

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JNANASANGAMA” BELAGAVI - 590 018

KARNATAKA



REPORT OF INTERNSHIP/PROFESSIONAL PRACTICE

Carried out in



TECH FORTUNE TECHNOLOGIES

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE OF

BACHELOR OF ENGINEERING

IN

Computer Science and Engineering

Submitted by:

ASHWINI
[1CG17CS010]

INTERNAL GUIDE

Ms. Lalitha Bandeppa,
Assistant Professor,
Dept. of CSE,
C.I.T, Gubbi, Tumkur.

EXTERNAL GUIDE

Mr. Mallikarjun V S,
CEO,
Tech Fortune Technologies,
Bangalore.

HOD

Shantala C P,
Professor & Head,
Dept. of CSE,
CIT, Gubbi



Channabasaveshwara Institute of Technology

(Affiliated to VTU, Belgaum & Approved by AICTE, New Delhi)

(NAAC Accredited & ISO 9001:2015 Certified Institution)

NH 206 (B.H. Road), Gubbi, Tumkur – 572216. Karnataka



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2020-2021



Channabasaveshwara Institute of Technology

(Affiliated to VTU, Belgaum & Approved by AICTE, New Delhi)

(NAAC Accredited & ISO 9001:2015 Certified Institution)

NH 206 (B.H. Road), Gubbi, Tumkur – 572216. Karnataka.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2020-2021

UNDERTAKING

I, **ASHWINI** bearing **1CG17CS010**, student of **VIII Semester B.E. in CSE, C.I.T, GUBBI, TUMKUR** hereby declare that the Internship carried out in **TECH FORTUNE TECHNOLOGIES, BANGALORE** and submitted in partial fulfillment of the requirements for the award of the degree **Bachelor of Engineering in COMPUTER SCIENCE & ENGINEERING** of the **Visvesvaraya Technological University , Belagavi** during the academic year 2020-2021.

Place: GUBBI

Date:

ASHWINI

[1CG17CS010]



Channabasaveshwara Institute of Technology

(Affiliated to VTU, Belgaum & Approved by AICTE, New Delhi)

(NAAC Accredited & ISO 9001:2015 Certified Institution)

NH 206 (B.H. Road), Gubbi, Tumkur – 572216. Karnataka.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2020-21

BONAFIDE CERTIFICATE

This is to certify that the Internship carried out in **TECH FORTUNE TECHNOLOGIES, BANGALORE** is a bonafide work of **ASHWINI – 1CG17CS010**, student of **VIII** semester **B.E. - CSE** from **Channabasaveshwara Institute of Technology, Gubbi, Tumkur**, in partial fulfillment of the requirements for the award of degree **B.E.**, in **COMPUTER SCIENCE & ENGINEERING** of **Visvesvaraya Technological University, Belgaum** during the academic year 2020-2021. It is certified that the Internship work carried out was under my supervision and guidance.

Guide

Ms. Lalitha Bandeppa
Assistant Professor
Dept., of CSE
C.I.T, Gubbi.



Channabasaveshwara Institute of Technology

(Affiliated to VTU, Belgaum & Approved by AICTE, New Delhi)
(NAAC Accredited & ISO 9001:2015 Certified Institution)

NH 206 (B.H. Road), Gubbi, Tumkur – 572216. Karnataka.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2020-2021

CERTIFICATE

This is to certify that the Internship entitled “**UBER PICKUPS IN NEW YORK CITY PREDICTION**” has been carried out by **ASHWINI - [1CG17CS010]** bonafide student of **CHANNABASAVESHWARA INSTITUTE OF TECHNOLOGY, GUBBI, TUMKUR**, in partial fulfillment of the requirement for the award of the degree **Bachelor of Engineering** in **COMPUTER SCIENCE & ENGINEERING** from the **Visvesvaraya Technological University, Belagavi** during the year **2020-2021**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report. The Internship report has been approved as it satisfies the academic requirements in respect of Internship/Professional practice prescribed for the said degree.

Signature of Guide

Ms. Lalitha Bandeppa
Assistant Professor,
Dept., of CSE
C.I.T, Gubbi.

Signature of HOD

Ms. Shantala C P
Professor & Head,
Dept., of CSE
C.I.T, Gubbi.

Signature of Principal

Dr. SURESH D S
Director & Principal
C.I.T, Gubbi.

External Viva

Examiners Name

Signature with Date

1. _____
2. _____

ACKNOWLEDGEMENT

Several special people have contributed significantly to this effort. First of all, I am grateful to my institution, **Channabasaveshwara Institute of Technology, Gubbi**, which provides me an opportunity in fulfilling my most cherished desire of reaching my goal.

I, acknowledge and express my sincere thanks to our beloved Director & Principal, **Dr. Suresh D S**, for his many valuable suggestion and continued encouragement by supporting me in mt academic endeavors.

I, express my sincere gratitude to **Mrs. Shantala C P, Professor and Head, Department of CSE** for providing her constructive criticisms and suggestions.

I, extend my gratitude to my Internship guide **Mrs. Lalitha Bandeppa, Assistant Professor, Department of CSE** for her guidance, support and suggestions throughout the period of this Internship.

I express my deep sense of gratitude to **TECH FORTUNE TECHNOLOGIES, BANGALORE** for giving such an opportunity to carry out the internship in their esteemed industry/organization.

I sincerely thank **Mr. Mallikarjun V S, CEO, Tech Fortune Technologies, Bangalore** for exemplary guidance and supervision.

Finally, I would like to thank all the individuals who supported me directly and indirectly for the successful completion of this internship work.

ASHWINI[1CG17CS010]

Ref – TFT/INT/2020/0187

2nd September 2020

TO WHOM IT MAY CONCERN

This is to certify that Ms Ashwini a Student of CIT Gubbi Karnataka, has successfully completed 6 (Six weeks) of the internship program from 15th July 2020 to 2nd September 2020 at Tech Fortune Technologies. During the internship Program, She has been observed to be punctual, hardworking and with about software development. Good knowledge.

We also assure that this student is capable of developing projects on Machine Learning

Technologies learnt and implemented during Internship:

- Python
- Numpy, Pandas, Matplotlib
- Machine Learning algorithms
- Software project implementation.

We wish him every success in life

Yours Faithfully,
For Tech Fortune Technologies



Mallikarjun. V.S
(Operation – Head)

TECH FORTUNE TECHNOLOGIES

#18 22ND MAIN SUBANNA GARDEN, VIJAYANAGAR, BANGALORE - 560040
Ph: +91 9591 68 7143/8317 35 3335 • www.tech-fortune.com | email: info@tech-fortune.com

ABSTRACT

On-demand, app-based ride services like Uber and Lyft have become an important part of today's transportation system with its flexibility and quick responsiveness. Compared with traditional taxi- cabs, Uber-like taxis have loggers to monitor and record trip information such as pickup location and trip distance, which can be a valuable data source for knowledge discovering. Nowadays, a Real-time prediction for ride service demand (always reflected by the number of pickups) is increasingly crucial for the purpose of improving the efficiency and sustainability of the urban transportation system. Newly aroused application topics like ride sharing and autonomous mobility dispatching are based on solid demand predictions. In this paper, we propose a deep learning based approach to make dynamic predictions for Uber pickups using historical data. A Long Short-Term Memory (LSTM) Networks model is developed to learn the long-term dependencies of the pickups over time. With the experimental comparison of time-varying Poisson model and regression tree model, the results demonstrate the superior performance of our proposed deep learning model.

LIST OF FIGURES

FIGURE NAME	PAGE NO.
1. Fig 1.1 : Introduction to Machine Learning	1
2. Fig 1.2 : How does Machine Learning Work	2
3. Fig 1.3 : Techniques of Machine Learning	3
4. Fig 3.1 : Architecture of Uber pickups	7
5. Fig 3.2 : Overview of Machine Learning Algorithms	9
6. Fig 3.3 : Architecture of LSTM	11
7. Fig 5.1 : Import The Libraries	16
8. Fig 5.2 : Loading The Data	16
9. Fig 5.3 : Day of Month and Create a Column	17
10. Fig 5.4 : Analysis of Day Month	18
11. Fig 5.5 : Analysis of Weekday	20
12. Fig 5.6 : Output Graph	21

TABLE OF CONTENTS

CONTENTS	PAGE NO.
1. Introduction	1-4
1.1 Objective	4
1.2 Scope of the Project	4
1.3 Problem Statement	4
2. Literature Survey	5-6
3. System Design	7-12
3.1 Machine Learning Workflow	7-10
3.2 Deep Learning Algorithm	10-12
4. Implementation	13-15
4.1 Technologies used	13-15
4.2 Libraries Imported	15
5. Results	16-21
6. Conclusion	22
7. References	23

CHAPTER 1

INTRODUCTION

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people. Although machine learning is a field within computer science, it differs from traditional computational approaches. In traditional computing, algorithms are sets of explicitly programmed instructions used by computers to calculate or problem solve. Machine learning algorithms instead allow for computers to train on data inputs and use statistical analysis in order to output values that fall within a specific range. Because of this, machine learning facilitates computers in building models from sample data in order to automate decision- making processes based on data inputs.

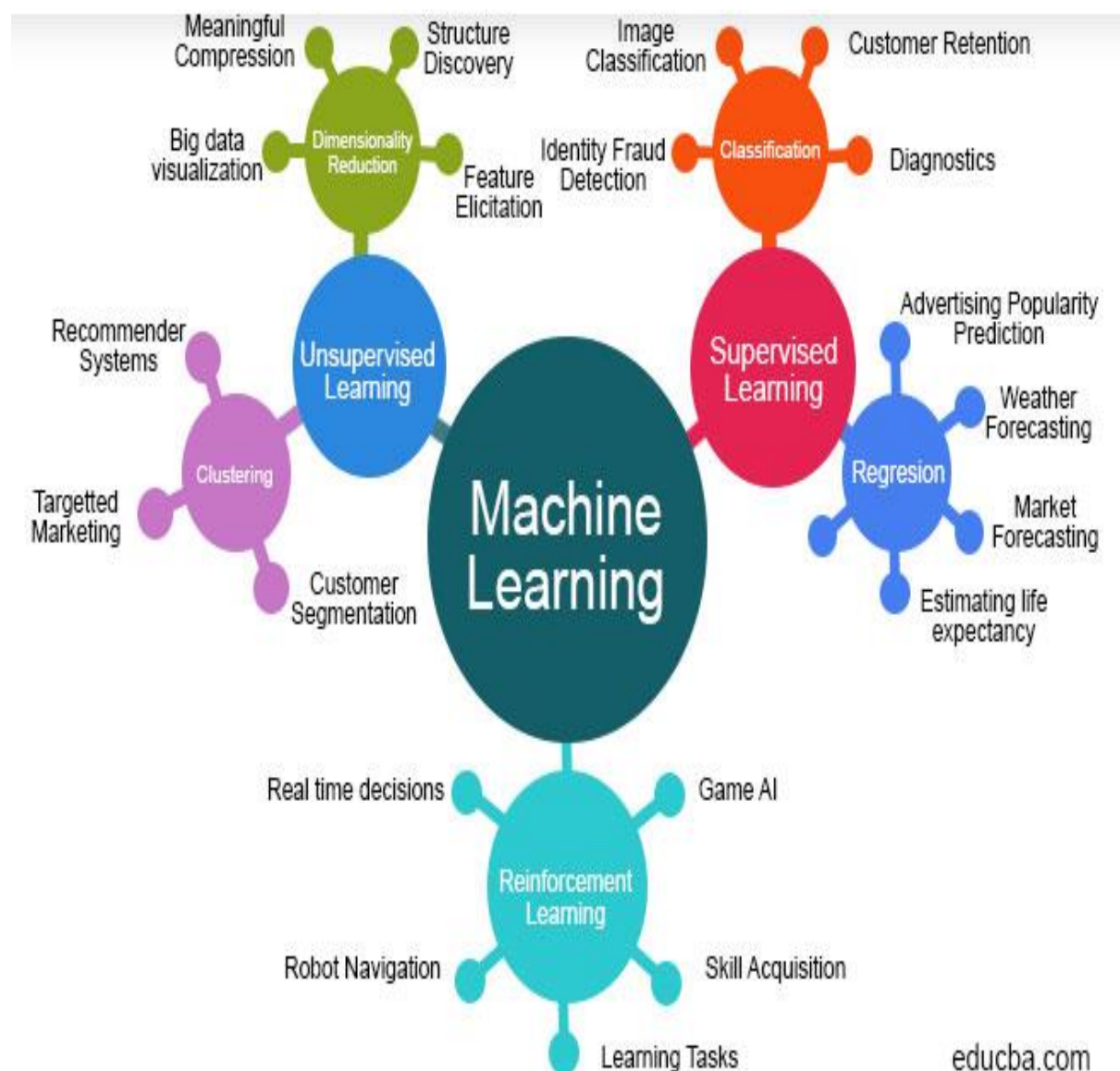


Fig 1.1 : Introduction to Machine Learning

How does Machine Learning work:

Machine learning is a form of artificial intelligence (AI) that teaches computers to think in a similar way to how humans do learning and improving upon past experiences. It works by exploring data, identifying patterns, and involves minimal human intervention. Almost any task that can be completed with a data defined pattern or set of rules can be automated with machine learning. This allows companies to transform processes that were previously only possible for humans to perform think responding to customer service calls, bookkeeping, and reviewing resumes.

Machine learning involves computers discovering how they can perform tasks without being explicitly programmed to do so. It involves computers learning from data provided so that they carry out certain tasks.

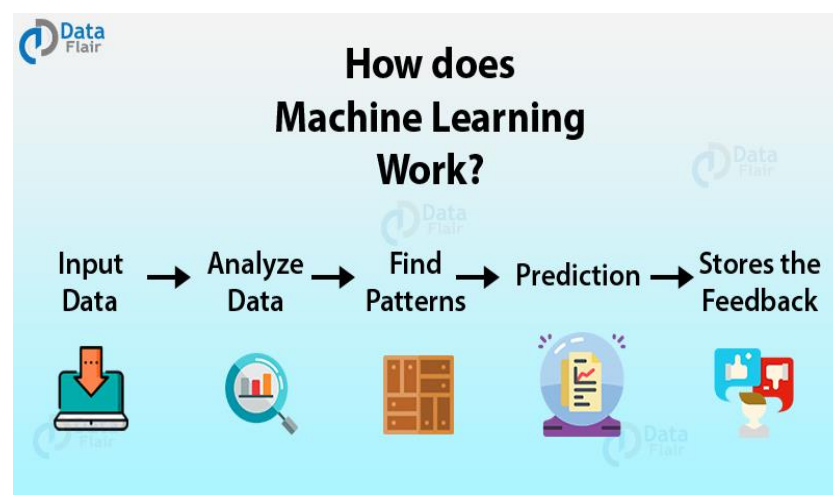


Fig 1.2 : How does Machine Learning Work

Machine Learning Techniques:

Machine learning uses two types of techniques:

- Supervised Learning
- Unsupervised Learning

Supervised learning:

Supervised learning is one of the most basic types of machine learning. In this type, the machine learning algorithm is trained on labelled data. Even though the data needs to be labelled accurately for this method to work, supervised learning is extremely powerful when used in the right circumstances.

In supervised learning, the ML algorithm is given a small training dataset to work with. This training dataset is a smaller part of the bigger dataset and serves to give the algorithm a basic idea of the problem, solution, and data points to be dealt with. The training dataset is also very similar to the final dataset in its characteristics and provides the algorithm with the labelled parameters required for the problem.

Unsupervised learning:

Unsupervised machine learning holds the advantage of being able to work with unlabeled data. This means that human labour is not required to make the dataset machine-readable, allowing much larger datasets to be worked on by the program.

In supervised learning, the labels allow the algorithm to find the exact nature of the relationship between any two data points. However, unsupervised learning does not have labels to work off of, resulting in the creation of hidden structures. Relationships between data points are perceived by the algorithm in an abstract manner, with no input required from human beings.

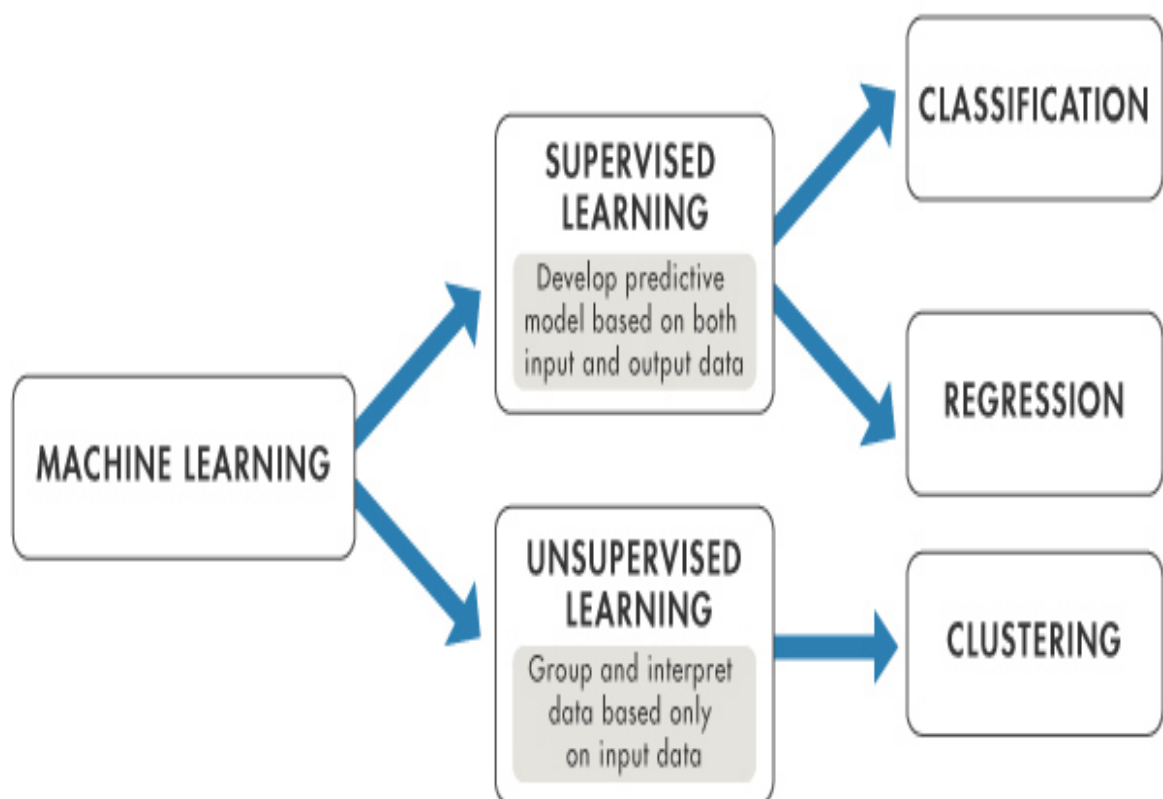


Fig 1.3 : Techniques of Machine Learning

1.1 Objective

The objective of this project is to understand the traffic in different boroughs of New York City and to try and categorize the various zones within various boroughs as being: office destinations, residential destination, popular brunch destination or party destination according to it's popularity given the time-range and the day of the week.

1.2 Scope of the Project

Uber's location based cab and ride sharing app offers useful features for the passengers like pick up and destination address, taxi arrival estimated time, real-time cab tracking, automated e-receipts, SOS, driver details, ratings and reviews and multiple payment options.

1.3 Problem Statement

This report contains the “UBER PICKUPS IN NEW YORK CITY PREDICTION” based on the dataset which contains information about the challenges Uber faces are laws in California that require ride-sharing companies to treat drivers as employees rather than independent contractors. Changes in laws can also affect how much Uber pays in taxes. Some countries and airports have banned ride-sharing companies altogether.

Predict the uber pickups using Deep Learning algorithm.

CHAPTER 2

LITERATURE SURVEY

Taxi complements other public transport modes with a flexible door-to-door service. A study conducted in Taipei City showed that 60–73% of their operation hours, taxi drivers were driving without passengers because they did not know where potential customers were, leaving them no other choice than wandering around the city. There is another study that applied time series forecasting techniques to real-time vehicle location systems in taxis to make short-term predictions of the passenger demand in the city of Porto, Portugal. A predictive model for the number of vacant taxis in a given area based on the time of day, day of the week, and weather conditions in Lisbon, Portugal, is presented.

An extensive variety of spatial information sources such as GPS have recently emerged. A GPS based system is also utilized to track all New York City taxis. Various recent research studies used this data source to analyze different aspects of taxi ridership in NYC. One recent study analyzed travel times and found that travel times from truck and taxi GPS data can be better Correa, Xie, Ozbay

estimated during AM and PM periods than during night time, which indicates that speed differences between taxis and trucks are greater for free-flow conditions. Research on modeling the variation of taxi pick-ups was developed using Poisson and negative binomial models, have been applied in using NYC taxi data. The model suggests that adjacent census tracts have correlated residuals, meaning that spatial autocorrelation exists.

Other studies that utilized NYC taxi trip data estimated a binary logit model to model the mode choice between transit and taxi modes, compared trip characteristics between summer (July) and non-summer (March) months, and developed a data visualization tool namely, TaxiVis, which is a software implementation that allows users to visually query taxi trips by considering spatial, temporal, and other constraints.

Another study used ten-month NYC taxi trip data from 2010 to estimate a multiple linear regression model for each hour of the day to model NYC taxi pick-ups and drop-offs. The results identified six important explanatory variables for taxi trips, which include population, education, age, income, transit access time, and employment, where the influence of these factors on taxi pick-ups and drop-offs changed at different times of the day. To model spatial variation of taxi trip demand and supply in NYC, Poisson-Gamma-Conditional Autoregressive (CAR) model is developed using a ten-month taxi data set in NYC York City. The errors of the CAR model provide insights into when and where there are insufficient taxi supply or surplus taxi supply relative to taxi demand.

In spatial analysis, spatial dependence can be modeled in two ways: using an error term and using a spatially lagged dependent variable. The former way is referred to as the spatial error specification that assumes the spatial dependence is only due to spatial error correlation effects. The latter is denoted as the spatial lag

specification which allows spatial dependence through both spatial error correlation effects and spatial spillover effects. An appropriate consideration of spatial dependence can help adjust the effects of casual factors in the statistic models.

Another study was performed to develop an incident duration model (22), which can account for the spatial dependence of duration observations, to investigate the impacts of a hurricane on incident duration. Moran's I statistics confirmed that durations of the neighboring incidents were spatially correlated. Moreover, Lagrange Multiplier tests suggested that the spatial dependence should be captured in a spatial lag specification. A spatial error model, a spatial lag model, and a linear model without consideration of spatial effects were estimated for incident duration.

Previous studies developed spatio-temporal models for taxi demand, detailed Uber pick-up data were not used in those studies, thus the relationship between Taxi and Uber in NYC has not been explicitly investigated. In other words, previous studies didn't investigate the effects of Uber on the overall taxi demand. Moreover, those studies didn't use spatial error and spatial lag models to account for spatial correlation between dependent and independent variables. In this paper, we intent to fill these gaps using Uber and taxi data from NYC.

CHAPTER 3

SYSTEM DESIGN

All of us are accustomed to Uber companies. A consumer can request a trip via the applying and inside a couple of minutes, a driver arrives close by his/her location to take them to their vacation spot. Earlier Uber was constructed on the “**monolithic**” software program structure mannequin. That they had a backend service, a frontend service, and a single database. They used Python and its frameworks and SQLAlchemy because the ORM-layer to the database. This structure was fantastic for a small variety of journeys in just a few cities however when the service began increasing in different cities Uber crew began going through the problem with the applying. After the 12 months 2014 Uber crew determined to modify to the “**service-oriented structure**” and now Uber additionally handles meals supply and cargo.

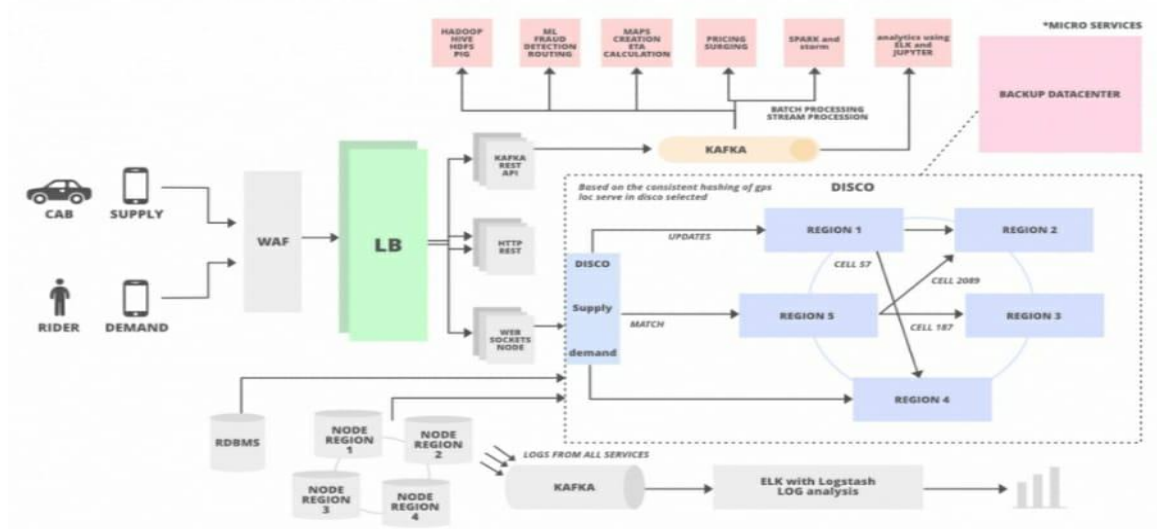


Fig 3.1: Architecture of Uber Pickups

3.1 Machine Learning Workflow

The various stages involved in the machine learning workflow are –

1. Data Collection,
2. Data Preparation,
3. Choosing Learning Algorithm,
4. Training Model,
5. Evaluating Model,
6. Predictions

1. Data Collection-

In this stage,

- Data is collected from different sources.
- The type of data collected depends upon the type of desired project.
- Data may be collected from various sources such as files, databases etc.
- The quality and quantity of gathered data directly affects the accuracy of the desired system.

2. Data Preparation-

In this stage,

- Data preparation is done to clean the raw data.
- Data collected from the real world is transformed to a clean dataset.
- Raw data may contain missing values, inconsistent values, duplicate instances etc.
- So, raw data cannot be directly used for building a model.

Different methods of cleaning the dataset are-

- Ignoring the missing values
- Removing instances having missing values from the dataset.
- Estimating the missing values of instances using mean, median or mode.
- Removing duplicate instances from the dataset.
- Normalizing the data in the dataset.

This is the most time consuming stage in machine learning workflow.

3. Choosing Learning Algorithm-

In this stage,

- The best performing learning algorithm is researched.
- It depends upon the type of problem that needs to be solved and the type of data we have.
- If the problem is to classify and the data is labeled, classification algorithms are used.
- If the problem is to perform a regression task and the data is labeled, regression algorithms are used.
- If the problem is to create clusters and the data is unlabeled, clustering algorithms are used.

The following chart provides the overview of learning algorithms-

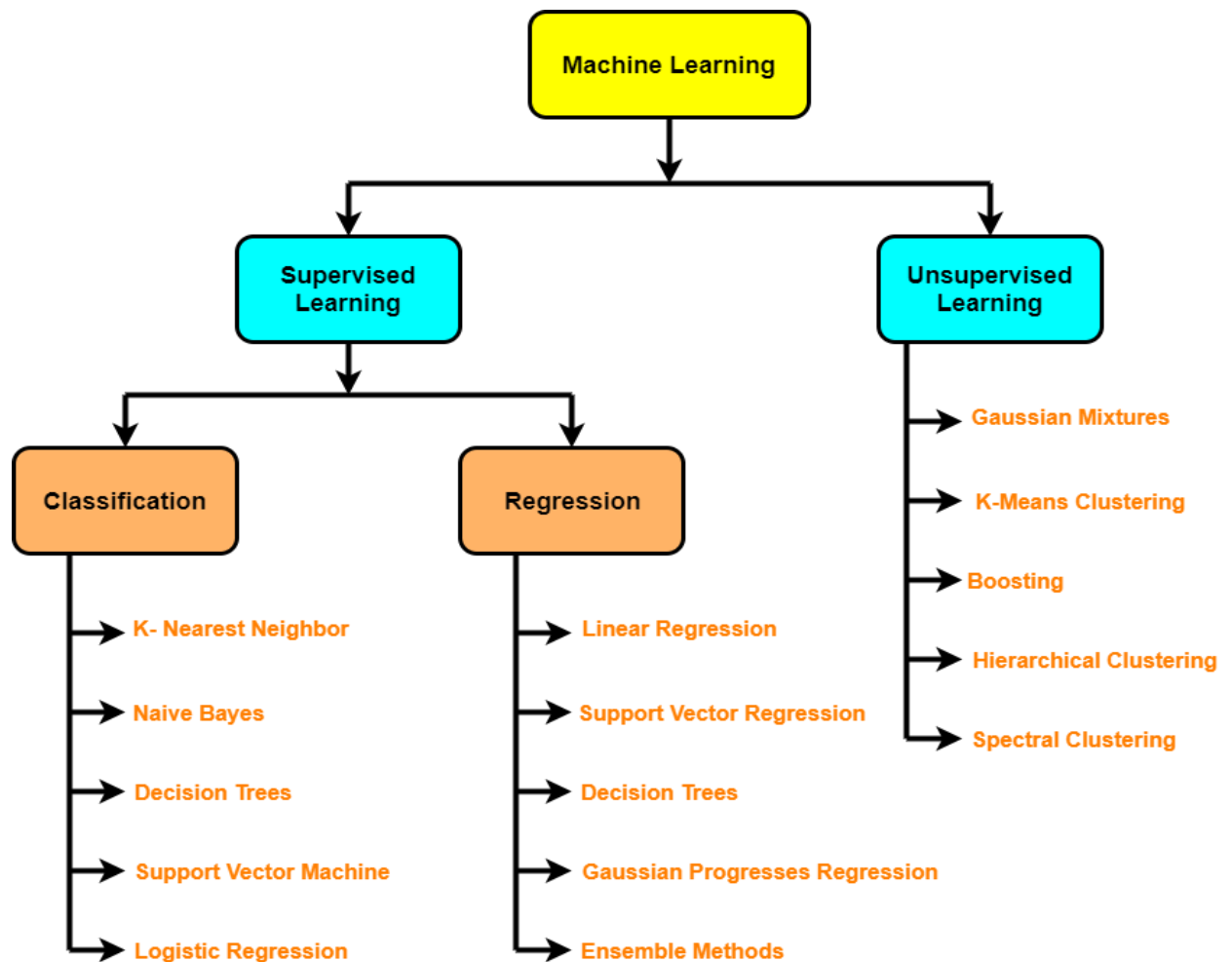
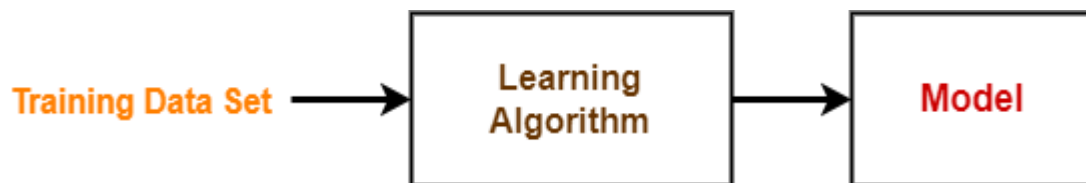


Fig 3.2: Overview of Machine Learning Algorithms

4. Training Model-

In this stage,

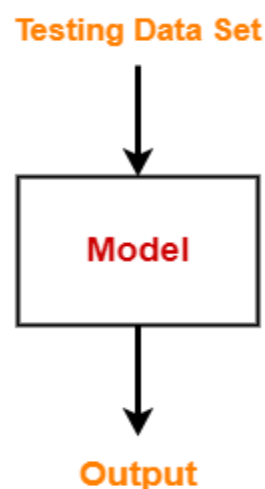
- The model is trained to improve its ability.
- The dataset is divided into training dataset and testing dataset.
- The training and testing split is order of 80/20 or 70/30.
- It also depends upon the size of the dataset.
- Training dataset is used for training purpose.
- Testing dataset is used for the testing purpose.
- Training dataset is fed to the learning algorithm.
- The learning algorithm finds a mapping between the input and the output and generates the model.



5. Evaluating Model-

In this stage,

- The model is evaluated to test if the model is any good.
- The model is evaluated using the kept-aside testing dataset.
- It allows to test the model against data that has never been used before for training.
- Metrics such as accuracy, precision, recall etc are used to test the performance.
- If the model does not perform well, the model is re-built using different hyper parameters.
- The accuracy may be further improved by tuning the hyper parameters.



6. Predictions-

In this stage,

- The built system is finally used to do something useful in the real world.
- Here, the true value of machine learning is realized.

3.2 Deep Learning Algorithm

That's the idea behind a deep learning algorithm. You get input from observation and you put your input into one layer. That layer creates an output which in turn becomes the input for the next layer, and so on. This happens over and over until your final output signal.

Deep learning is a class of machine learning algorithms that uses multiple layers to progressively extract higher-level features from the raw input. For example, in image processing, lower layers may identify edges, while higher layers may identify the concepts relevant to a human such as digits or letters or faces.

LSTM

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems. This is a behavior required in complex problem domains like machine translation, speech recognition, and more. LSTMs are a complex area of deep learning.

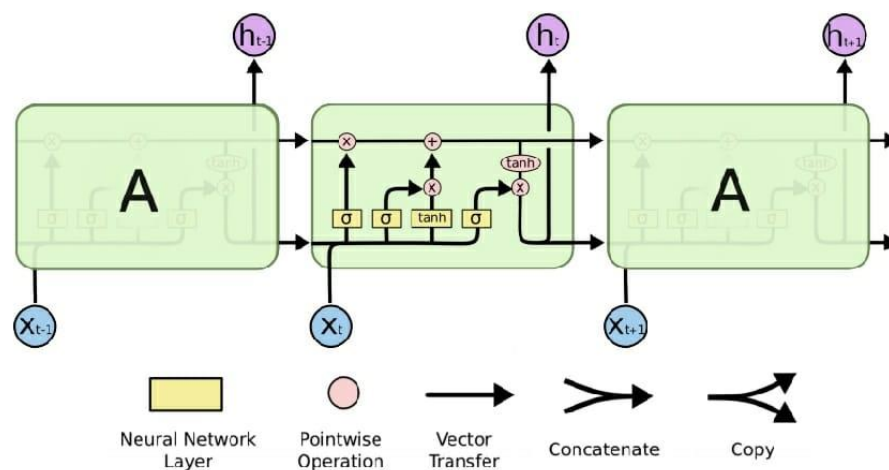


Fig 3.3: LSTM Architecture

Advantages of Deep Learning

1. Electronics: **Deep learning** is being utilized in automated speech translation. You can think of home assistance devices which respond to your voice and understand your preferences.
2. Automated driving: With the help of **deep learning**, automotive researchers are now able to detect objects like traffic lights, stop signs etc automatically. They're also using it to detect pedestrians that helps lower accidents.
3. Medical research: **Deep learning** is being used by cancer researchers to detect cancer cells automatically.

Disadvantages of Deep Learning

1. It requires very large amount of data in order to perform better than other techniques.
2. It is extremely expensive to train due to complex data models. Moreover **deep learning** requires expensive GPUs and hundreds of machines.
3. There is no standard theory to guide you in selecting right deep learning tools as it requires knowledge of topology, training method and other parameters. As a result it is difficult to be adopted by less skilled people.
4. It is not easy to comprehend output based on mere learning and requires classifiers to do so. Convolutional neural network based algorithms perform such tasks.

Deep learning is a class of machine learning algorithms that uses multiple layers to progressively extract higher-level features from the raw input. For example, in image processing, lower layers may identify edges, while higher layers may identify the concepts relevant to a human such as digits or letters or faces.

CHAPTER 4

IMPLEMENTATION

4.1 Technologies used:

1. Anaconda

Anaconda is a distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment. The distribution includes data-science packages suitable for Windows, Linux, and macOS. It is developed and maintained by Anaconda, Inc., which was founded by Peter Wang and Travis Oliphant in 2012.

2. Jupyter Notebook

The Jupyter Notebook is an open source web application and an interactive computing environment that you can use to create and share documents that contain live code, equations, visualizations, and text.

These documents provide a complete and self-contained record of a computation that can be converted to various formats and shared with others using email, Dropbox, version control systems (like git/GitHub) or nbviewer.jupyter.org.

The Jupyter Notebook combines three components:

- **The notebook web application:** An interactive web application for writing and running code interactively and authoring notebook documents.
- **Kernels:** Separate processes started by the notebook web application that runs users' code in a given language and returns output back to the notebook web application. The kernel also handles things like computations for interactive widgets, tab completion and introspection.
- **Notebook documents:** Self-contained documents that contain a representation of all content visible in the notebook web application, including inputs and outputs of the computations, narrative text, equations, images, and rich media representations of objects. Each notebook document has its own kernel.

3. Python

Python is a dynamic, high level, free open source and interpreted programming language. It supports object-oriented programming as well as procedural oriented programming.

Features of Python

1.Easy to code: Python is a high-level programming language. Python is very easy to learn the language as compared to other languages like C, C#, Javascript, Java, etc. It is very easy to code in python language and anybody can learn python basics in a few hours or days. It is also a developer-friendly language.

2. Free and Open Source: Python language is freely available at the official website. Since it is open-source, this means that source code is also available to the public.

3. Object-Oriented Language: One of the key features of python is Object-Oriented programming. Python supports object-oriented language and concepts of classes, objects encapsulation, etc.

4.GUI Programming Support: Graphical User interfaces can be made using a module such as PyQt5, PyQt4, wxPython, or Tk in python. PyQt5 is the most popular option for creating graphical apps with Python.

5. High-Level Language: Python is a high-level language. When we write programs in python, we do not need to remember the system architecture, nor do we need to manage the memory.

6. Extensible feature: Python is a **Extensible** language. We can write us some Python code into C or C++ language and also we can compile that code in C/C++ language.

7. Python is Portable language: Python language is also a portable language. For example, if we have python code for windows and if we want to run this code on other platforms such as Linux, Unix, and Mac then we do not need to change it, we can run this code on any platform.

8. Python is Integrated language: Python is also an Integrated language because we can easily integrated python with other languages like c, c++, etc.

9. Interpreted Language: Python is an Interpreted Language because Python code is executed line by line at a time. like other languages C, C++, Java, etc. there is no need to compile python code this makes it easier to debug our code. The source code of python is converted into an immediate form called **bytecode**.

10. Large Standard Library: Python has a large standard library which provides a rich set of module and functions so you do not have to write your own code for every single thing. There are many libraries present in python for such as regular expressions, unit-testing, web browsers, etc.

11. Dynamically Typed Language: Python is a dynamically-typed language. That means the type (for example- int, double, long, etc.) for a variable is decided at run time not in advance because of this feature we don't need to specify the type of variable.

4.2 Libraries Imported

import numpy as np - Numpy provides a large set of numeric datatypes that you can use to construct arrays. Numpy tries to guess a datatype when you create an array, but functions that construct arrays usually also include an optional argument to explicitly specify the datatype.

import pandas as pd - pandas is a popular Python-based data analysis toolkit which can be imported using import pandas as pd. It presents a diverse range of utilities, ranging from parsing multiple file formats to converting an entire data table into a NumPy matrix array. This makes pandas a trusted ally in data science and machine learning. Similar to NumPy, pandas deals primarily with data in 1-D and 2-D arrays; however, pandas handles the two differently.

import seaborn as sns - Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures. Seaborn helps you explore and understand your data. Its plotting functions operate on dataframes and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. Its dataset-oriented, declarative API lets you focus on what the different elements of your plots mean, rather than on the details of how to draw them.

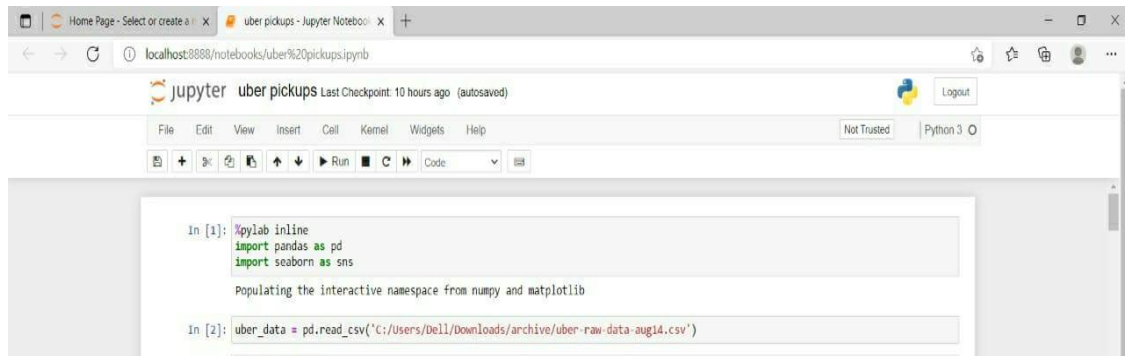
import matplotlib.pyplot as plt - matplotlib.pyplot is stateful, in that it keeps track of the current figure and plotting area, and the plotting functions are directed to the current axes and can be imported using import matplotlib.pyplot as plt.

%matplotlib inline - %matplotlib inline sets the backend of matplotlib to the 'inline' backend: With this backend, the output of plotting commands is displayed inline within frontends like the Jupyter notebook, directly below the code cell that produced it.

from sklearn.model_selection - sklearn model selection for splitting data arrays into two subsets: for training data and for testing data. With this function, you don't need to divide the dataset manually. By default, sklearn train_test_split will make random partitions for the two subsets.

CHAPTER 5

RESULTS



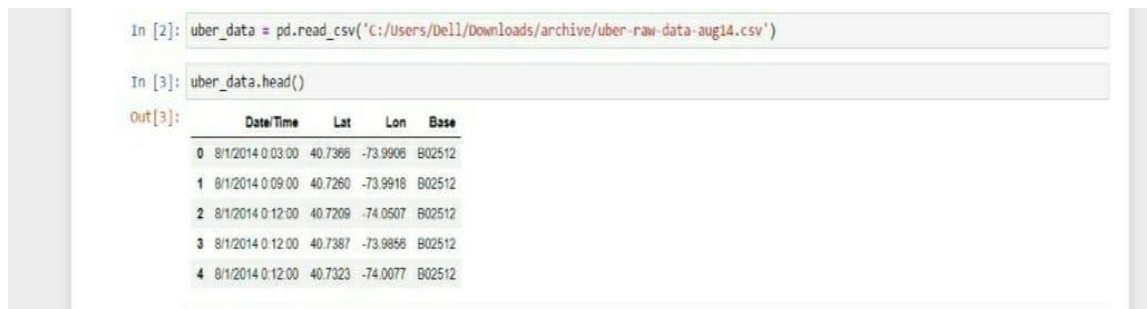
The screenshot shows a Jupyter Notebook interface with the following code in the first two cells:

```
In [1]: %pylab inline
import pandas as pd
import seaborn as sns

Populating the interactive namespace from numpy and matplotlib

In [2]: uber_data = pd.read_csv('C:/Users/Dell/Downloads/archive/uber-raw-data-aug14.csv')
```

Fig 5.1: Import the libraries



The screenshot shows the second cell of the Jupyter Notebook being executed, displaying the first five rows of the loaded data:

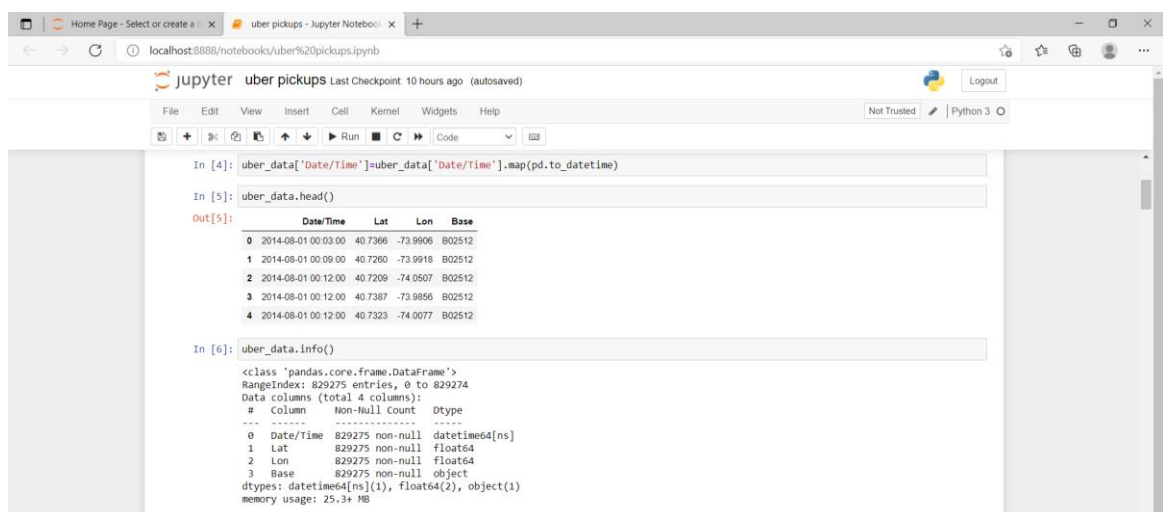
```
In [2]: uber_data = pd.read_csv('C:/Users/Dell/Downloads/archive/uber-raw-data-aug14.csv')

In [3]: uber_data.head()

Out[3]:
```

	Date/Time	Lat	Lon	Base
0	8/1/2014 0:03:00	40.7366	-73.9906	B02512
1	8/1/2014 0:09:00	40.7260	-73.9918	B02512
2	8/1/2014 0:12:00	40.7209	-74.0507	B02512
3	8/1/2014 0:12:00	40.7387	-73.9856	B02512
4	8/1/2014 0:12:00	40.7323	-74.0077	B02512

Fig 5.2: Loading the data



The screenshot shows the third cell of the Jupyter Notebook being executed, converting the 'Date/Time' column to a datetime format and displaying the first five rows of the resulting DataFrame:

```
In [4]: uber_data['Date/Time'] = uber_data['Date/Time'].map(pd.to_datetime)

In [5]: uber_data.head()

Out[5]:
```

	Date/Time	Lat	Lon	Base
0	2014-08-01 00:03:00	40.7366	-73.9906	B02512
1	2014-08-01 00:09:00	40.7260	-73.9918	B02512
2	2014-08-01 00:12:00	40.7209	-74.0507	B02512
3	2014-08-01 00:12:00	40.7387	-73.9856	B02512
4	2014-08-01 00:12:00	40.7323	-74.0077	B02512

The output also shows the result of the `info()` method:

```
In [6]: uber_data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 829275 entries, 0 to 829274
Data columns (total 4 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   Date/Time    829275 non-null  datetime64[ns]
1   Lat          829275 non-null  float64
2   Lon          829275 non-null  float64
3   Base         829275 non-null  object
dtypes: datetime64[ns](1), float64(2), object(1)
memory usage: 25.3+ MB
```

Fig 5.3: Converting date column to date time

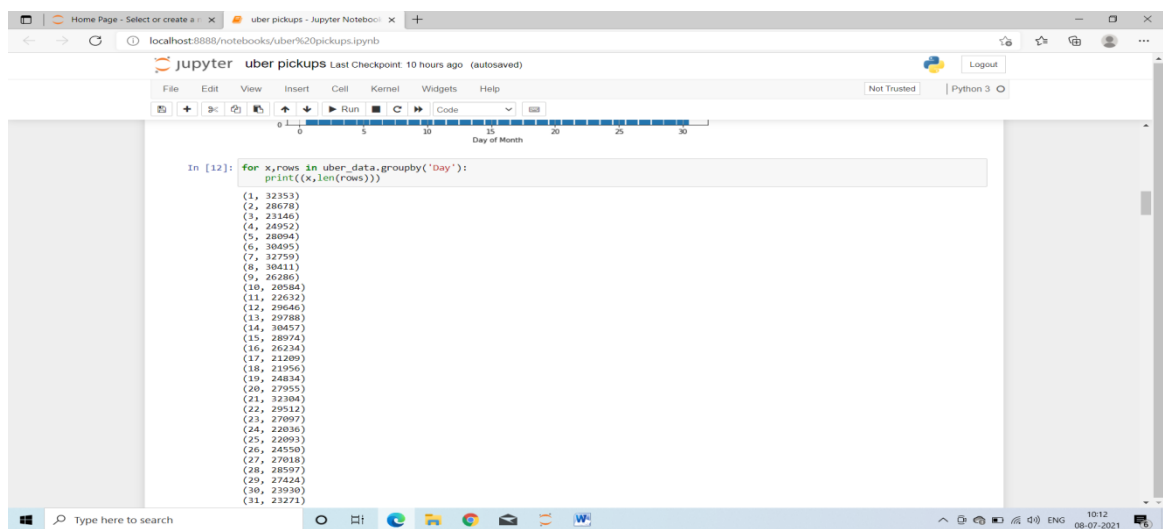
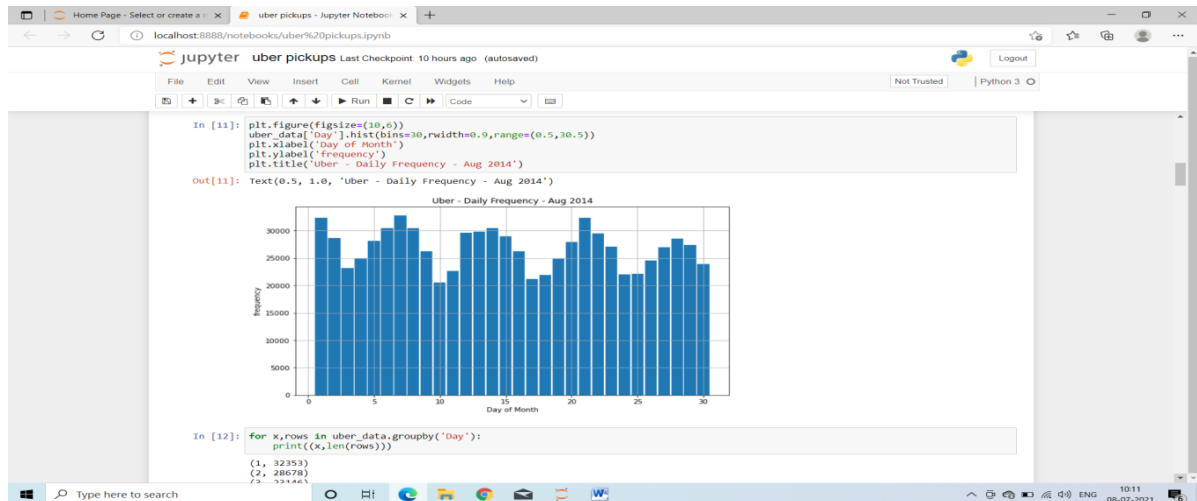
```

In [7]: uber_data['Day'] = uber_data['Date/Time'].apply(lambda x: x.day)
In [8]: uber_data['WeekDay'] = uber_data['Date/Time'].apply(lambda x: x.weekday())
In [9]: uber_data['hour'] = uber_data['Date/Time'].apply(lambda x: x.hour)
In [10]: uber_data.tail()
Out[10]:

```

	Date/Time	Lat	Lon	Base	Day	WeekDay	hour
829270	2014-08-31 23:55:00	40.7552	-73.9753	B02764	31	6	23
829271	2014-08-31 23:55:00	40.7552	-73.9753	B02764	31	6	23
829272	2014-08-31 23:55:00	40.7617	-73.9788	B02764	31	6	23
829273	2014-08-31 23:59:00	40.7395	-73.9889	B02764	31	6	23
829274	2014-08-31 23:59:00	40.7270	-73.9802	B02764	31	6	23

Fig 5.3: Day of month and create a column out of it



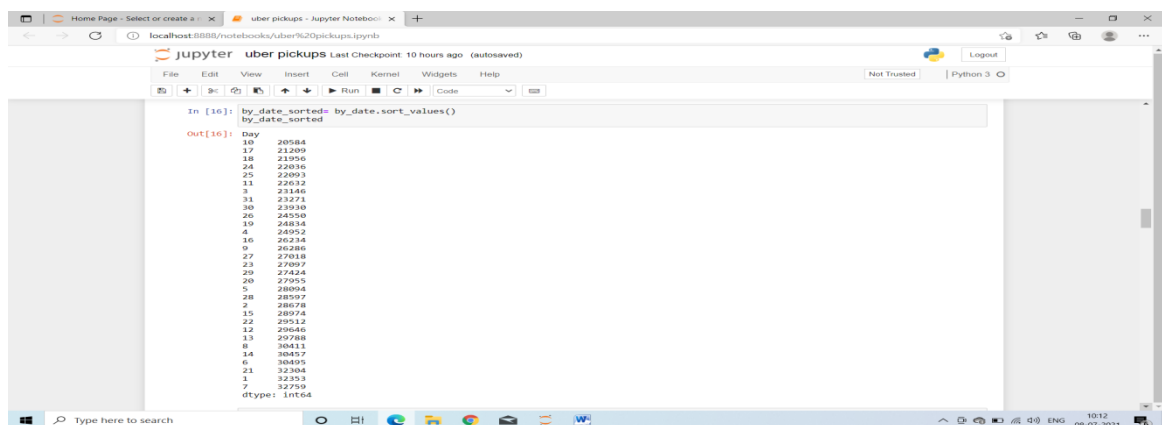
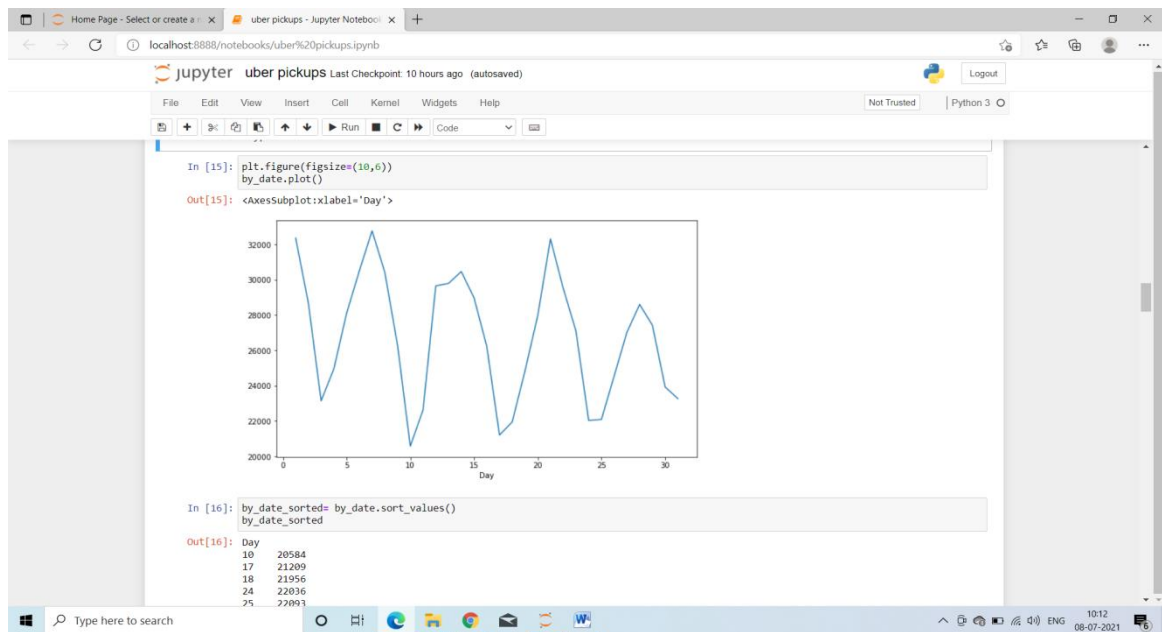
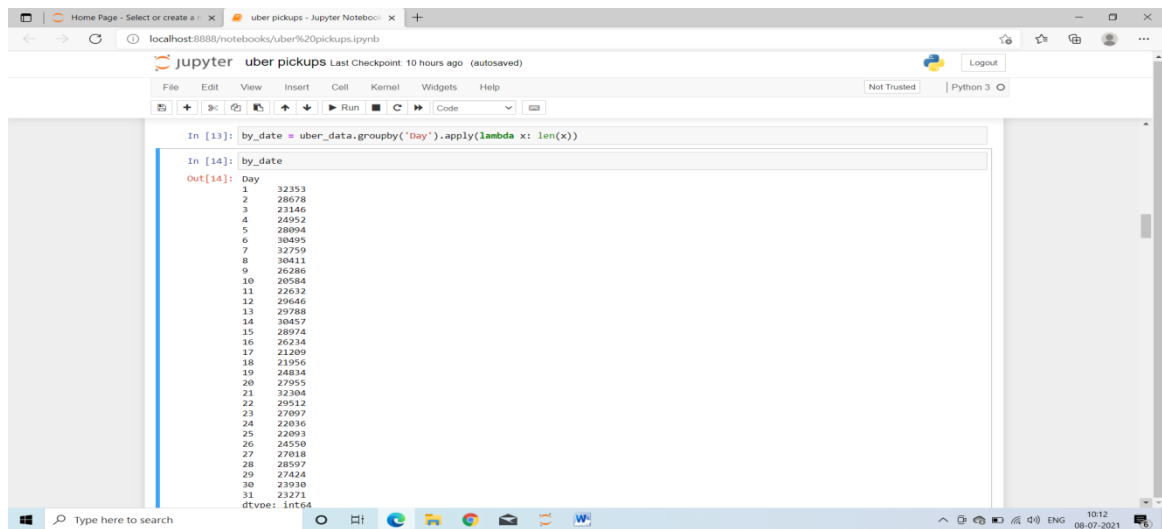
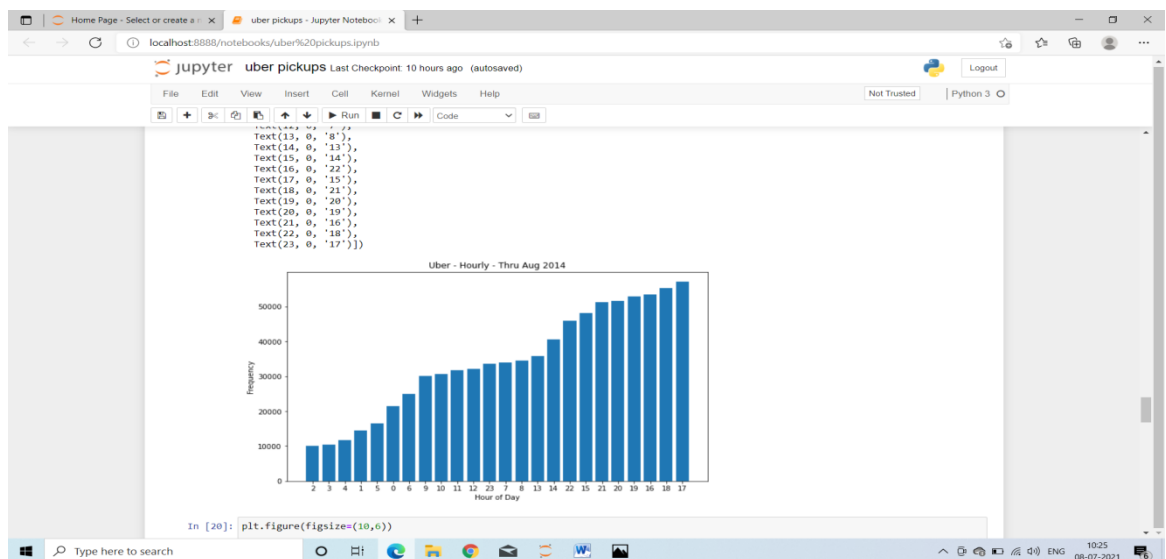
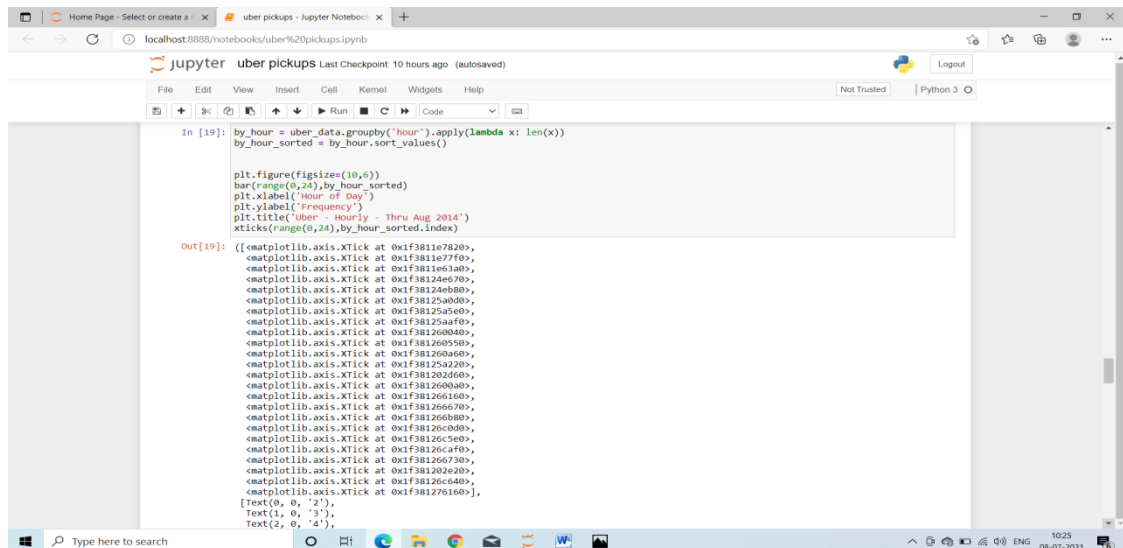
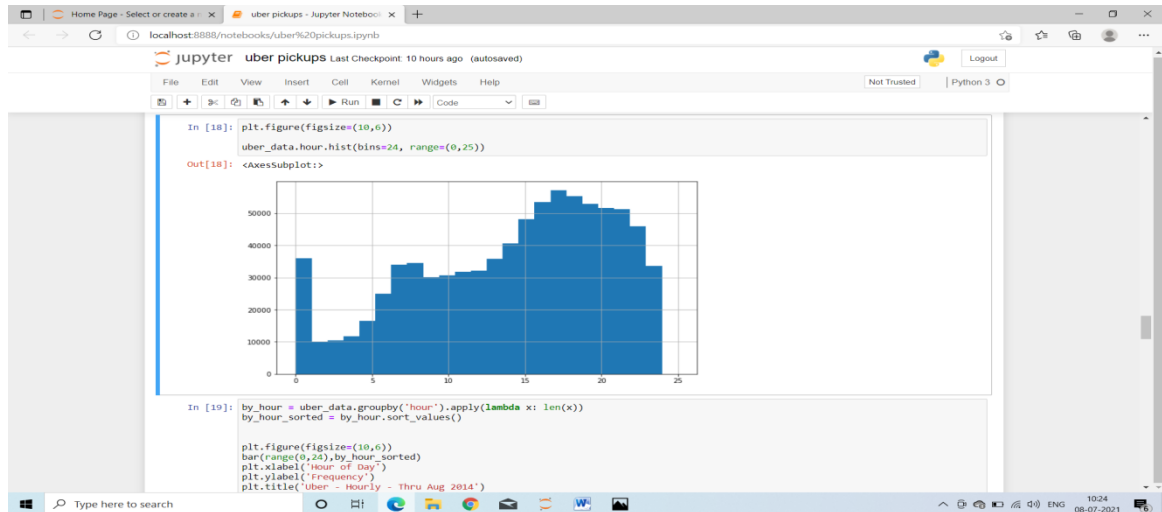


Fig 5.4: Analysis of day month



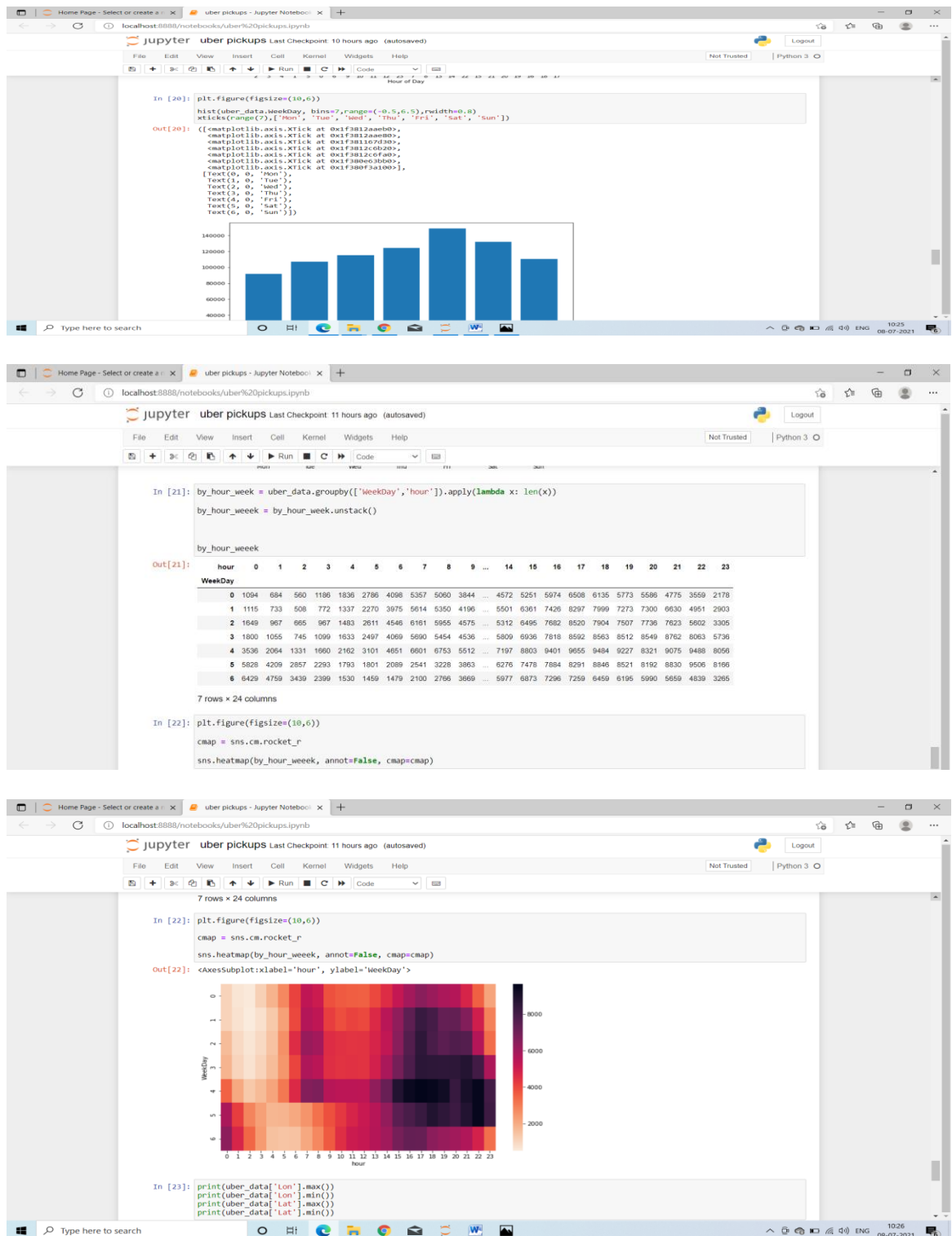


Fig 5.5: Analysis of weekday

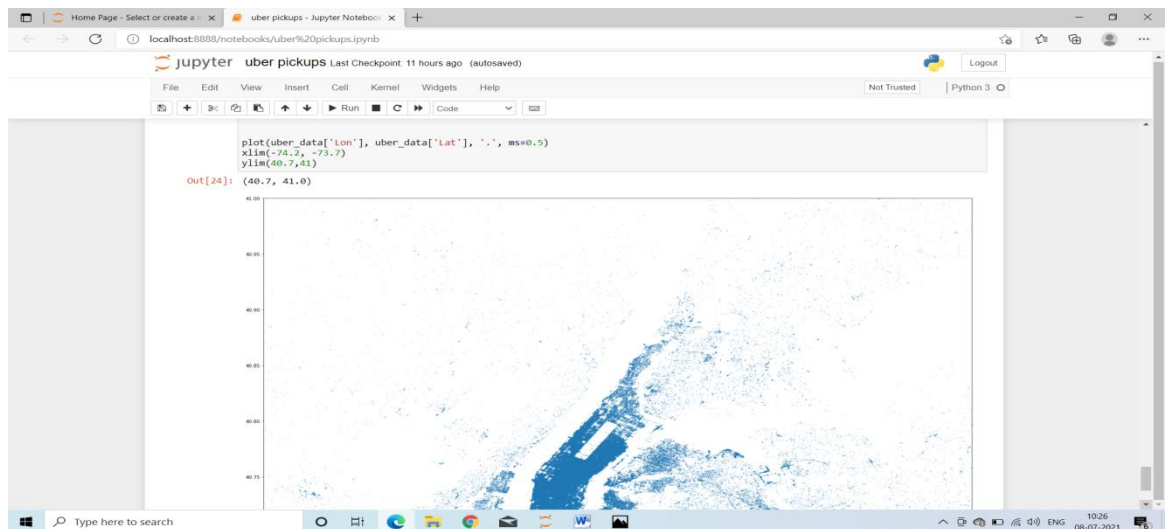


Fig 5.6: Output Graph

CHAPTER 6

CONCLUSION

I have understood the deep learning based method to predict the number of Uber pickups. A long short-term memory (LSTM) model is proposed aiming to capture the long-term dependencies of the pickup sequence over time.

For future work, it would be interesting to compare the Uber pickup data with New York taxi pickup data. Fivethirtyeight shows some statistic temporal and spatial difference of Uber and taxi pickups. More advanced data mining techniques like tensor mining may help to discover deeper knowledge that may help to improve the Uber and/or taxi service system. On the other hand, with an accurate prediction of the pickups, we are able to design a Uber dispatching system and guide the Uber cabs to the area with high possibilities of pick demand. The efficiency improvement of the dispatching system would be a good topic. We also need to model the interaction between taxi usage and the use of all other alternative transportation modes such as, like bike sharing and other sharing service like Lyft that might also have an impact on taxi usage. However, obtaining micro-level data for Lyft and other ride-sharing services remains to be a major challenge for studies such as this one. These new companies are highly encouraged to provide more data to researchers to enable transportation community to plan better for the future. Considerations like if Uber is being used for first-mile/last-mile problems as well as are pick-ups/drop-offs clustered near subway stations in the outer boroughs will be the focus of future research.

REFERENCES

1. Min, W. and L. Wynter, Real-time road traffic prediction with spatio-temporal correlations. Transportation Research Part C: Emerging Technologies, Vol. 19, No. 4, 2011, pp. 606–616.
<https://www.sciencedirect.com/science/article/abs/pii/S0968090X10001592>
2. Lv, Y., Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, Traffic flow prediction with big data: a deep learning approach. IEEE Transactions on Intelligent Transportation Systems, Vol. 16, No. 2, 62015, pp. 865–873.
<https://ieeexplore.ieee.org/document/6894591>
3. Dataset download from kaggle
4. Chen, C., D. Zhang, Z.-H. Zhou, N. Li, T. Atmaca, and S. Li, B-Planner: Night bus route planning using large-scale taxi GPS traces. In Pervasive Computing and Communications (PerCom), 2013 IEEE International Conference on, IEEE, 2013, pp. 225–233.
<https://ieeexplore.ieee.org/document/6526736>