

HIGH LEVEL DESIGN

(HLD)

STORE SALES

PREDICTION

Revision Number: 1.0

Last date of revision: 01/06/2021

Manu Vats

DOCUMENT VERSION CONTROL

Date Issued	Version	Description	Author
8/06/2021	1	Initial HLD – V1.0	Manu Vats
23/06/2021	2	Updated KPI – V1.1	Manu Vats

ABSTRACT

Nowadays, shopping malls and Big Marts keep track of individual item sales data in order to forecast future client demand and adjust inventory management. In a data warehouse, these data stores hold a significant amount of consumer information and particular item details. By mining the data store from the data warehouse, more anomalies and common patterns can be discovered. With the help of these patterns and anomalies, the store management can identify the possible future scenarios and allocate the resources in a better way.

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 prior to 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.

1.1 Definitions

<i>Term</i>	<i>Description</i>
<i>EDA</i>	Exploratory Data Analysis
<i>Database</i>	Collection of all the records in the dataset
<i>IDE</i>	Integrated Development Environment
<i>AWS</i>	Amazon Web Services

2 GENERAL DESCRIPTION

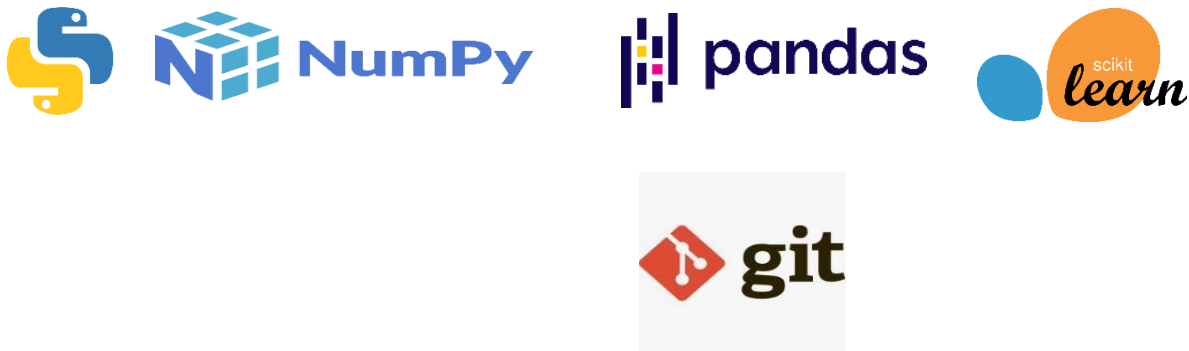
2.1 Product Perspective & Problem Statement

With the retail companies being able to record various features of sales data, the need to analyse that data and predict future outcomes has become more and more imperative. The store needs to predict the future sales figure so that they allocate resources, manage inventory and optimize marketing campaigns more efficiently.

The objective of the project is to perform machine learning models to predict future sales of the store. This project aims to apply Python as coding language to create machine learning models.

2.2 Tools used

Python programming language and frameworks such as NumPy, Pandas and Scikit-learn, are used to build the whole model.



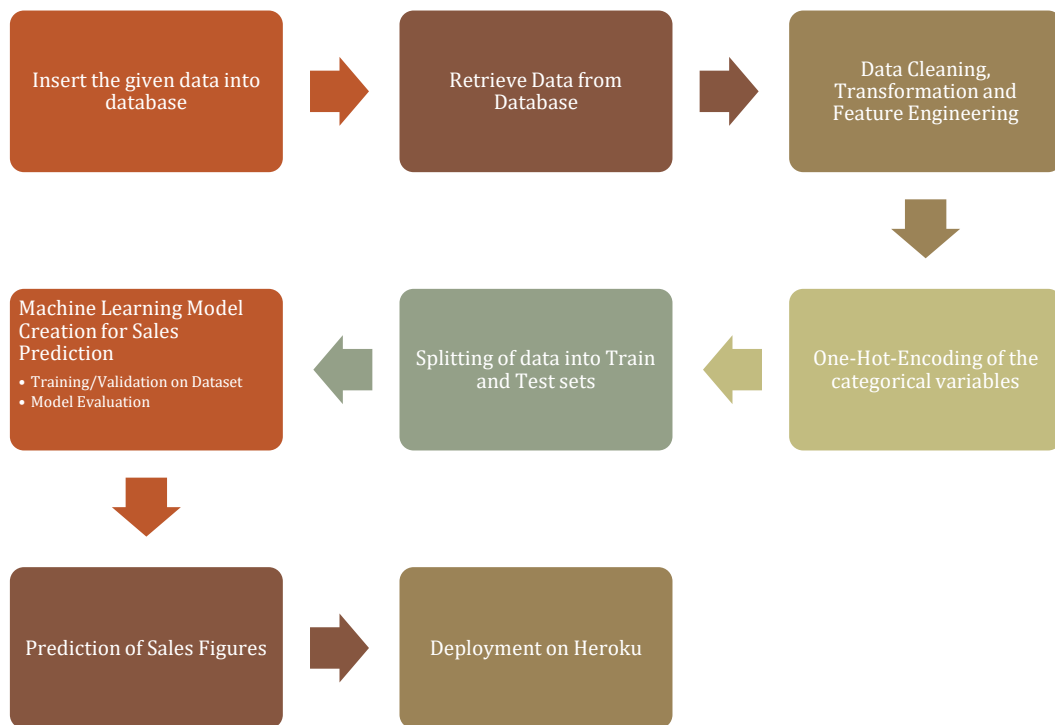
- Visual Studio Code is used as IDE.
- For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
- Heroku is used for deployment of the model.
- Cassandra is used to store, retrieve, insert, delete, and update the database.
- Front end development is done using HTML/CSS
- Python is used for backend development.
- GitHub is used as version control system.

2 DESIGN DETAILS

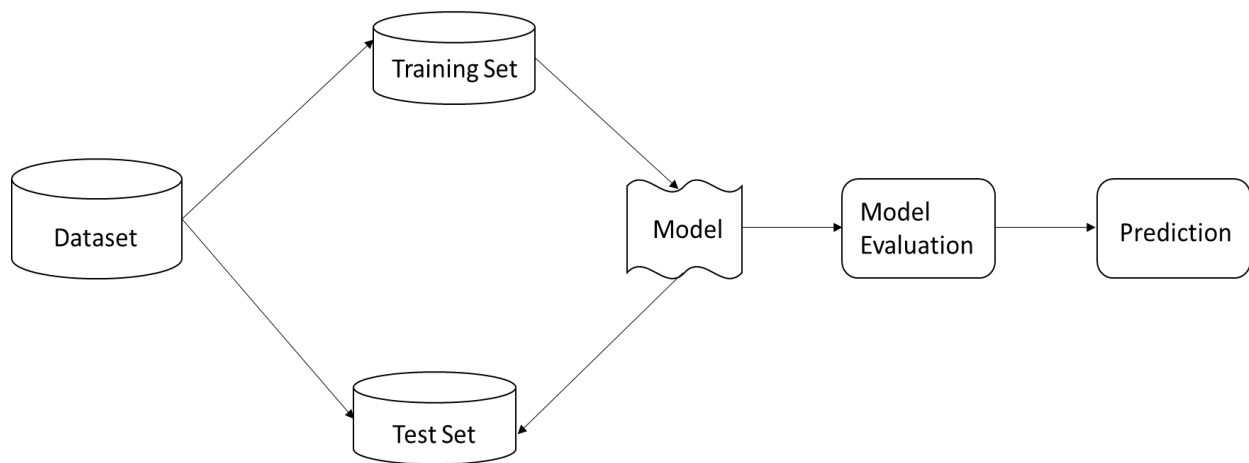
2.1 Process Flow

For identifying the different types of anomalies, we will use a deep learning base model. Below is the process flow diagram is as shown below.

Proposed methodology



3.1.1 MODEL TRAINING AND EVALUATION



2.2 Event log

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

Initial Step-By-Step Description:

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

2.3 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.

Performance

The Sore Sales Prediction solution is used for prediction of sales in the store. Based on this solution, the store will plan their inventory, marketing campaigns and other store operations. The data obtained from this solution will be very crucial for efficient functioning of the store. Bad performing model can incur huge losses for the store so it should be as accurate as possible and not mislead the managers.

2.4 Reusability

The code written and the components used should have the ability to be reused with no problems.

2.5 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.

2.6 Resource Utilization

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

2.7 Deployment

The application will be deployed on Heroku Cloud.



3 CONCLUSION

The Store Sales Prediction algorithm will be used to predict the sales in the store to help the managers plan and manage the store operations.

4 REFERENCES

1. https://en.wikipedia.org/wiki/Unmanned_ground_vehicle
2. Google.com for images of UGV hardware.

<https://www.ros.org/>