Forecasting Ride-Share Driver Demand Using Machine Learning

Presentation by Group 7

Manidedeepya Chennapragada
Kalyan Vikkurthi
Dharnidhar Reddy Banala
Nikhil Mylarusetty

DATA 270

Project Background and Executive Summary

Motivation

The motivation behind the project stems from the need to improve taxi dispatch efficiency and enhance customer service for ride-sharing companies and transportation providers. Accurately estimating driver demand at different locations and times is crucial for optimizing operations and increasing revenue.

Needs

Ride-sharing companies and transportation providers need accurate driver demand forecasts to make informed scheduling decisions. By understanding various factors that affect demand, such as time of day, day of the week, and pickup location, machine learning models can provide valuable insights for effective operations management.

Project Background and Executive Summary

Target Problem

The NYC Yellow Taxi Number of Pickups project seeks to offer valuable insights into the demand for taxi services in the city and how it varies based on different factors such as location, time of day, and day of the week. The data collected can be used to make informed decisions regarding transportation planning, traffic management, and other urban policies.

Project Requirements

FUNCTIONAL REQUIREMENTS

- Python Libraries for Data Pre-processing and EDA
- Data Storage - GCP
- Configure the models with appropriate hyperparameters
- Evaluation metrics

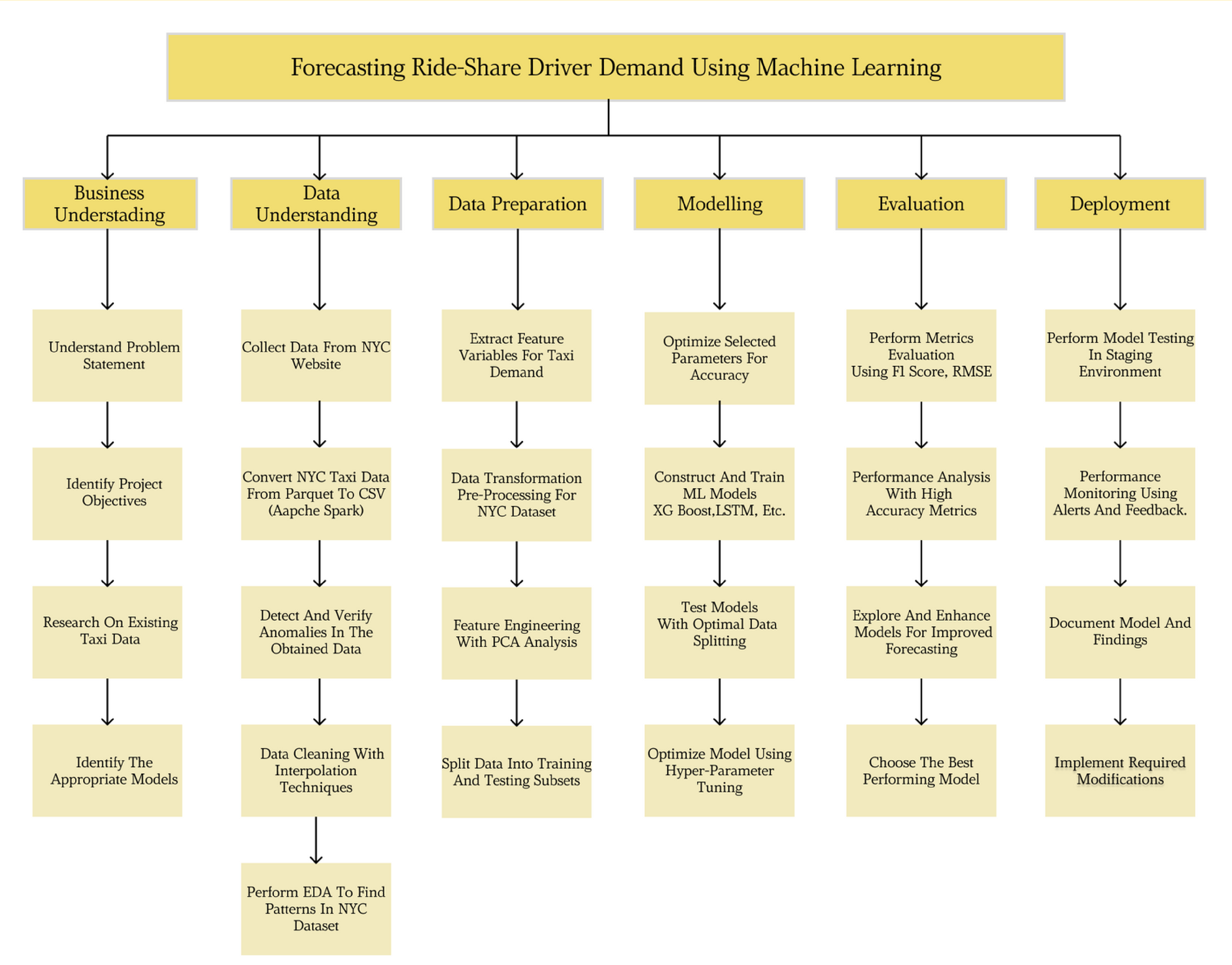
DATA REQUIREMENTS

- NYC TLC Data Source - collected from the official New York City website
- Data Volume
- Data Availability– it should be easily acquired
- Data Quality

AI REQUIREMENTS

- Random Forest
- XG-Boost
- LSTM
- Facebook Prophet

Project Plan

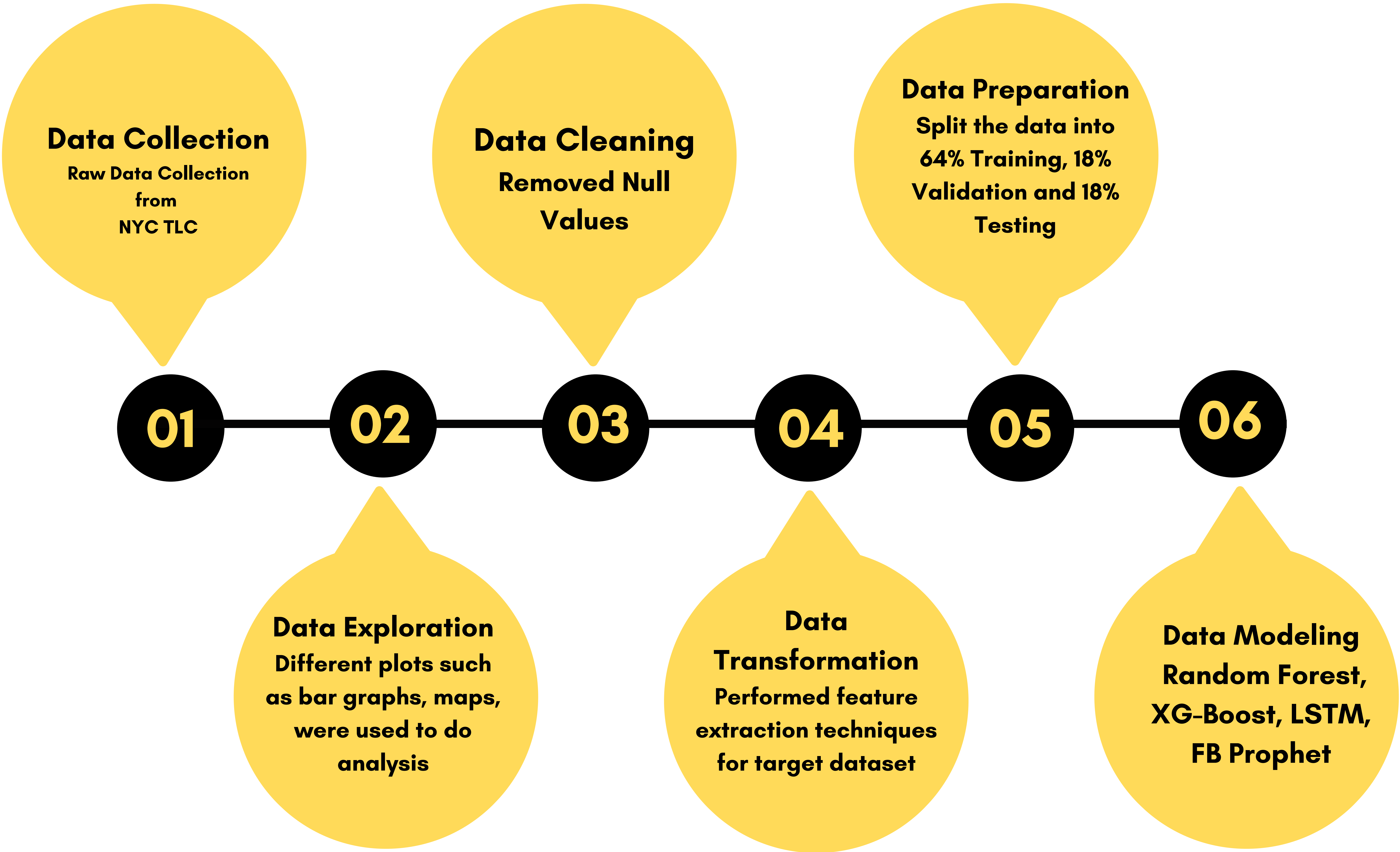


Project Resource, Cost estimation and Plan

Function	Resource Type	Resource	Duration	Cost Estimation
Local Machine	Hardware	64- bit machine or higher	3 Months	$2,000
Data Storage	Hardware	NYC TLC data , GCP bucket	3 Months	Free (Used $100 credit by SJSU.edu organization)
Cloud Service management Tool	Software	GCP Terminal	3 Months	Free (Used $100 credit by SJSU.edu organization)
Model Deployment : Train, validation, Test Datasets	Software	Google Colab	3 Months	Free (Used $100 credit by SJSU.edu organization)
ML Frameworks	Software	Scikit-learn	3 Mothns	Free
Visualization Tool	Software	Matplotlib,Seaborn	3 months	Free

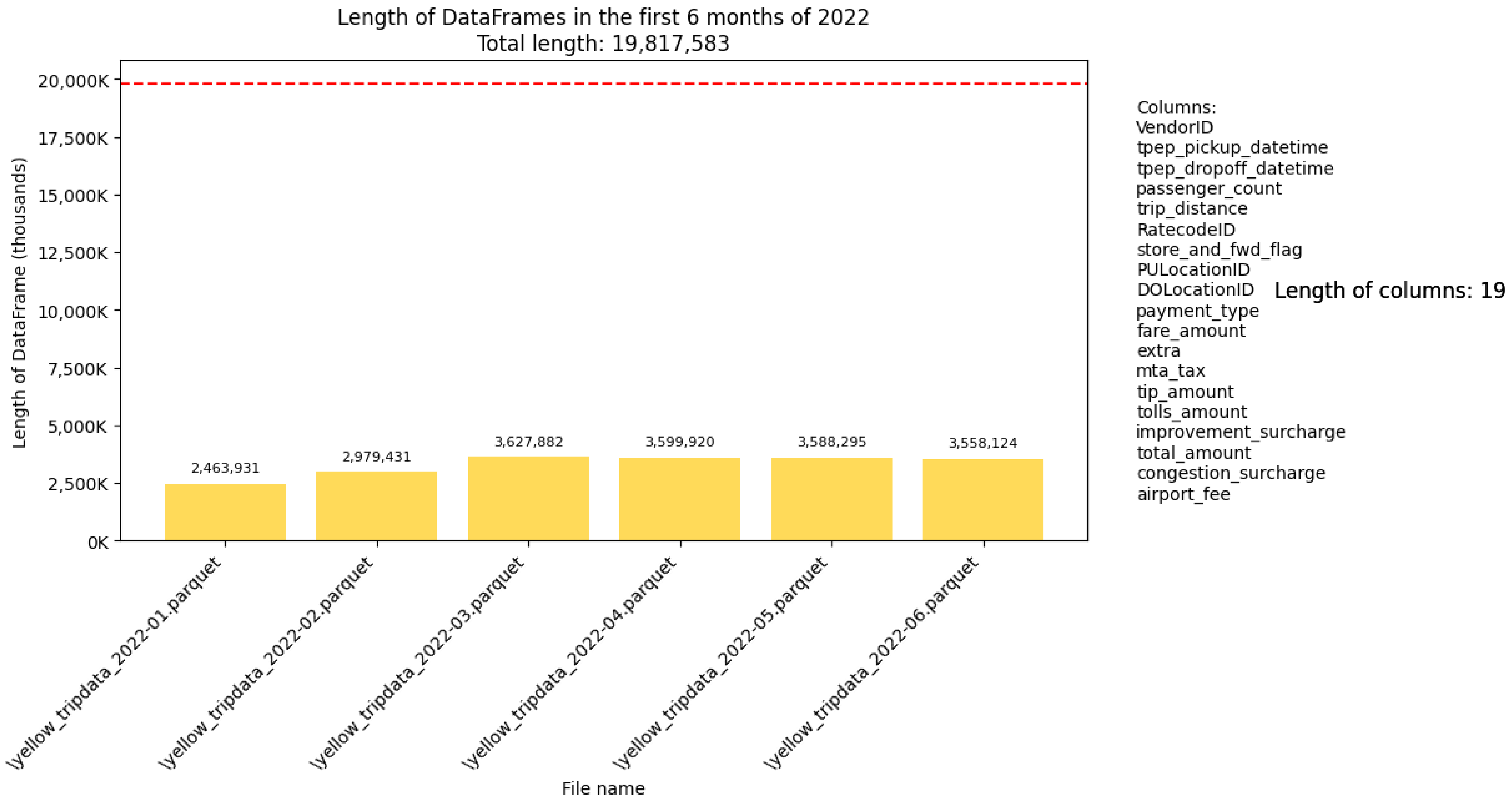
Literature / Technological Survey

Paper Name	Methodology	Technology Findings	Comparision
Silveira-Santos et al. (2021)	Facebook Prophet, Random Forest	Prophet was good at predicting Long term , Random Forest was good at short term prediction	MAPE values for prophet is 8 % (Long term), Random Forest is 6 % (Short term) and Both the algorithms performed better with their respective short term and long term prediction
Kankanamge, K. D. et al. (2021)	XG Boost, SVR, Neural Networks	XG boost is good at distributed computing, avoiding overfitting by using regularization	Mape value is lowest for XGBoost (MAPE : 17) when compare to SVR(MAPE : 20) and Neural Networks (MAPE : 24)
Askari, B. et al. (2020)	LSTM, XG Boost	LSTM algorithm is modified with Deep Sequence Model	The Deep Sequence LSTM Ouperformed XGBoost, RF with MSE and SMAPE as 1.41 and 17.25 respectively.
Wang and Mi (2018)	ARIMA, LSTM	Three different ARIMA models are used by tuning different parameters and these parameters are decided by KNN Algorithm	LSTM outperformed the ARIMA in terms of RMSE metric with value 12


Data Collection
Raw Data Collection
from
NYC TLC
Data Cleaning
Removed Null
Values
Data Preparation
Split the data into
64% Training, 18%
Validation and 18%
Testing
01
02
03
04
05
06
Data Exploration
Different plots such
as bar graphs, maps,
were used to do
analysis
Data
Transformation
Performed feature
extraction techniques
for target dataset
Data Modeling
Random Forest,
XG-Boost, LSTM,
FB Prophet

Data Collection

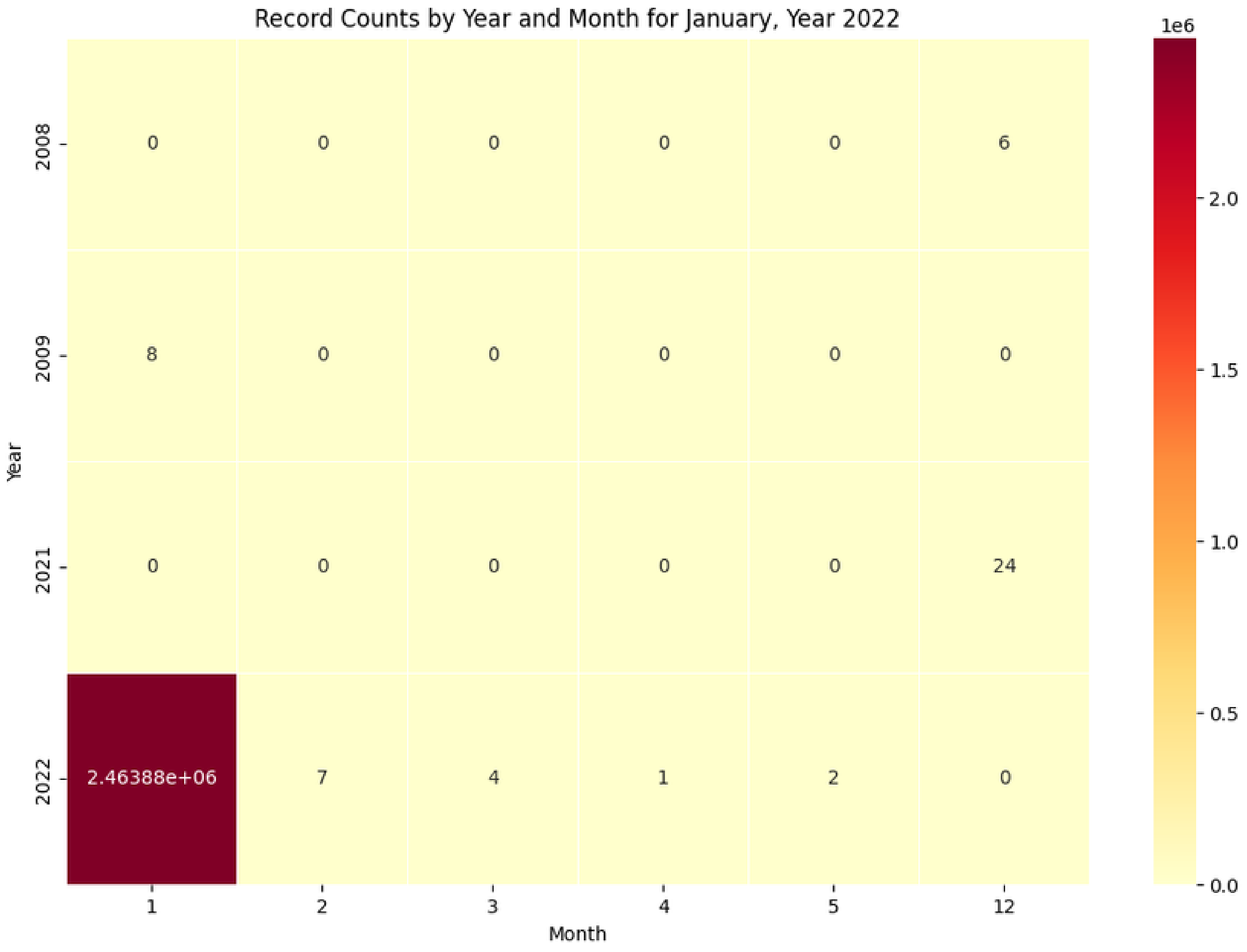
- For this project the data in collected from official NYC TLC Taxi dataset in parquet format for 6 months starting from January to June for the year 2022



Data Quality

January Raw Dataset

Data Entry Issues



Before

	VendorID	tpep_pickup_datetime	tpep_dropoff_datetime	passenger_count	trip_distance	RatecodeID	store_and_fwd_flag	PULocationID	DOLocationID
0	1	2022-01-01 00:35:40	2022-01-01 00:53:29	2.0	3.80	1.0	N	142	236
1	1	2022-01-01 00:33:43	2022-01-01 00:42:07	1.0	2.10	1.0	N	236	42
2	2	2022-01-01 00:53:21	2022-01-01 01:02:19	1.0	0.97	1.0	N	166	166
3	2	2022-01-01 00:25:21	2022-01-01 00:35:23	1.0	1.09	1.0	N	114	68
4	2	2022-01-01 00:36:48	2022-01-01 01:14:20	1.0	4.30	1.0	N	68	163
...	...	...	...	...	...	...	...	...	...
2463926	2	2022-01-31 23:36:53	2022-01-31 23:42:51	NaN	1.32	NaN	None	90	170
2463927	2	2022-01-31 23:44:22	2022-01-31 23:55:01	NaN	4.19	NaN	None	107	75
2463928	2	2022-01-31 23:39:00	2022-01-31 23:50:00	NaN	2.10	NaN	None	113	246
2463929	2	2022-01-31 23:36:42	2022-01-31 23:48:45	NaN	2.92	NaN	None	148	164
2463930	2	2022-01-31 23:46:00	2022-02-01 00:13:00	NaN	8.94	NaN	None	186	181

2463931 rows × 21 columns

After

	VendorID	tpep_pickup_datetime	tpep_dropoff_datetime	passenger_count	trip_distance	RatecodeID	store_and_fwd_flag	PULocationID	DOLocationID
0	1	2022-01-01 00:35:40	2022-01-01 00:53:29	2.0	3.80	1.0	N	142	236
1	1	2022-01-01 00:33:43	2022-01-01 00:42:07	1.0	2.10	1.0	N	236	42
2	2	2022-01-01 00:53:21	2022-01-01 01:02:19	1.0	0.97	1.0	N	166	166
3	2	2022-01-01 00:25:21	2022-01-01 00:35:23	1.0	1.09	1.0	N	114	68
4	2	2022-01-01 00:36:48	2022-01-01 01:14:20	1.0	4.30	1.0	N	68	163
...	...	...	...	...	...	...	...	...	...
2463926	2	2022-01-31 23:36:53	2022-01-31 23:42:51	NaN	1.32	NaN	None	90	170
2463927	2	2022-01-31 23:44:22	2022-01-31 23:55:01	NaN	4.19	NaN	None	107	75
2463928	2	2022-01-31 23:39:00	2022-01-31 23:50:00	NaN	2.10	NaN	None	113	246
2463929	2	2022-01-31 23:36:42	2022-01-31 23:48:45	NaN	2.92	NaN	None	148	164
2463930	2	2022-01-31 23:46:00	2022-02-01 00:13:00	NaN	8.94	NaN	None	186	181

2463879 rows × 21 columns

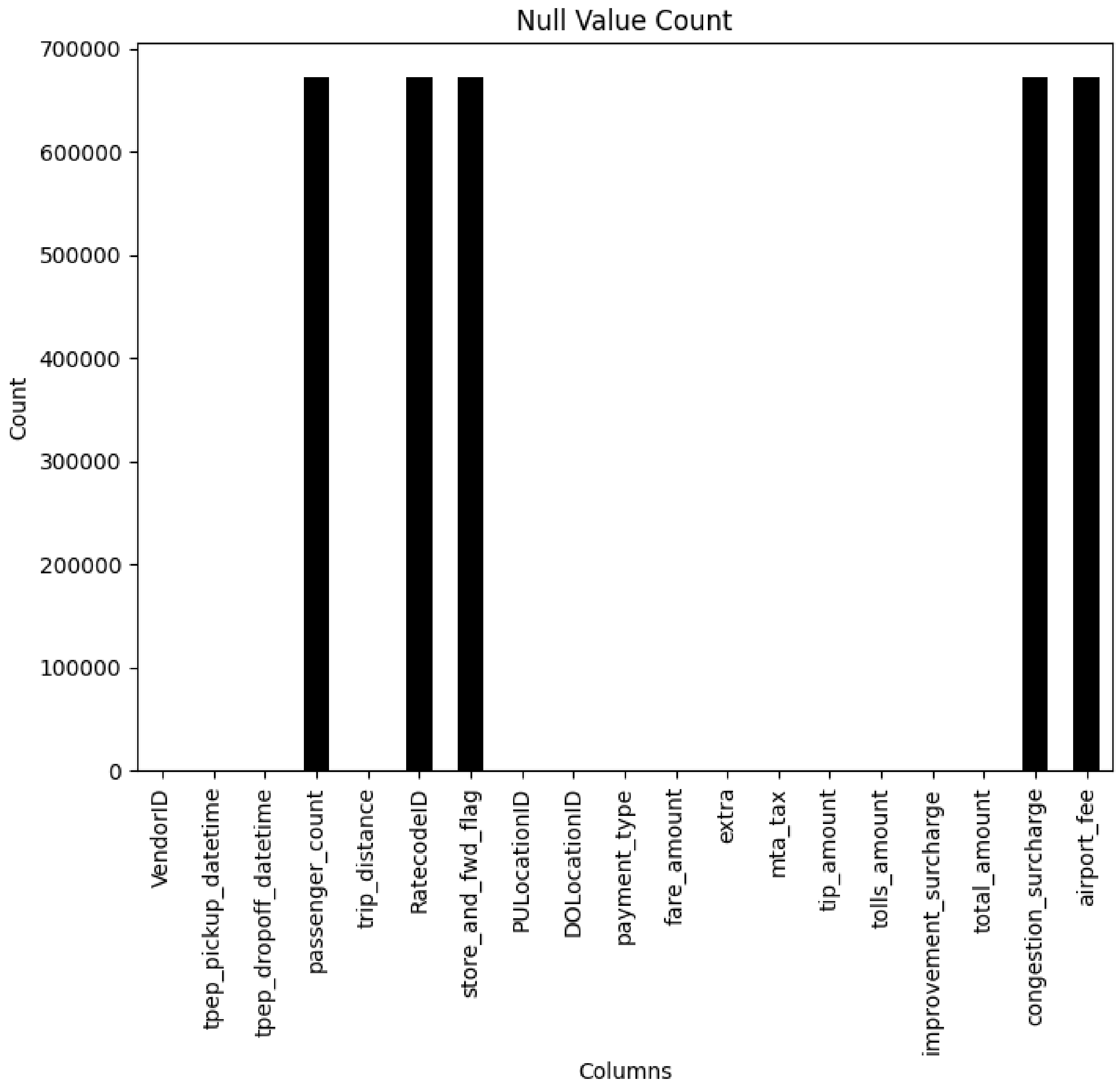
Raw Dataset

Merged Dataset for 6 months after removal of data entry issues

	VendorID	tpep_pickup_datetime	tpep_dropoff_datetime	passenger_count	trip_distance	RatecodeID	store_and_fwd_flag	PULocationID	DOLocationID
0	1	2022-01-01 00:35:40	2022-01-01 00:53:29	2.0	3.80	1.0	N	142	236
1	1	2022-01-01 00:33:43	2022-01-01 00:42:07	1.0	2.10	1.0	N	236	42
2	2	2022-01-01 00:53:21	2022-01-01 01:02:19	1.0	0.97	1.0	N	166	166
3	2	2022-01-01 00:25:21	2022-01-01 00:35:23	1.0	1.09	1.0	N	114	68
4	2	2022-01-01 00:36:48	2022-01-01 01:14:20	1.0	4.30	1.0	N	68	163
...	...	...	...	...	...	...	...	...	...
3558119	1	2022-06-30 23:45:51	2022-06-30 23:51:48	NaN	0.00	NaN	None	148	256
3558120	2	2022-06-30 23:25:00	2022-06-30 23:40:00	NaN	5.01	NaN	None	79	262
3558121	2	2022-06-30 23:29:00	2022-06-30 23:37:00	NaN	1.55	NaN	None	164	79
3558122	2	2022-06-30 23:24:15	2022-06-30 23:50:19	NaN	5.30	NaN	None	211	239
3558123	2	2022-06-30 23:33:53	2022-06-30 23:54:58	NaN	4.41	NaN	None	255	158

19816565 rows × 21 columns

EDA on Raw Dataset




```
Number of duplicate rows: 0
Duplicate rows:
Empty DataFrame
Columns: [VendorID, tpep_pickup_datetime, tpep_dropoff_datetime, passenger_count, trip_distance, RatecodeID, store_and_fwd_fla
g, PULocationID, DOLocationID, payment_type, fare_amount, extra, mta_tax, tip_amount, tolls_amount, improvement_surcharge, tota
l_amount, congestion_surcharge, airport_fee, year, month]
Index: []

[0 rows x 21 columns]
```

Data Pre-Processing

	VendorID	tpep_pickup_datetime	tpep_dropoff_datetime	trip_distance	PULocationID	DOLocationID	payment_type	fare_amount	extra	mta_tax	tip_am
0	1	2022-01-01 00:35:40	2022-01-01 00:53:29	3.80	142	236	1	14.50	3.0	0.5	
1	1	2022-01-01 00:33:43	2022-01-01 00:42:07	2.10	236	42	1	8.00	0.5	0.5	
2	2	2022-01-01 00:53:21	2022-01-01 01:02:19	0.97	166	166	1	7.50	0.5	0.5	
3	2	2022-01-01 00:25:21	2022-01-01 00:35:23	1.09	114	68	2	8.00	0.5	0.5	
4	2	2022-01-01 00:36:48	2022-01-01 01:14:20	4.30	68	163	1	23.50	0.5	0.5	
...	...	...	...	...	...	...	...	...	...	...	
3558119	1	2022-06-30 23:45:51	2022-06-30 23:51:48	0.00	148	256	0	9.20	0.5	0.5	
3558120	2	2022-06-30 23:25:00	2022-06-30 23:40:00	5.01	79	262	0	18.86	0.0	0.5	
3558121	2	2022-06-30 23:29:00	2022-06-30 23:37:00	1.55	164	79	0	10.03	0.0	0.5	
3558122	2	2022-06-30 23:24:15	2022-06-30 23:50:19	5.30	211	239	0	24.34	0.0	0.5	
3558123	2	2022-06-30 23:33:53	2022-06-30 23:54:58	4.41	255	158	0	21.16	0.0	0.5	

19816565 rows × 16 columns

DATASET	STATISTICS
PROCESSED DATASET	19616565 X 16

Data Transformation

Transformed Dataset

	pickup_date	PULocationID	daily_pickups
0	2022-01-01	1	43
1	2022-01-01	3	2
2	2022-01-01	4	122
3	2022-01-01	5	1
4	2022-01-01	7	83
...	...	...	...
39302	2022-06-30	261	543
39303	2022-06-30	262	1608
39304	2022-06-30	263	2284
39305	2022-06-30	264	1257
39306	2022-06-30	265	360

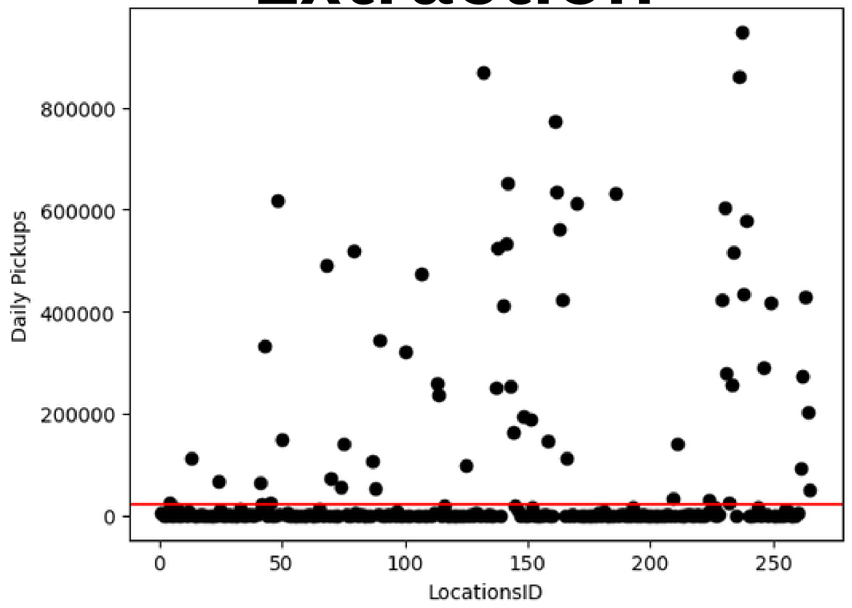
39307 rows × 3 columns

Feature Extraction

	pickup_date	PULocationID	daily_pickups	Pickup_year	pickup_dayofweek	day	month
0	2022-01-01	1	43	2022	5	1	1
1	2022-01-01	3	2	2022	5	1	1
2	2022-01-01	4	122	2022	5	1	1
3	2022-01-01	5	1	2022	5	1	1
4	2022-01-01	7	83	2022	5	1	1
...	...	...	...	...	...	...	...
39302	2022-06-30	261	543	2022	3	30	6
39303	2022-06-30	262	1608	2022	3	30	6
39304	2022-06-30	263	2284	2022	3	30	6
39305	2022-06-30	264	1257	2022	3	30	6
39306	2022-06-30	265	360	2022	3	30	6

39307 rows × 7 columns

Outliers After Feature Extraction



Data Transformation

Dataset of After Outlier Removal from Feature Dataset

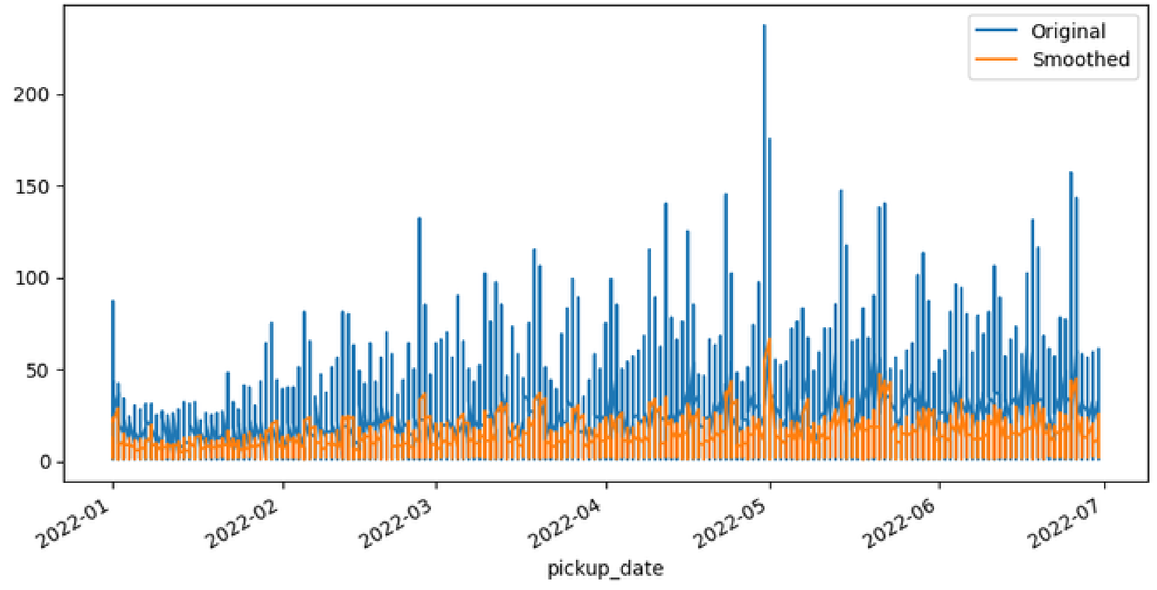
	pickup_date	PULocationID	daily_pickups	Pickup_year	pickup_dayofweek	day	month
0	2022-01-01	1	43	2022	5	1	1
1	2022-01-01	3	2	2022	5	1	1
3	2022-01-01	5	1	2022	5	1	1
5	2022-01-01	8	1	2022	5	1	1
6	2022-01-01	10	14	2022	5	1	1
...	...	...	...	...	...	...	...
39296	2022-06-30	254	3	2022	3	30	6
39298	2022-06-30	256	30	2022	3	30	6
39299	2022-06-30	257	1	2022	3	30	6
39300	2022-06-30	258	5	2022	3	30	6
39301	2022-06-30	260	23	2022	3	30	6

26456 rows × 7 columns

Smoothed Dataset

	pickup_date	daily_pickups	PULocationID	Pickup_year	pickup_dayofweek	day	month
6	2022-01-01	12.2	10	2022	5	1	1
7	2022-01-01	3.8	11	2022	5	1	1
8	2022-01-01	14.0	12	2022	5	1	1
10	2022-01-01	14.2	14	2022	5	1	1
11	2022-01-01	17.0	17	2022	5	1	1
...	...	...	...	...	...	...	...
39296	2022-06-30	2.4	254	2022	3	30	6
39298	2022-06-30	7.8	256	2022	3	30	6
39299	2022-06-30	7.6	257	2022	3	30	6
39300	2022-06-30	8.0	258	2022	3	30	6
39301	2022-06-30	12.4	260	2022	3	30	6

26452 rows × 7 columns



Data Preparation

Training Dataset

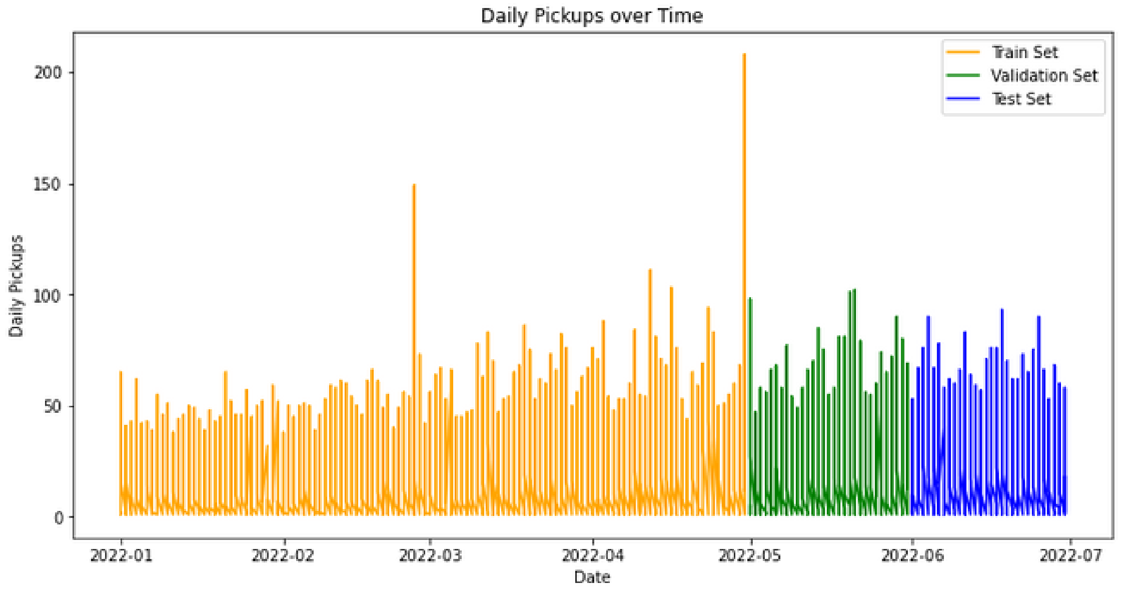
	pickup_date	PULocationID	daily_pickups	Pickup_year	pickup_dayofweek	day	month	normalized_pickups
0	2022-01-01	1	2	2022	5	1	1	0.004831
1	2022-01-01	5	1	2022	5	1	1	0.000000
2	2022-01-01	7	65	2022	5	1	1	0.309179
3	2022-01-01	8	1	2022	5	1	1	0.000000
4	2022-01-01	10	8	2022	5	1	1	0.033816
...	...	...	...	...	...	...	...	...
16223	2022-04-30	254	1	2022	5	30	4	0.000000
16224	2022-04-30	255	208	2022	5	30	4	1.000000
16225	2022-04-30	256	113	2022	5	30	4	0.541063
16226	2022-04-30	260	25	2022	5	30	4	0.115942
16227	2022-04-30	265	12	2022	5	30	4	0.053140

16228 rows × 8 columns

Validation Dataset

	pickup_date	PULocationID	daily_pickups	Pickup_year	pickup_dayofweek	day	month	normalized_pickups
16228	2022-05-01	1	5	2022	6	1	5	0.019324
16229	2022-05-01	3	3	2022	6	1	5	0.009662
16230	2022-05-01	7	81	2022	6	1	5	0.386473
16231	2022-05-01	10	35	2022	6	1	5	0.164251
16232	2022-05-01	12	46	2022	6	1	5	0.217391
...	...	...	...	...	...	...	...	...
20492	2022-05-31	256	17	2022	1	31	5	0.077295
20493	2022-05-31	257	5	2022	1	31	5	0.019324
20494	2022-05-31	259	1	2022	1	31	5	0.000000
20495	2022-05-31	260	20	2022	1	31	5	0.091787
20496	2022-05-31	265	13	2022	1	31	5	0.057971

4269 rows × 8 columns



Test Dataset

	pickup_date	PULocationID	daily_pickups	Pickup_year	pickup_dayofweek	day	month	normalized_pickups
20497	2022-06-01	1	2	2022	2	1	6	0.004831
20498	2022-06-01	7	45	2022	2	1	6	0.212560
20499	2022-06-01	10	29	2022	2	1	6	0.135266
20500	2022-06-01	11	1	2022	2	1	6	0.000000
20501	2022-06-01	12	49	2022	2	1	6	0.231884
...	...	...	...	...	...	...	...	...
24783	2022-06-30	255	23	2022	3	30	6	0.106280
24784	2022-06-30	256	17	2022	3	30	6	0.077295
24785	2022-06-30	258	4	2022	3	30	6	0.014493
24786	2022-06-30	260	13	2022	3	30	6	0.057971
24787	2022-06-30	265	18	2022	3	30	6	0.082126

4291 rows × 8 columns

DATA SET	STATISTICS
Trainig	16228 x 8
Validation	4269 x 8
Test	4291 x 8

Model Development

LSTM

- LSTM is a form of recurrent neural network (RNN). This LSTM addresses the issue of vanishing gradients that plagues traditional RNNs.
- This time series forecasting algorithm is designed to be easy to use and extremely accurate.

Facebook Prophet

- It is capable of handling absent data and incorporating external variables to enhance its accuracy.
- Weather forecasting, demand forecasting, and sales forecasting have all been successfully performed with the tool.

XG - Boost

- Another ensemble learning algorithm known as Extreme Gradient Boosting (XGBoost) increases its overall accuracy by using weak models, often decision s.
- Gradient boosting is used in XGBoost to iteratively improve a model's performance.

Random Forest

- Random Forest is an ensemble learning model that uses decision trees to make predictions.
- By generating multiple decision trees and combining their predictions, it generates the output.
- It is capable of managing complex datasets and avoiding overfitting.

Model Justification

Random Forest

Model complexity and performance can be balanced well for a large amount of data with multiple features

XG - Boost

An effective method for capturing complex nonlinear relationships between input features and the target variable that is robust to overfitting

LSTM

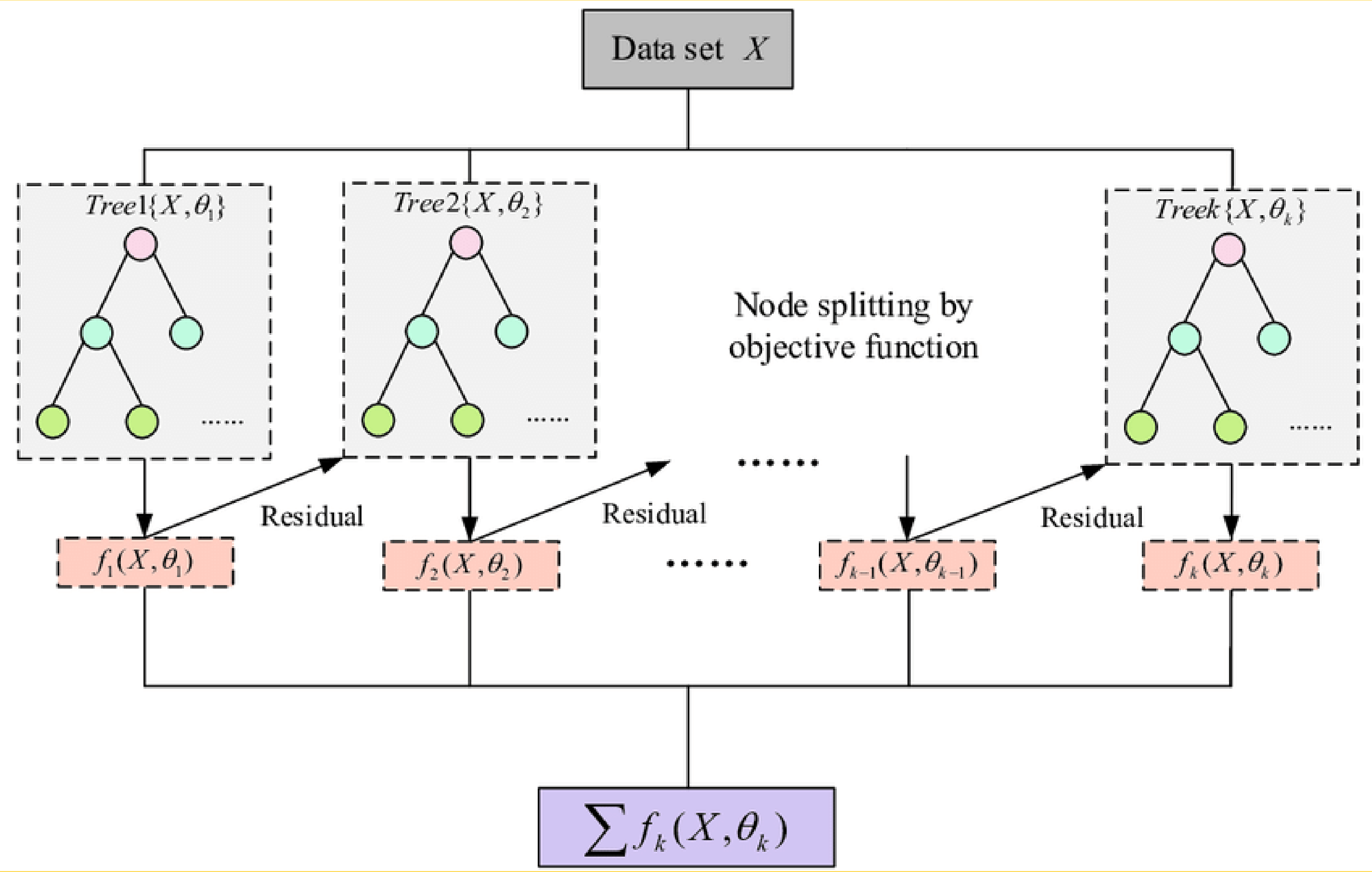
Data patterns can be learned by capturing long-term dependencies

Facebook Prophet

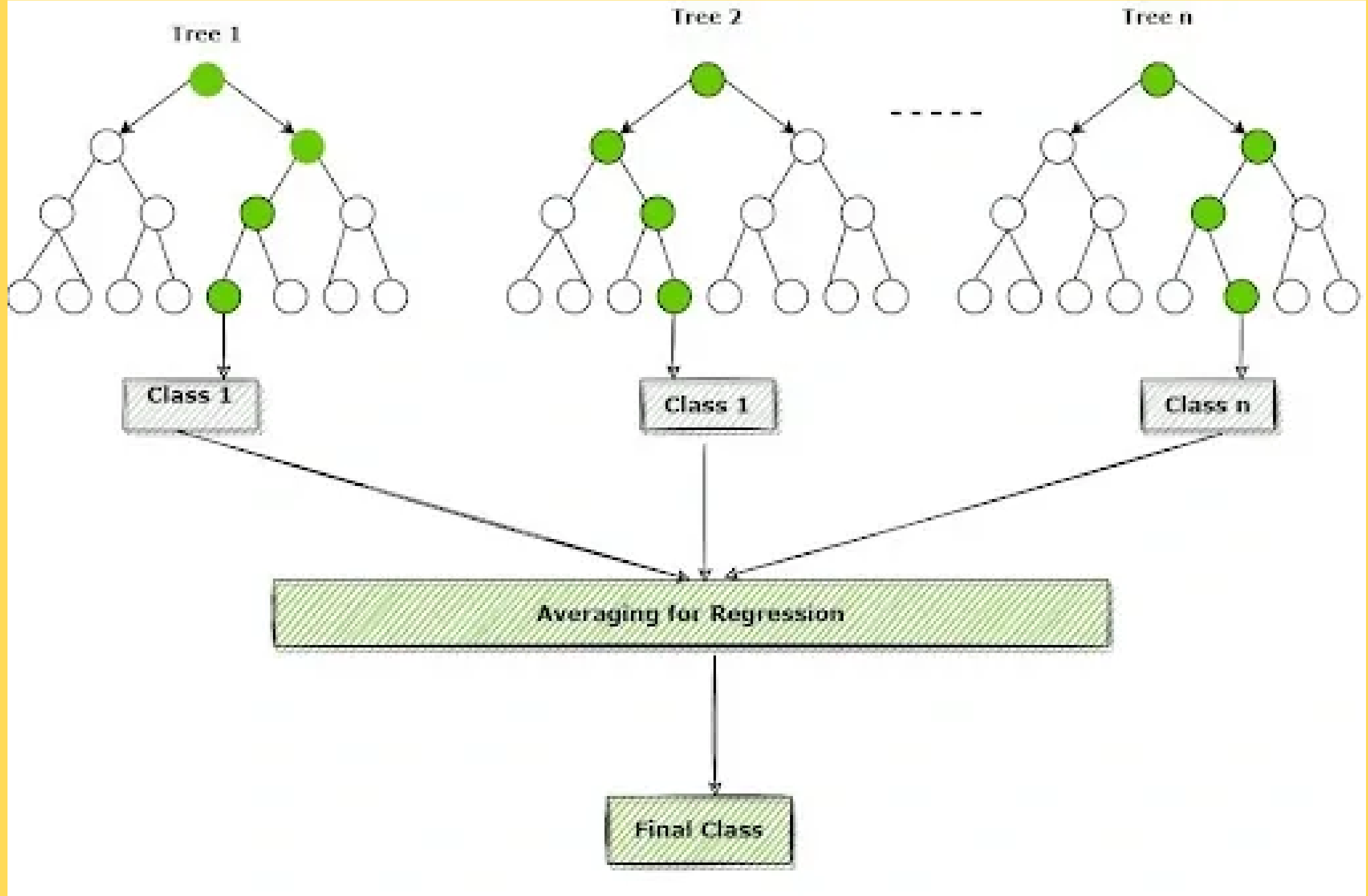
Seasonality and trend changes can be handled by the model

Combination of Random Forest and XG-Boost

A new model was developed by averaging the results of XGBoost and Random Forest.



+



Evaluation Metrics

R- Square

The R-square indicates how well the regression model performs and explains observed data.

MAPE

A MAPE is a measure of how much the predicted value differs from the true value on average

MSE

An average of squared errors is measured by MSE

MAE

Error magnitude averaged without regard to error sign

RMSE

Measures the distance between prediction and true target based on Euclidean distance

Model Comaprisions

	TRAINING DATASET			TESTING DATASET		
	MSE	RMSE	Accurac y	MSE	RMSE	Accuracy
RANDOM FOREST	5.89	2.42	0.866	4.232	2.057	0.889
XG-BOOST	8.24	2.87	0.857	6.148	2.479	0.873
FACEBO OK PROPHET	11.43	3.38	0.568	9.49	3.08	0.662
LSTM	18.96	4.35	0.437	14.34	3.78	0.572
NEW MODEL	3.3124	1.82	0.874	2.43	1.56	0.886

Conclusion

- Random Forest and XG-Boost performed very good results as top two models
- Facebook Prophet outperformed LSTM and stood at third position
- Combination of Random Forest and XG-Boost gave similar results and outperformed the top models w.r.t RMSE metrics

Future Scope

- To improve taxi dispatching system efficiency, we plan to use stream processing frameworks like Apache Kafka or Apache Flink to enable real-time taxi demand prediction.
- As a way to reduce operational costs and improve scalability, we would like to deploy the models on cloud platforms such as AWS and GCP.

References

Askari, B., Quy, T. L., & Ntoutsi, E. (2020). Taxi Demand Prediction using an LSTM-Based Deep Sequence Model and Points of Interest. (2020, July 1). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/9202791

Kankanamge, K. D., Witharanage, Y. R., Withanage, C. S., Hansini, M., Lakmal, D., & Thayasiva, U. (2019). Taxi Trip Travel Time Prediction with Isolated XGBoost Regression. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/8818915

Silveira-Santos, T., González, A. B. R., Rangel, T., Pozo, R. F., Vassallo, J. M., & Díaz, J. J. V. (2022). Were ride-hailing fares affected by the COVID-19 pandemic? Empirical analyses in Atlanta and Boston. Transportation. https://doi.org/10.1007/s11116-022- 10349-x

Wang, Y., & Mi, X. (2018). A Comparative Study on Demand Forecast of Car Sharing Users Based on ARIMA and LSTM. (2020, May 1). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/9237552?casa_token=LpxKbFJoVBQA AAAA:HRPyIzepgOZxLZjFzsNwq9w4EOEhgu78eB0i5dZ4X0Ad2mKC1IrbNfHT2 ditdMFqJFEhIohf

https://www.turing.com/kb/random-forest-algorithm

https://www.researchgate.net/figure/Flow-chart-of-XGBoost_fig3_345327934

Thank

You