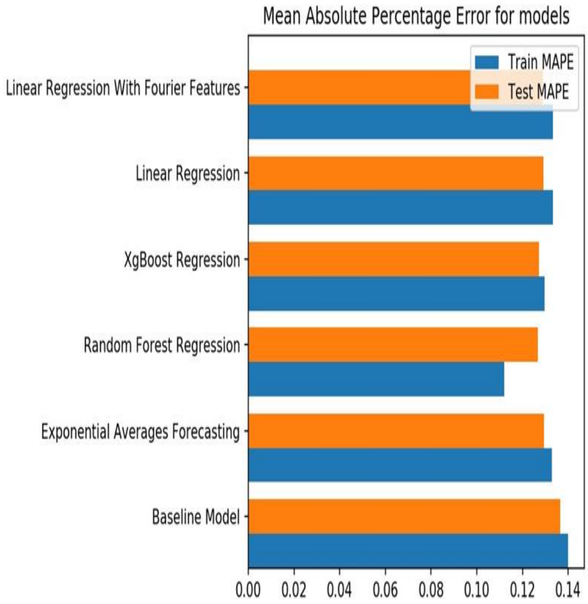

**Fig -7:** MAPE of the models

Our approach towards predicting the taxi demand at a particular area at a particular time interval provides a simple and an efficient method for taxi service companies to improvise their business model based on the demand for taxi and the availability of customers.

## REFERENCES

[1] Filipe Rodriguesa, Ioulia Markoua, Francisco C. Pereiraa (2018) "Combining time-series and textual data for taxi demand prediction in event areas: a deep learning approach" Technical University of Denmark (DTU), Bygning 116B, Lyngby, Denmark.

[2] Kai Zhao, Denis Khryashchev, Juliana Freire, Cl´audio Silva, and Huy Vo (2016) "Predicting Taxi Demand at High Spatial Resolution: Approaching the Limit of Predictability" Center for Urban Science and Progress, New York University.

[3] Ioulia Markou, Filipe Rodrigues, and Francisco C. Pereira (2018) "Real-Time Taxi Demand Prediction using data from the web

[4] Stephan Krygsmana, Martin Dijsta, Theo Arentze (2004) "Multimodal public transport: an analysis of travel time elements and the interconnectivity ratio" Urban and Regional Research Centre Utrecht (URU), Utrecht University, The Netherlands.

[5] Jun Xu, Rouhollah Rahmatizadeh, Ladislau B¨ol¨oni and Damla Turgut "Real-time Prediction of Taxi Demand UsingRecurrent Neural Networks".

[6] Predicting Taxi-Passenger Demand using Streaming Data. Luis Moreira-Matias, Joaao Gama, Michel Ferreira, Joaao Mendes-Moreira, Luis Damas.