Bike Share Prediction

Last date of revision: 02/06/2021

Khushali Shah

Amit K Gupta

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date Issued | Version | Description | Author |
| 18th May 2020 | 1.1 | First Draft | Amit K Gupta |
| 20th May 2020 | 1.2 | Added Workflow chart | Amit K Gupta |
| 20th May 2020 | 1.3 | Added Exception Scenarios Overall, Constraints | Vrunda Patel |
| 21st May 2021 | 1.4 | Added KPIs | Sukritha Joshi |
| 26th May 2021 | 1.5 | Added user I/O flowchart | Amit K Gupta |
| 26th May 2021 | 1.6 | Added EHR, LSTM model diagrams | Nagesh |
| 31st May 2021 | 1.7 | Added dataset overview and updated user I/O flowchart. | Amit K Gupta |
| 02nd June 2021 | 1.8 | Restructure and reformat LLD | Khushali Shah |
|  |  |  |  |

**Contents**

[Document Version Control 2](#_gjdgxs)

[**Abstract** 4](#_30j0zll)

[1](#_1fob9te) Introduction 5

[1.1](#_3znysh7) Why this Low-Level Design Document? 5

[1.2](#_2et92p0) Scope 6

[1.3](#_tyjcwt) Constraints 6

[1.4](#_3dy6vkm) 1.3 Risks 6

[1.5](#_1t3h5sf) 1.4 Out of Scope 7

[2](#_4d34og8) Technical specifications 7

[2.1 Predicting Disease 7](#_2s8eyo1)

[2.2 Logging 7](#_17dp8vu)

[2.3 Database 7](#_3rdcrjn)

[3](#_26in1rg) Technology stack 8

[4](#_lnxbz9) Proposed Solution 9

[5](#_35nkun2) Model training/validation workflow 10

[6](#_1ksv4uv) User I/O workflow 12

[7](#_44sinio) Exceptional scenarios 13

**Abstract**

Bike sharing systems are a new generation of traditional bike rentals where the whole

process from membership, rental and return back has become automatic. Through these

systems, users are able to easily rent a bike from a particular position and return back at

another position. Currently, there are about over 500 bike-sharing programs around the

world which is composed of over 500 thousand bicycles. Today, there exists great interest in

these systems due to their important role in traffic, environmental and health issues. Apart

from interesting real-world applications of bike sharing systems, the characteristics of data

being generated by these systems make them attractive for the research.

The goal here is to build an end-to-end regression task. Here the user will provide the data

and the result will be given by the best performing hyper tuned Machine Learning model.

The user will also get privileges to choose the deployment options.

.

# Introduction

## Why this Low-Level Design Document?

The purpose of this document is to present a detailed description of the Bike Share Prediction System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate. This document is intended for both the stakeholders and the developers of the system and will be proposed to the higher management for its approval.

The main objective of the project is to predict the total number of bikes in demand for a particular day based on given data. The dataset used for developing the model is Bike Share dataset.

Bike Share dataset can:

* Date.
* Weather and Season details.
* Importance of a day like (Holiday, Working day, Weekend )

Weather details contain different weather situations information, such as:

* Clear, Few clouds, Partly cloudy, Partly cloudy
* Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
* Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
* Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
* Temperature
* Humidity
* Feeling Temperature

This project shall be delivered in two phases:

Phase 1: All the functionalities with PyPi packages.

Phase2: Integration of UI to all the functionalities.

## Scope

This software system will be a Web application This system will be designed to find the demand for the bikes at earliest for better system management, improved service, and more efficient resource allocation using previous Bike Share records available. More specifically, Early understanding of user demand is important for better service management. This system is designed to predict the bikes demand for users from weather details, season information , details of a day etc.

## Constraints

We will only be selecting a few of the bike sharing systems data.

## Risks

Document specific risks that have been identified or that should be considered.

## Out of Scope

Delineate specific activities, capabilities, and items that are out of scope for the project.

# Technical specifications

## 2.1 Dataset

|  |  |
| --- | --- |
| Bike Sharing Dataset | https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset |

## 2.1.1 Bike Sharing dataset overview

This dataset contains the hourly and daily count of rental bikes between years 2020 and 2021 in Capital bikeshare system with the corresponding weather and seasonal information.

## 2.1.2 Input schema

 instant: record index  
- dteday : date  
- season : season (1:winter, 2:spring, 3:summer, 4:fall)  
- yr : year (0: 2011, 1:2012)  
- mnth : month ( 1 to 12)  
- hr : hour (0 to 23)  
- holiday : weather day is holiday or not (extracted from []](http://dchr.dc.gov/page/holiday-schedule))  
- weekday : day of the week  
- workingday : if day is neither weekend nor holiday is 1, otherwise is 0.  
+ weathersit :  
- 1: Clear, Few clouds, Partly cloudy, Partly cloudy  
- 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist  
- 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds  
- 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog  
- temp : Normalized temperature in Celsius. The values are derived via (t-t\_min)/(t\_max-t\_min), t\_min=-8, t\_max=+39 (only in hourly scale)  
- atemp: Normalized feeling temperature in Celsius. The values are derived via (t-t\_min)/(t\_max-t\_min), t\_min=-16, t\_max=+50 (only in hourly scale)  
- hum: Normalized humidity. The values are divided to 100 (max)  
- windspeed: Normalized wind speed. The values are divided to 67 (max)  
- casual: count of casual users  
- registered: count of registered users  
- cnt: count of total rental bikes including both casual and registered

## 2.2 Predicting Number of bikes

* The system displays the choices of the Season.
* The User chooses the date from the calendar.
* The User selects the Weather situation.
* The user gives temperature, feeling temperature, Humidity.
* The system should be able to predict number of bikes needed based on the given information.

## 2.3 Logging

We should be able to log every activity done by the user.

* The System identifies at what step logging required
* The System should be able to log each and every system flow.
* Developers can choose logging methods. You can choose database logging/ File logging as well.
* System should not be hung even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

## Database

System needs to store every request into the database and we need to store it in such a way that it is easy to retrain the model as well.

The system stores each and every data given by the user or received on request to the database. Database you can use is Cassandra.

**2.5 Deployment**

1. AWS



# Technology stack

|  |  |
| --- | --- |
| **Front End** | HTML/CSS |
| **Backend** | Python Flask |
| **Database** | Cassandra |
| **Deployment** | AWS |

# Proposed Solution

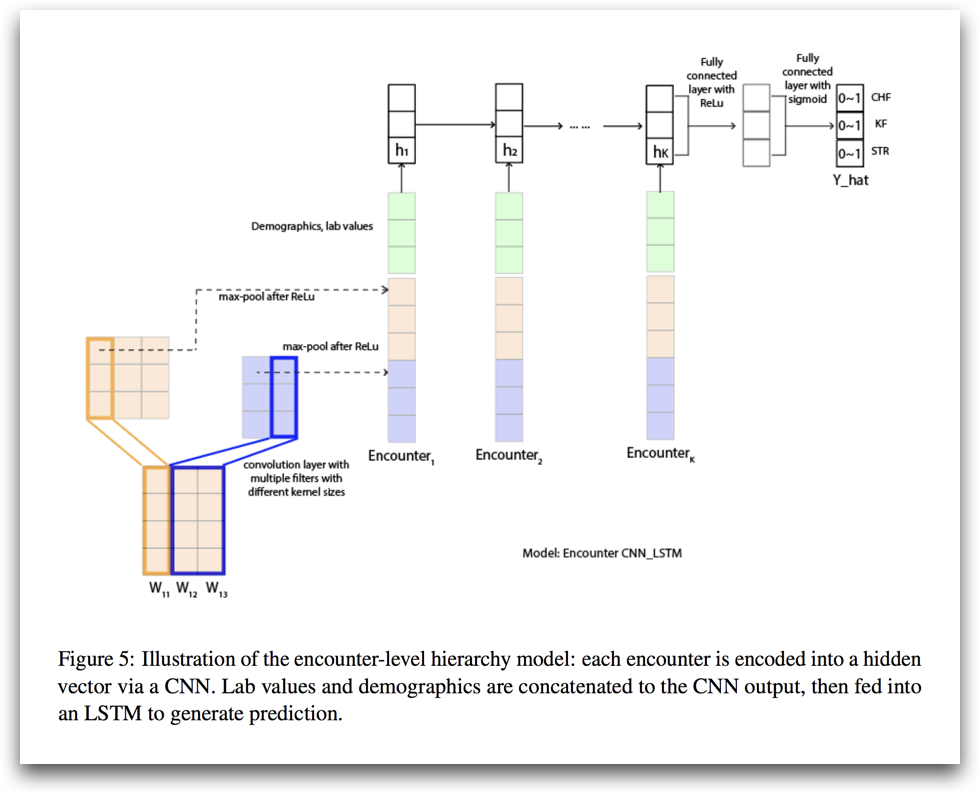
refer: <https://arxiv.org/abs/1808.04928>

Based on the actual research paper, if we are using history of the patient to predict the future then we might want to consider using LSTM. However, drawing a baseline in the form of some Machine Learning algorithm would be helpful. Why making a baseline model important? Well, to compare the performance of our actual model, let say LSTM in this case, is very important to ascertain that we are in the right direction as if performance of LSTM is not better than the baseline model then there is no point of using LSTM.

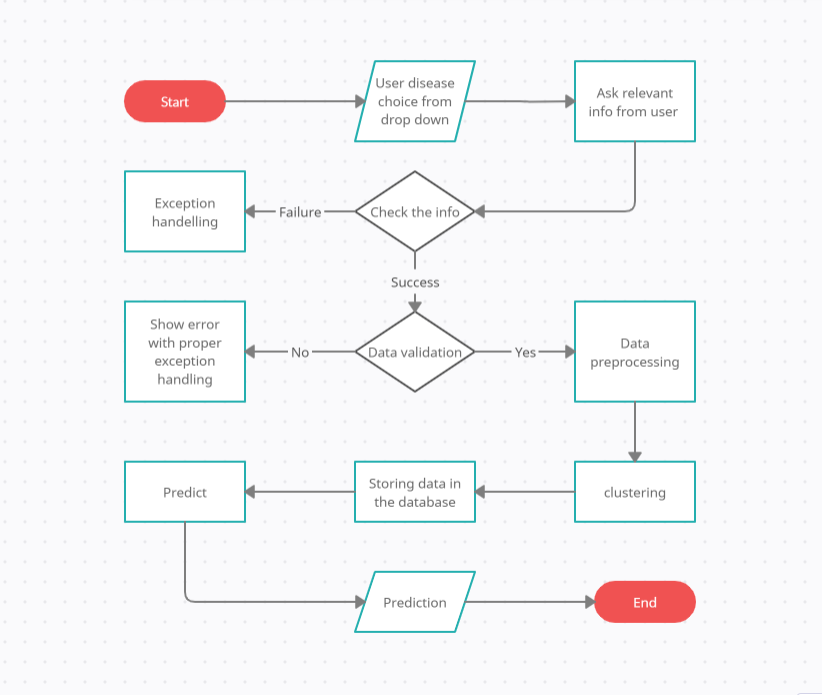
1. Baseline Model: Logistic Regression, since this is a classification problem.
2. Actual model: LSTMs.

# Model training/validation workflow



****

# User I/O workflow

****

# Exceptional scenarios

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Exception | Mitigation | Module |
| 18th May 2020 | 1.1 | First Draft | Amit K Gupta |
| 20th May 2020 | 1.2 | Added Workflow chart | Amit K Gupta |

# Test cases

|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Steps to perform test case | Module | Pass/Fail |
|  |  |  |  |

# Key performance indicators (KPI)

* Time and workload reduction using the EHR model.
* Comparison of accuracy of model prediction and doctor’s prediction.
* Number of times a patient visits the hospital.
* Time between symptom onset and detection of illness/visit to hospital.
* Immunity of patient (based on previous illnesses).
* Vaccines the patient has taken.
* Length of stays in hospital.