Low Level Design (LLD)

FLIGHT FARE PREDICTION

Created on – 09/08/2023

Swapnil Ritundmokashi

Contents

Abstract

Introduction

Importance of Low-Level Design (LLD)?

1. Architecture
2. Architecture Design
   1. Data Collection
   2. Tools Used
   3. Data Description
   4. Importing data into databases
   5. Exporting data from the databases
   6. Data pre-processing
   7. Modelling
   8. UI integration
   9. Data from user
   10. Data validation
   11. Rendering the results
3. Deployment

Abstract

For years, the introduction and growth of platforms to be used by end customers and passengers has been based on transparency about cost, routs, global alternatives, and required travel time. This was a prerequisite and starting point to increase in the growth rate of low-cost carriers and for questioning the traditional business model of legacy airlines. The revolution in over in that segment, new business models will come but we find ourselves in the post revolution era.

However, there are good opportunities to take ideas and learnings from industry 4.0 and use them in aircraft production. In particular, the use of tools for customer-specific options could be reduced by software driven solutions and intelligent implementation of laser pointing mechanisms. When it comes to the implementation of industry 3.0 i.e., automation and industry 4.0 that is digitalization the level of maturity is not homogenous throughout the supply chain.

INTRODUCTION

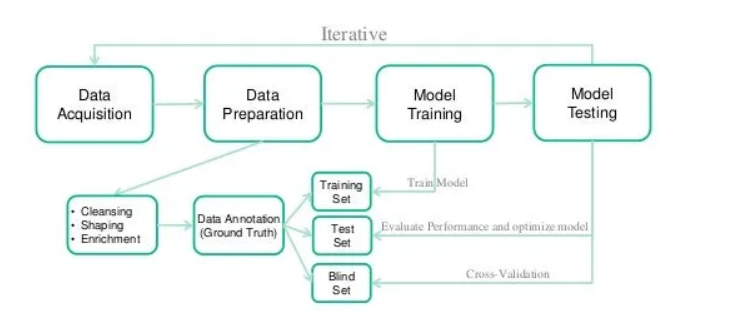
Importance of LLD Documentation?

The main purpose of the low-level documentation is to give end to end information about the project and describe each and every sub step in the complete process. It also gives you the complete idea of the machine learning model which is applied on this project.

Key Points:

* Describes the Design flow
* Implementation
* Software requirements
* Architecture of project

1. ARCHITECTURE



1. ARCHITECTURE DESIGN

This project is completely based on the life cycle of the machine learning, where we will be predicting the price of the flight fare from source to destination. The tools used in this project are python, pandas, NumPy, matplotlib, seaborn, scikit learn, flask for backed, HTML, CSS for front-end and Heroku for deployment.

2.1 Data collection

The data which is used in this project is available open source on the kaggle.com website.

2.2 Tools Used

* Python version 3.8 is used in this project.
* Some python libraries like NumPy, pandas, matplotlib, seaborn and scikit learn are used for implementation of machine learning algorithms.
* Jupyter notebook and Visual studio code is used as IDE for writing the code.
* Cassandra database is used as the database for this project.
* HTML and CSS are used for developing the front end of our web application.
* Flask is used for the backend development.
* Github is used as the version control system.
* Heroku is used for deployment of model.

2.3 Data Description

We have given two different dataset one is training dataset and other one is testing dataset. Training dataset contains more than 10000 rows of data and 10 feature column and 1 target column that is “Price”. Whereas, we have testing dataset as well which contains about 2000+ rows of data.

More description about the dataset:

1. Airline: This is the name of airline.
2. Date\_of\_Journey: This is the date of which journey is to be done.
3. Source: Starting point of the journey.
4. Destination: Ending point of the journey.
5. Route: It contains the stoppage point of the flight.
6. Dep\_Time: Departure time of the flight.
7. Arrival\_Time: Arrival time to the destination of the flight.
8. Duration: Time taken by flight to go from source to destination.
9. Total\_Stops: Total number of stops from source to destination.
10. Additional\_Info: Additional information about the flight if any.
11. Price: This is the final price of the flight when we supply the input parameters.

2.4 Importing data into the databases

* + Step 1: Making connection with the database
  + Step 2: Database creation with the name flight fare
  + Step 3: Cqlsh command is written for creating data table with the required features.
  + Step 4: Finally, Cqlsh is used for uploading the dataset into data table for bulk insertion.
  1. Exporting data to the database

While creating the API which is mentioned in the above step the importing URL is also created in parallel so that it would download the data in the .csv format.

* 1. Data Preprocessing
* Checking the basic profile of the dataset.
* Checking for the null values, there are some null values so in this case we are dropping it because they are very less in numbers.
* There are some features like departure time, arrival time, duration which our machine learning algorithm is unable to understand so we need to extract basic information like hours and minutes which we are storing in new columns.
* Similarly, we must extract the journey date and journey column from date of journey column.
* There are some categorical features like number of stops, source, destination all this feature must be encoded this process is called as one hot encoding.

Now, data is ready to passed to machine learning algorithm.

* 1. Modelling

After the selection of best possible features and one hot encoding now it is the time to select the best possible machine learning algorithm which will give the highest accuracy to our model. We have tried linear regression, decision tree and Random Forest with its performance metrices but the highest accuracy we were getting using Random Forest itself therefore we have selected random forest as the main algorithm which will be used in this project. This model is been fitted using hyperparameter tuning which is done using randomized search CV.

* 1. UI Integration

If you have to make user interact to the model than we have to create the user interface, this user interface is been created using HTML and CSS creatively.

* 1. Data from the user

Data from the user is been retrieved using the HTML & CSS template.

* 1. Data Validation

The data which is entered by the user is been validated by the main.py file which is built using the python flask and then this data is transferred to our model.

* 1. Rendering the result

The result for our model is been rendered at the HTML and CSS page.

1. Deployment

This machine learning model is deployed on AWS which is the cloud platform by amazon.