IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

TAXI FARE PREDICTION

Aditya Natrajan
Student, Dept. of ISE
B.M.S. College of Engineering Bangalore, India

Apoorva SP
Student, Dept. of ISE
B.M.S. College of Engineering Bangalore, India

Arpit Sharma
Student, Dept. of ISE
B.M.S. College of Engineering Bangalore, India

Darshan ND
Student, Dept. of ISE
B.M.S. College of Engineering Bangalore, India

Darshan S
Student, Dept. of ISE
B.M.S. College of Engineering Bangalore, India

Dr. M Dakshayini
Professor, Dept. of ISE
B.M.S. College of Engineering Bangalore, India

***Abstract*—** Estimating taxi fares has been an important research field within the transport industry since it homeworks pricing trends and transparency of the system both for service providers and customers. The project aims to find machine learning models that predict taxi trip fares using numerous variables, such as distance travelled during trips, time of the day, traffic conditions, number of passengers, and weather. Fare estimate will be addressed with robust data preprocessing, optimal feature engineering, and advanced model training using a synthetic dataset "constructed for practical regression tasks.

The dataset is a rich source of over a 1000 data points with value-key trips varying the trips duration and fare amount, as well as contextual parameters like traffic and weather. This dataset will provide you with real world problems like missing values, outliers, correlation of features all together in one bundle. Applying and comparing ML models like Random Forest and Logistic regression and decision tree on this dataset based on Realization above had proven that Random forest Model proved the best with lower values in MSE, and was capable of fitting even non-linear relationship between the features. Along with its machine-learning train and evaluation functionality, it also provides a lightweight mechanism for predicting fares from input at application runtime.

It also aims to collect the data on traffic, updates from weather and real-life datasets which could to be placed into next architecture to enhance the data feeding and to improve the model adaptability in future attempts. The predictive analytics embodied in this work speaks to power as it pertains to the taxi domain in such a way where it burns stronger in the empirical sense given the scope of paradigm machine in the usage to enhance fare predictions and decision making in transportation's dynamic environment.

I. INTRODUCTION

It has revolutionized urban transportation, transforming into a very convenient and flexible mode, mostly seen as a means of transportation or commuting. Digitalization has enabled millions of travel services to be purchased through applications; with accurate pricing, loyalty, efficiency and customer satisfaction have increased. Estimating travel prices is an essential element for operational and competitive advantage, and also affects customer trust, driver satisfaction and, as a result, the competitiveness of the platform. BAI offers its users a more convenient, flexible and often cheaper alternative to taxi services. Digital platforms serve millions of passengers worldwide every day, based on accurate fare estimates to ensure fairness, efficiency and user satisfaction. Accurately estimating the cost of a trip is an important aspect of the operation, as it is the best way for the platform to win customers, drivers and win the competition. , travel time, traffic and weather updates and travel time decisions of all passengers. Effectively modeling all these variables requires robust learning algorithms that can capture the interrelationships between these features and withstand uncertainties such as acceleration or sudden drop in the air. A prediction method for taxi rides using three machine learning models: random forest, decision tree, and linear regression. We will evaluate each model based on its performance in cost estimation, thus providing a comparative analysis of the advantages and disadvantages of these models. Besides, the application creates a web interface using Flask and HTML so that the user can get an idea, it shows the travel, time, transportation, and culture of Africa, it shows the estimated cost, and all three models are online.

It handles the difference between categorical and numerical variables very well with engineering design to handle missing values, so that the model becomes as robust and comprehensive as possible. To bridge the gap between theory and practice, this project applies machine learning concepts to real-world learning machines through predictive models supported by web applications. Introduce a machine learning model to estimate the initial cost.

The ultimate user-friendly platform for dynamic and comparative price predictions.

View and solve real-world problems such as traffic and weather.

Check the performance of different machine learning systems on real-world users. This result and application will form the basis for future implementation of real-time traffic cost predictions that include real-time traffic and weather data.

II. LITERATURE SURVEY

Paper 1: Predictive Analysis of Taxi Fare Using Machine Learning

Objective and Scope

The research will address the issue of taxi fare prediction using the machine learning techniques. The purpose is to identify a model, which considers many factors (distance, time, location, no. of passengers, etc.), for accurate estimation of the taxi fare. It shows how this process of the calculation of fares of taxis has become more complicated with the progressing technology and factors.

Dataset and Preprocessing

● Source of Dataset: Kaggle

○ It has some core features: the amount of the fare, the pickup/drop-off coordinates (latitude and longitude), pickup date and time, and the passenger count.

○ This dataset has approximately 5 million rows. This research work, however, only took 80,000 records covering 2009 to 2016.

● **Preprocessing**

Handling Missing Data: It used median in order not to have biased effects with mode and mean.

Removing Noisy Data:

Outlier pickup/drop-off coordinates have been removed.

Rows containing all zeros in key fields have been deleted.

Methods

The paper's machine learning pipeline is described in five general steps:

1. Data Visualization:

Features were visualized with scatter plots, bar graphs, and histograms to understand data distributions and trends.

Example insights:

■ Most of the trips were within some range of longitude (-73 to 40) and latitude.

■ Travelers and amount: counts distribution.

2. Feature Engineering

○ A Correlation Matrix which analyzes the significance value of variable input.

○ All those independent variables. For instance the number of passengers and latitudes

Were contributing toward establishing the prediction required for fare on which fare reliant.

3. Modeling:

○ Supervised learning: It learns, trains based upon the availability of labeled data independent and dependent also.

Tworegredient models have to be run here

■ **RandomForest**

It is an ensemble technique based on the tree structure.

Merging of several decision trees for better predictive accuracy and minimization of overfitting.

■ **Linear Regression:**

Builds a linear relationship between the dependent variable and independent variables.

Formulated by the equation y=a+bxy = a + bx y=a+bx, which will be your dependent variable (fare) and xxx the independent variable.

4. Training and Testing:

○ The dataset was divided into 75% training and 25% testing datasets.

○ The model was trained on historical data and tested on unseen data for the predictions.

5. Evaluation Metrics:

○ R^2 (Coefficient of Determination): This measures the goodness of fit of the model in predicting the

variability of the target variable.

○ MSE(Mean Square Error): It is a measure of average squared difference between

the actual and predicted values.

○ RMSE (Root Mean Square Error): It depicts the error of the model in the prediction in the

same unit as the target variable.

This article compares RandomForest vs Linear Regression:

RandomForest

R^2=0.5

MSE=2.163

RMSE=1.470

Saw gives better prediction because it's an ensemble technique and takes care of nonlinearity and interaction features.

Linear Regression

R^2=0.4

MSE=2.642

RMSE=1.625

Failed since complex relations among the variables were unable to be grasped

Conclusion

●Best Model: It was a proof that the model of RandomForest that predicts the fare of a taxi is

a better accuracy and strength as compared to Linear Regression.

● Key Insights:

Random Forest's ensemble method, or in other words, using multiple decision trees improved

accuracy and decreased prediction error.

Linear Regression is less complex but it fails to capture the non-linear relationships.

Future Scope

The authors propose the future scope for accuracy enhancement with the advanced techniques:

Algorithms like Decision tree and Ridge Regression may be tried.Feature engineering and including other variables may increase the prediction performance

Paper 2: Real-Time Prediction of Cab Fare Using Machine Learning

Objective

A machine learning-based real-time taxi fare prediction system will be developed for metropolitan cities. In the model design, different factors such as the pickup and drop-off location, total number of passengers, distance of the trip, and time will be taken into account for improved accuracy of predictions. The main motivation behind the research is the increasing trend in ride-hailing services where fare prediction plays a significant role for both service providers and customers.

Methodology

Framework: CRISP-DM

The study employs the Cross-Industry Standard Process for Data Mining (CRISP-DM), a methodology that provides a structured framework for data-driven projects. The key phases are:

- Business understanding
- Data understanding
- Data preparation
- Modeling
- Evaluation
- Deployment

Data Collection and Preprocessing

- Dataset:

The dataset includes features such as:

- Fare amount (dependent variable)
- Pickup and drop-off latitude and longitude
- Passenger count
- Date and time of the trip

The dataset is taken to represent diverse cab rides across a city.

Preprocessing Steps:

- Handling Missing Values: Median dib was used to impute the missing data of columns passenger count and fare amount which performs better than mean or KNN imputation
- Outlier Removal: It used statistical techniques to identify latitude and longitude that result in extreme values and also unnecessary fares and eliminate them. It "purified" the data to have consistency and to minimize randomness.
- Features Engineering: He had the haversine function compute the distance between

pickup and drop-off destinations. This equation would better express what distance represents as the minimum distance between any two points having their geographic locations. He has pulled the time of the day, the day of week, and month off from the date stamp.

Exploratory Data Analysis

- Plotting: Probability density functions among other visualizations had somehow placed meaning in distributions of variables. This study compared various fare distributions and passenger counts to extract trends and anomalies.
- Correlation Analysis:The correlation matrix shows that distance is the best predictor for the amount of fare. Other factors, such as passenger count, have a weak correlation with the fare amount.

Modeling and Implementation

We compare two supervised learning models in the case study:

- Random Forest: A cooperative algorithm works in conjunction with a group of decision trees that enhance the performance but with the drawback of overfitting. It is highly complex and nonlinear when the number of variables is large. Some other importantly important characteristics are rank ordering in terms of the feature's importance and its noise-robustness.
- Linear Regression:An algorithm that is comparatively simple predicts the fare using a linear relationship between the independent and dependent variables. Assumptions of linear regression (for example, normality of the data, linearity, homoscedasticity) have been checked before the implementation.
- Training and Testing: Data Training and Testing: This data opinion in 80% training set and 20% in test set. Both the models were designed using historical data and then verified using the unseen data to gauge how they respond under prediction.

Evaluation Metrics

- MeanAbsolute Error (MAE): ○ Measures the average magnitude of errors between predicted and actual fares.
- Accuracy: correct
  - Ratio of correct predictions to total predictions, calculated as: Accuracy=Number of predictionsTotal \frac{\text{Number of correct predictions made\text{Accuracy} predictions}}{\text{Total = predictions made}}Accuracy=Total predictions madeNumber of correct predictions
  - Alternatively, accuracy was derived using MAE: Accuracy=1−MAPE (Mean Absolute Percentage Error)\text{Accuracy} = 1-\text{MAPE (Mean Absolute Percentage Error)}Accuracy=1−MAPE (Mean Absolute Percentage Error)

Results

- Random Forest: The prediction accuracy has improved as compared to Linear Regression which was confirmed by greater accuracy with even lower MAE. The random forests in themselves will have an ensemble behavior to capture non-linear relationships and complex interactions among the variables.
- Linear Regression: Gave quite a good model corresponding to prediction, maybe because nonlinearities with respect to the data set and outliers will catch even higher errors.

Conclusion

Machine learning methods have been successfully developed for forecasting cab fares. Because model superiority, a more accurate and robust model, is the case for the Random Forest model against that of the linear regression one. This particular application demonstrates where the difference made by pre-processing (missing value treatment and removal of outliers) for a feature engineering process, such as the need for distance calculation with Haversine distance, becomes a huge deal.

Future Study

- Advanced Algorithm: Try working with Gradient boosting techniques such as those using Decision Trees, Light GBM, perhaps even Neural Nets to get greater accuracy.
- Dynamical Price : It takes care of the real-time parameters, which include traffic, weather, and demand climbs and predicts fares in a dynamic pricing system.
- Real-Time Applications: Implement this model in the real-time operating systems so that it may serve the companies of ride-hailing as well as the passengers in estimating fares.

Paper 3 : Real-Time Prediction of Taxi Demand Using Recurrent Neural Networks

Objective

This study will use real-time city-wide area-by-area taxi demand predictions using LSTM neural networks. This improves the fleet management, decreases passenger waiting time, and maximizes the effective execution of operations, especially in the future self-driving taxi cases.

Key Methodologies

- Problem Framing: This one is probably the most challenging one because the variance of factors such as past demand, time, weather, and drop-off patterns need to be put into consideration so that an effective prediction for taxi demand can be made. Applying LSTM networks that model time-series data with long-term dependencies and patterns so that future demands in said regions can be predicted.
- Dataset: New York City Taxi and Limousine Commission (TLC) data- a period of 3.5 years (Jan 2013 - Jun 2016).-About 600 million taxi trips with pick-up/drop-off times, GPS coordinates, and timestamps are recorded after cleaning. The city was divided into 6,500 areas using Geohash coding. It has a grid size of 153 m x 153 m, which allows more granular demand prediction.

- Methodology : Take the dataset as weekly sequences and then time step-wise to the interval of 20 minutes to produce sequential input data for modeling.
  Developed two models as described below:
  - LSTM-MDN: It suggests a probabilistic prediction of the taxi demand as a probability distribution rather than as a deterministic value. Output parameters of a mixed distribution such as mean and variance and mixing coefficients can be utilized to include uncertainty.
  - LSTM-MDN-Conditional: one can invoke listening cells for which contextual awareness was augmented in one region and conditioning the predictions with demand of another region's taxi.

- Performance Metrics: Symmetric Mean Absolute Percentage Error (sMAPE): Percent error measure alongside actual demand. Root Mean Square Error (RMSE): Values that represent the different demands forecasted to the actual demand. The performance was assessed with these metrics, both over the whole city and at certain locations.

Experimental Setup

The data was split 80% for training and 20% for validation. The experiments are run on the GTX 1080 GPU and the training time ranges from 2–4 hours per run.

The models are compared to two baselines:

- Feed-Forward Neural Networks (FC): A less complex neural network without the memory of times.
- Naive Average Predictor: Makes a prediction for the demand as an average of the past five time-steps.

Results

- City-Wide Performance
  - LSTM-MDN performs better than baselines:
    - **sMAPE:** The median error is around 17%, or ~83% in accuracy.
    - **RMSE:** Has smaller deviations than FC and Naive models, especially at peak demand.It has comparable performance at different time-step lengths, such as 10, 20, and 60 minutes.
- Area-Specific Performance
  - The application of **LSTM-MDN-Conditional** should be better than **LSTM-MDN**, especially in high-demand and regular-demand areas as the demand of the neighboring areas is incorporated as an auxiliary context . The FC and Naive models performed poorly as the model could not capture many patterns, thus more errors
- Effect of Auxiliary Features
  - A slight positive deviation in the predictions was found after including variables like time of day and drop-offs and week day and other similar factors The only feature that was prominent and which could be extracted from pick-up data is 'Historical' and the feature that is coming out here is of lowest importance is 'weather'.

Strength of Proposed Approach

- **Sequence Learning Using LSTMs:** LSTMs considerably outperformed in learning long-term dependencies like predicting return journeys based on events, say concerts or shopping.
- **Mixture Density Networks:** The networks made probabilistic predictions that added robustness with uncertainty
- **Scalability** :The model was able to handle the simultaneous prediction of 6,500 city areas by learning one to infer others.
- **Real-Time Feasibility:**After training the model, it took less than a second to make predictions hence suitable for real-time application.

Limitation

- High computational needs for training, especially for the LSTM-MDN-Conditional model.
- Less influence of external factors such as weather due to coarse-grained data.

Future Work

- Incorporating other sources of data (for example, business location, events) for better demand predictions.
- Real-time reorganization of taxi fleets based on the predictions of the model, especially for autonomous taxi systems.
- Advanced architectures, such as attention mechanisms, to further improve performance.

Conclusion

The study shows that LSTM-based models, such as LSTM-MDN and LSTM-MDN-Conditional, outperform traditional methods significantly in real-time taxi demand prediction. These models show promise for optimizing taxi dispatch systems, improving passenger experience, and fuel consumption reduction with a median prediction accuracy of ~83%. This work can serve as a foundation for future intelligent transportation systems, especially for autonomous vehicles.

III. IMPLEMENTATION

It entailed developing the project as a robust pipeline in preprocessing, training, evaluation, and the deployment of machine learning models to predict trip fares using user inputs and contextual information, such as traffic and weather conditions. The following describe the technical implementation of the system:

6.1. Data Preprocessing

Data preprocessing is very fundamental in ensuring that the model achieves accuracy and good performance. The preprocessing pipeline consisted of the following

- **Handling Missing Data:**Numerical features like Trip_Distance_km and Base_Fare were filled using the mean strategy.Categorical features like Time_of_Day and Traffic_Conditions were filled using the most frequent strategy, which makes sure there are no missing categories.
- **Feature Encoding:**All categorical features, including Time_of_Day, Day_of_Week, Traffic_Conditions, and Weather, were encoded using one-hot encoding to make them numerical suitable for machine learning models.
- **Feature Scaling:**StandardScaler was used to normalize the numerical features so that all variables were contributing equally to the model's performance.
- **Dimensionality Reduction:**PCA was applied to reduce the dimensionality of the dataset to avoid overfitting and computational complexity.
- **Data Splitting:** The dataset was split into 80% training and 20% testing to ensure unbiased model evaluation.

6.2. Model Training

Three machine learning models were trained for the prediction of trip fares:

- Linear regression: A simple baseline model to predict fares based on a linear relationship of features and the target variable.
- Decision Tree Regressor:A non-linear model that splits data hierarchically depending on feature importance.
- Random Forest Regressor:An ensemble model combining multiple decision trees to improve accuracy and robustness.

Each model used the training dataset created after preprocessing.

6.3. Model Evaluation

The performance of the models was measured by using the following metrics:

MSE: Average square difference between the actual value and predicted values.

MAE: Average absolute difference between the actual value and the predicted value.

R² Score: This is the fraction of the dependent variable variance accounted for by the model.

It was tested with performance in these three settings.Data was tested without PCA

Data with PCA in order to get an understanding if dimensionality is reduced.The model has seen to perform better without PCA because all columns contribute equally.

6.4. Fare Prediction Logic

Input Data:

Information sought from the user (for example: source, destination, time of day) and contextual information (for example: traffic, weather conditions) were fetched through a web interface.

Feature Engineering:

Google Maps API gave an idea of trip distance and duration.Synthetic data generators gave random values for traffic and weather.

Prediction:

The trained models predicted the trip fare, and the final fare was calculated as the average of predictions from all models.

6.5. Deployment

The system was deployed using a Flask-based web application:

Frontend:HTML templates provided an interactive interface for users to input trip details and view predictions.

Backend: Flask handled API calls, preprocessing of user inputs, and integration with the trained models.Predictions were returned to the frontend for display.

6.5 Deployment

The system was deployed using a Flask built web application.

Front-end: The HTML templates made the portal interactive for the user to input information regarding the trips and procure predictions.

Back-end: The Flask managed the calls to the API besides preprocessing the inputs provided by the user and model integration. The predictions are sent back to the front-end to be displayed.

6.6. Validation

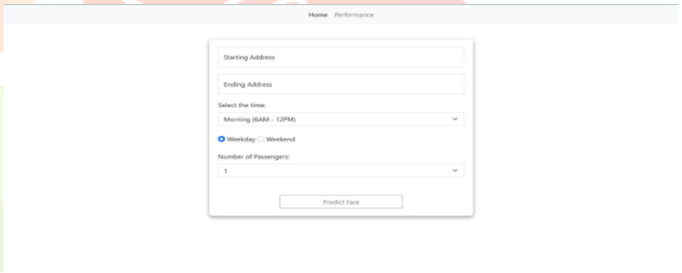
Data integrity is checked with pre-processing pipeline and models stressed on rugged tests.No null was found after the preprocessing of the data.Cross-verifying the model outputs with test data is ensured for correctness and reliability

Implementation: it combined preprocessing techniques, solid machine learning models with an interface towards the creation of an accurate scalable trip fare prediction system.

IV. RESULTS

The above project proves that the pipeline implemented is capable of predicting trip fares from the input features. This section presents pertinent performance metrics, model-to-model comparisons, and key observations. Users have to input the starting address, destination address, time, day, and number of passengers. On inputting this data, the models will predict the fare.

They need to enter the starting and destination address, the time, the day, and the number of passengers. Once this data is input, it will predict the fare by the models.



Starting Address

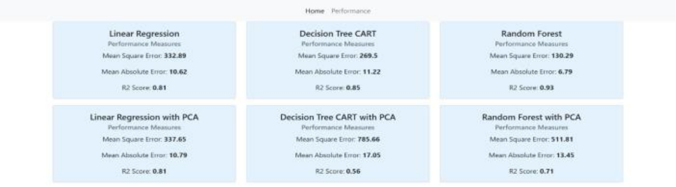
Ending Address

Select the time:
Morning (6AM - 12PM)

● Weekday ○ Weekend

Number of Passengers:
1

Fig :7.1



Linear Regression
Performance Measures
Mean Square Error: **332.89**
Mean Absolute Error: **10.62**
R2 Score: **0.81**

Decision Tree CART
Performance Measures
Mean Square Error: **269.5**
Mean Absolute Error: **11.22**
R2 Score: **0.85**

Random Forest
Performance Measures
Mean Square Error: **130.29**
Mean Absolute Error: **6.79**
R2 Score: **0.93**

Linear Regression with PCA
Performance Measures
Mean Square Error: **337.65**
Mean Absolute Error: **10.79**
R2 Score: **0.81**

Decision Tree CART with PCA
Performance Measures
Mean Square Error: **785.66**
Mean Absolute Error: **17.05**
R2 Score: **0.56**

Random Forest with PCA
Performance Measures
Mean Square Error: **511.81**
Mean Absolute Error: **13.45**
R2 Score: **0.71**

Fig :7.2

V. CONCLUSION

This gives a good demonstration of those machine learning models further showing how the trip-related information could be learned from these models in order to predict taxi fares. It was found that the model with the best accuracy was Random Forest because it could handle multi-dimensional features with non-linear relations. An easy user interface in a web format provides full theoretical application of machine learning system applications with realistic, dynamic, and visually observable fare estimates. More accurate and dynamic systems will soon be created and implemented with transportation systems through future modifications that can use predictive analytics.

Out of the above three models, Linear Regression, Decision Tree, and Random Forest, Random Forest Regressor performed best among all these models based on minimum error metrics and maximum accuracy.The decision tree had average performance with extremely efficient computing since it made the network very prone to overfitting. The linear regression, as compared to the latter, is not very good since it performed rather badly concerning the dataset due to non-linear relations within the dataset.

Effect of PCA: Nearly all the performances decreased when PCA is used implying that original features carry most of the information needed to predict the fare and is lost as soon as the dimensions are reducedAlthough PCA's dimension reduction has a significantly much smaller time for computation, the result per prediction accuracy in that dimension reveals that it is not very well-suited for the problem.

Prediction Accuracy: The models are fair in terms of accuracy without PCA, and this provides a good basis for estimating trip fares.Adding several conditions including that imposed by traffic, weather, and passenger effects enhances the predictive power of the models.These models had a pretty fair accuracy level Page 3, having a proper basis for the prediction of the trip fare. Diversity in features such as the traffic condition, weather, and passenger load at the time consolidated predictability in the models.

The study proves that the choice of an appropriate machine learning model and preprocessing methods depends on the nature of the dataset and the problem. The ability of this model to deliver accuracy and interpretability will place Random Forest Regressor in the position to be the most trusted model used in predicting fares in ridesharing. PCA negative results recommend the avoidance of the dimension reduction process applied with less thought, but applied instead.

REFERENCES

1. Real-Time Prediction of Cab Fare Using Machine Learning
S.Muhammad, M.F.Sohail, and M.M.F.Iqbal, “Real-Time Prediction of Cab Fare Using Machine Learning,” *2022 IEEE International Conference on Engineering, Technology and Applications (IETA)*, 2022.Available:https://ieeexplore.ieee.org/abstract/document/9752315

2. Real-Time Prediction of Taxi Demand Using Recurrent Neural Networks
W.Shi, Y.Ma, H.Xiao, and Y.Tang, “Real-Time Prediction of Taxi Demand Using Recurrent Neural Networks,” *2017 IEEE 37th International Conference on Distributed Computing Systems Workshops(ICDCSW)*,2017.Available:https://ieeexplore.ieee.org/abstract/document/8082792

3. Google Scholar
Accessible for scholarly articles and research papers at https://scholar.google.com

4. Wikipedia
A multilingual open encyclopedia at https://www.wikipedia.org

5. Dataset for Taxi Price Prediction
D.Kuznetz, “Taxi Price Prediction Dataset,” *Kaggle*, 2019. Available: https://www.kaggle.com/datasets/denkuznetz/taxi-price-prediction/data

6. Exploring Machine Learning Research
Larochelle et al., “Exploring Machine Learning

Research," *MLR Press*,2011.Available:https://proceedings.mlr.press/v15/larochelle11a.html

7. Emerging IoT Trends "Emerging IoT Trends," *Academia.edu*, 2020. Available: https://www.academia.edu/download/78765194/CSEIT2062108.pdf

8. Climate Data Analysis "Climate Data Analysis," *MDPI Open Access Journal*, 2013. Available: https://www.mdpi.com/2076-3417/13/18/10192

9. IoT for Smart Cities "IoT for Smart Cities," *Academia.edu*, 2019. Available: https://www.academia.edu/download/73852982/jjraset.2019.pdf

10. Taxi Fare Prediction Analysis Benjamin Hagen, "Taxi Fare Prediction Analysis," 2018. Available: https://www.benjaminhagen.com/docs/TaxiFares.pdf

11. Socioeconomic Studies "Socioeconomic Studies," *Tilburg University Repository*, 2019. Available: http://arno.uvt.nl/show.cgi?fid=175974