

HARDIKKUMAR SUTARIYA

High Level Design (HLD)

Metro Interstate Traffic Prediction



Contents

Document Version Control	2
Abstract.....	4
1. Introduction	5
1.1 Why this High-Level Design Document?	5
1.2 Scope... ..	5
2. General Description.....	6
2.1 Product Perspective	6
2.2 Problem Statement.....	6
2.3 Proposed Solution	6
2.4 Further Improvements	6
2.5 Technical Requirements	6
2.2 Data Requirements.....	6
2.6 Tools Used	7
2.7 Constraints	8
2.2 Assumptions.....	8
3. Design Detail.....	9
3.1 Process Flow	9
3.1.1 Model Training and Evaluation.....	9
3.1.2 Deployment Process.....	10
3.2 Event Log	10
3.3 Error Handling	10
4. Performance	10
4.1 Reusability.....	11
4.2 Application Compatibility.....	11
4.3 Resource Utilization.....	11
4.4 Deployment	11
5. Conclusions	12

Abstract

As the world progresses with advanced technologies and comforting appliances, population of the world also is becoming a major problem for advancement of human beings itself. The more the population, more is the number of vehicles in the world. Some people cannot afford even a cycle, some own a vehicle while some have a collection of vehicles. No matter how necessary vehicles are in the present, nearly every family has a car or motorbike for transportation. As a result, traffic related issues are increasing day by day and thereby directly contributing to noise pollution, air pollution and time wastage. Not only vehicle population but different factors such as time, weather conditions and occasions also have a direct effect on traffic volume.

Thus, building technologies to get insight of traffic behaviors and finding out the factors causing traffic problems becomes an important solution to focus on. Model that could predict the traffic volumes based on collective information of different factors causing traffic problems is a perfect solution of dealing with the increasing traffic problems.

1 Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions before coding and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:
 - Security
 - Reliability
 - Maintainability
 - Portability
 - Reusability
 - Application compatibility
 - Resource utilization
 - Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

2 General Description

2.1 Product Perspective

The goal of this project is to build a model that can predict the traffic volume depending on the conditions or factors that causes traffic problems. For, this We have used ensemble learning algorithms to develop the prediction model.

2.2 Problem Statement

To create traffic prediction model for using machine learning algorithms which will help us to predict and analyze the traffic volume based on different conditions. This project aims to implement the following use cases:

- To predict traffic volume at different times of the day
- To predict traffic volume at different weather conditions.

2.3 Proposed Solution

The solution proposed here is a prediction model which will help us to find out the traffic volume based on different factors such as time, weather conditions and holidays. This entire project is build using DVC (Data Version Control) and MLFlow for retraining, experimenting and model monitoring purpose.

2.4 Further Improvements

With continuous retraining and model configurations, model can be improved continuously. In future, other features can be added to improvise our prediction model.

2.5 Data Information

The dataset used in this project is a UCI machine learning dataset. The data at every transformation is uploaded and extracted from mongoDB database. The dataset contains hourly interstate 94 westbound traffic volume for MN DoT ATR station 301. The region of data lies between regions of Minneapolis. It includes features such as holiday, time, weather, etc. which impacts the traffic volume traffic volume directly.

Information of attributes of dataset as follows:

- holiday: Indicates if the date is a holiday and if it specifies the holiday, if not None.
- temp: Indicates the temperature in Kelvin.
- rain_1h: Amount in mm of rain that occurred in the hour.
- snow_1h: Amount in mm of snow that occurred in the hour.
- clouds_all: Percentage of cloud cover.
- weather_main: Short textual description of the current weather.
- weather_description: Longer textual description of the current weather.
- date_time: Hour of the data collected in local CST time.
- traffic_volume: Hourly I-94 ATR 301 reported westbound traffic volume.

2.6 Tools Used

The whole project is built upon Python Programming language. Visual Studio code and jupyter notebooks are used as a prime development interface. Python libraries such as NumPy, Pandas, Matplotlib, DVC, MLflow, etc. are majorly used in the project.



- Visual Studio Code is used as IDE.
- Python is used as the main programming language.
- HTML is used for website design.
- Git is used for continuous integration and version control.
- DVC is used for Data Version Controls
- MLflow is a platform to streamline machine learning development, including tracking experiments, packaging code into reproducible runs, and sharing and deploying models.
- Jupyter is used for analysis notebooks and for testing codes.
- Sklearn is used for performing machine learning operations.
- Python libraries such as numpy, pandas, matplotlib, seaborn, etc. are also used for required functions.
- Railways and Heroku are the deployment platforms used for this project.

2.7 Constraints

The machine learning based prediction system must be user friendly, as automated as possible and users should not be required to know any of the workings.

2.8 Assumptions

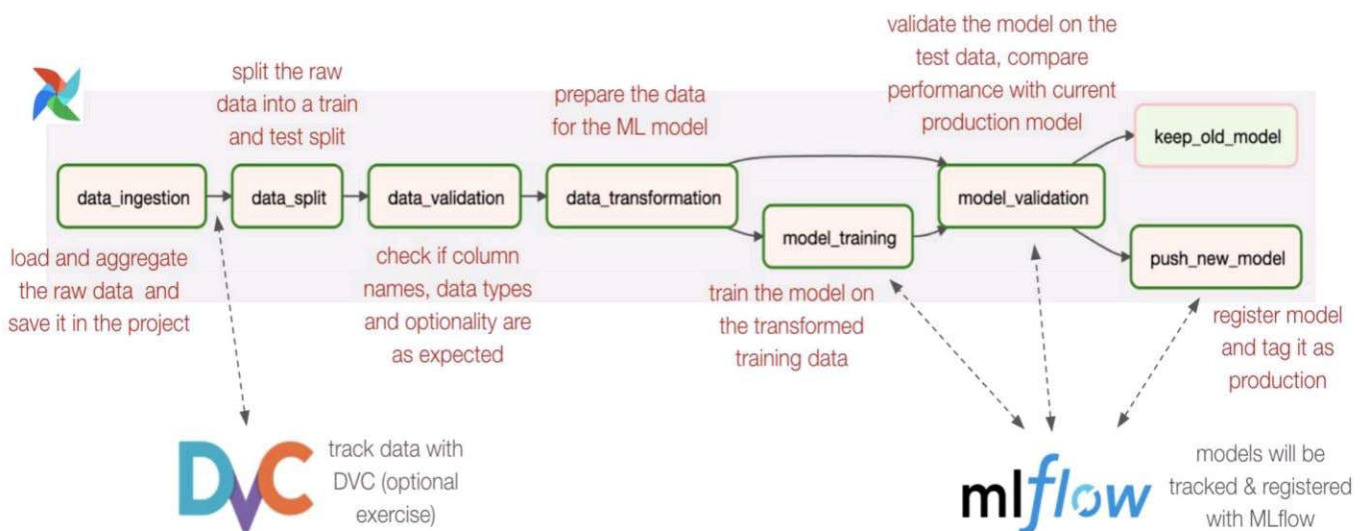
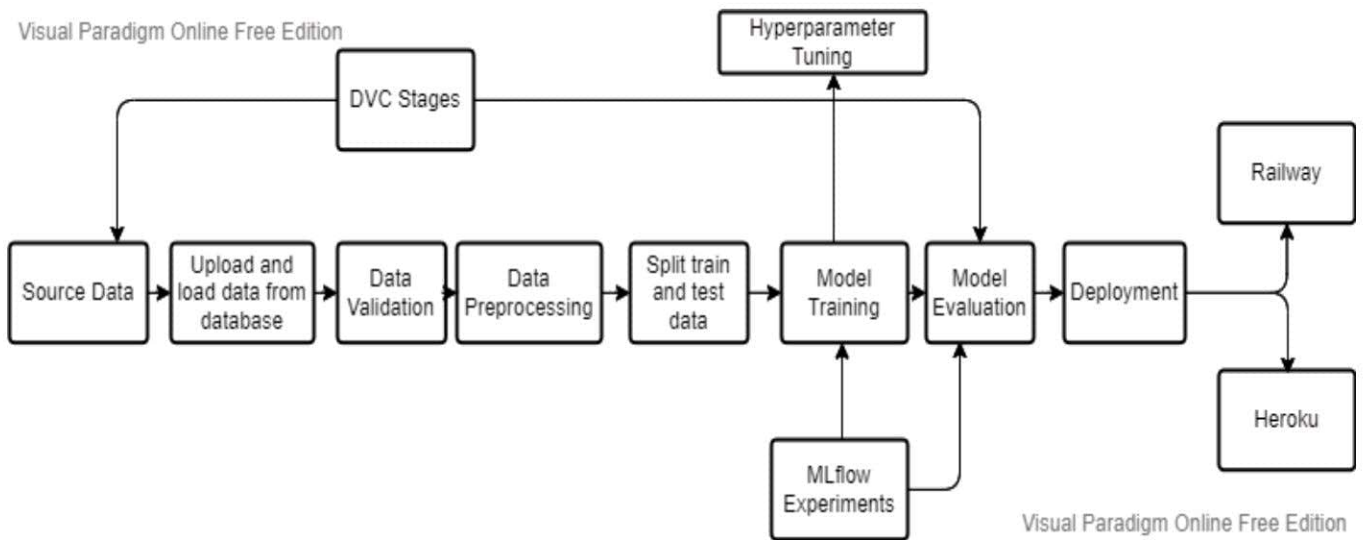
The main objective of this project is to build a predictive model which predicts the traffic Volume on untrained data. XGBoost Regression – an ensemble learning approach is used for model training. It is assumed that the project has the ability to predict accurate results when appropriate requirements are satisfied. Model retraining can be performed with the help of mlflow.

3 Design Details

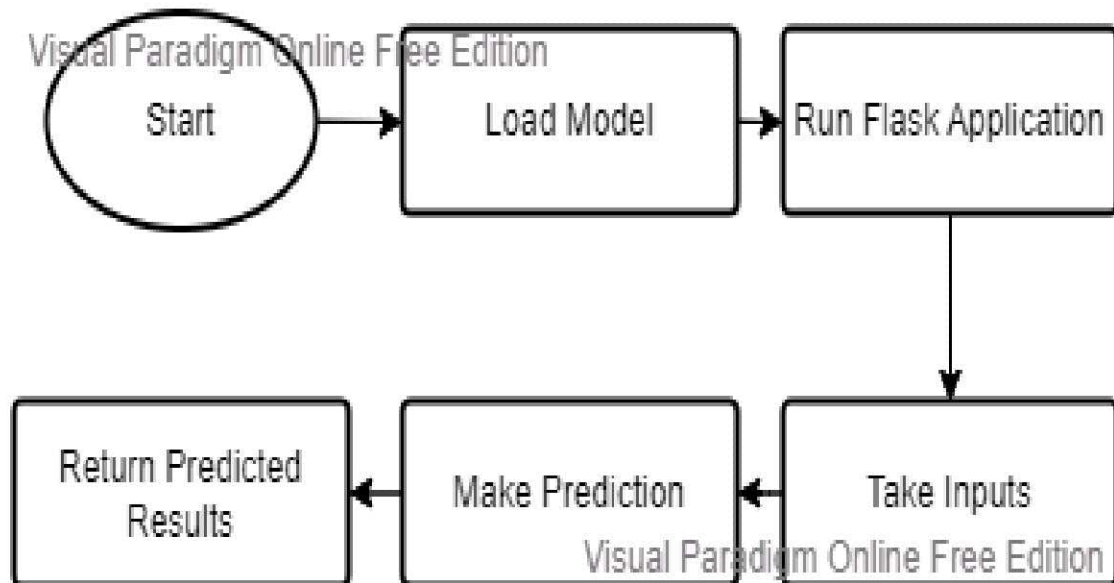
3.1 Process Flow

For prediction model, we will be using machine learning models. Below is the process flow diagram of the project.

3.1.1 Proposed Methodology



3.1.2 Deployment



3.2 Event log

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description:

The System identifies at what step logging required

The System should be able to log each and every system flow.

Developer can choose logging method. You can choose database logging/ File logging as well.

System should not hang even after using so many loggings.

Logging just because we can easily debug issues so logging is mandatory to do.

3.3 Error Handling

Should errors be encountered, an explanation will be displayed as to what went wrong?

An error will be defined as anything that falls outside the normal and intended usage.

4 Performance

The Metro Interstate Traffic Prediction is a prediction model which predicts the traffic Volume based on the weather condition, time and occasions. The model therefore, is expected to be accurate as overfitting or an underfitting model can lead to unexpected results.

4.1 Error Handling

The code written and the components used should have the ability to be reused with no problems. Custom Exception class is used to handle errors.

4.2 Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

4.3 Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

4.4 Deployment



Railway



 **render**

- **Railway** - Railway is an infrastructure platform where you can provision infrastructure, develop with that infrastructure locally, and then deploy to the cloud.
- **Heroku** - Heroku is a platform as a service (PaaS) that enables developers to build, run, and operate applications entirely in the cloud.
- **Render** - Render is a unified cloud to build and run all your apps and websites with free TLS certificates, global CDN, private networks and auto deploys from Git.

Metro Interstate Traffic Prediction
Home
source code

Enter the details as indicated:

Prediction:

Weather Category:

Clear : 0
Clouds : 1
Drizzle : 2
Fog : 3
Haze : 4
Mist : 5
Rain : 6
Smoke : 7
Snow : 8
Squall : 9
Thunderstorm : 10

Metro Interstate Traffic Prediction [Home](#) [source code](#)

Enter the details as indicated:

Prediction:

Weather Category:

Clear : 0
Clouds : 1
Drizzle : 2
Fog : 3
Haze : 4
Mist : 5
Rain : 6
Smoke : 7
Snow : 8
Squall : 9
Thunderstorm : 10

Metro Interstate Traffic Prediction [Home](#) [source code](#)

Enter the details as indicated:

Prediction:

3465.243408203125

Weather Category:

Clear : 0
Clouds : 1
Drizzle : 2
Fog : 3
Haze : 4
Mist : 5
Rain : 6
Smoke : 7
Snow : 8
Squall : 9
Thunderstorm : 10

5 Conclusion

The aim of this project is to build a prediction model which will help us to find out the traffic volume based on different factors such as time, weather conditions and holidays. With continuous improvisations the project can upgrade its accuracy and parameters and hence can give insights to real world traffic problems.