High Level Design (HLD)

FLIGHT FARE PREDICTION

Revision Number: 1.0

Last Date of Revision: 31/05/2023

Document Version Control

Date Issued	Version	Description	Author
31/05/2023	1.0	Initial HLD	Debasish Bhagawati

Contents

Abstract

There are numerous variables that determine the overall cost of airline tickets, such as the airline, the date of travel, the source and destination, the route, the length of the trip, and so forth. Each service provider appears to have its own particular set of rules and procedures for establishing prices. Recent developments in machine learning (ML) and artificial intelligence (AI) make it possible to derive such principles and model price volatility.

1. Introduction

1.1 What is High Level Document?

The goal of this High-Level Design (HLD) Document is to provide the current project description with the additional depth needed to describe an appropriate model for coding. This paper can be used as a reference guide for how the modules interact at a high level and is also meant to aid in identifying conflicts before coding.

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. DESCRIPTION

2.1 Problem Perspective

Since the privatisation of the airline firm, the pricing of airfares has developed into a sophisticated system of rules, including computer simulations that determine airfare marketing methods. Research has shown that these principles are affected by a variety of events, despite the fact that they are still mostly unknown. While still crucial, traditional factors like proximity no longer determine pricing structure exclusively. Increasingly, sociological, economic, and marketing considerations all have an impact on how much flights cost. The flight fare prediction is a machine learning model that aids in predicting travel costs and aids users in comprehending the cost of their journey.

2.2 Problem Statement

Travelling through flights has become an integral part of today's lifestyle as more and more people are opting for faster travelling options. The flight ticket prices increase or decrease every now and then depending on various factors like timing of the flights, destination, and duration of flights various occasions such as vacations or festive season. Therefore, having some basic idea of the flight fares before planning the trip will surely help many people save money and time.

2.3 Purposed Solution

The main goal is to predict the fares of the flights based on different factors available in the provided dataset.

2.4 Further Improvements

We can even use the features like weekdays or weekend, season, rating for predicting the price.

2.5 Data Requirement

To build the airline ticket pricing model, we first need information on aircraft business. We now have two datasets, one for training and one for testing. We can access the dataset on kaggle within the type of standout sheet(.xlsx).

2.6 Tool Used

- Jupyter Notebook is used for data pre-processing, model building etc.
- PyCharm is used as IDE.
- Streamlit is used for backend development.
- AWS is used for deployment of the model.
- Github is used as version control system.

2.7 Constraints

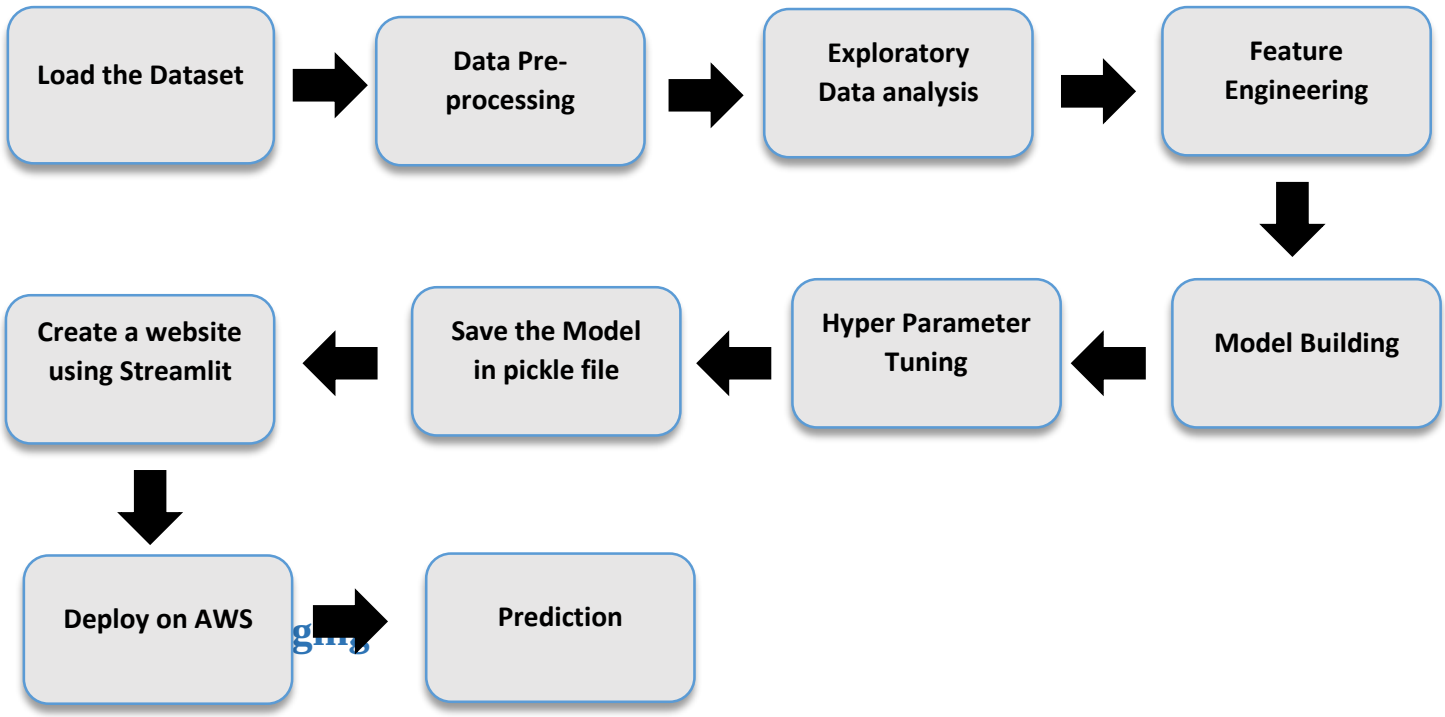
The flight fare prediction answer should be user friendly, as automatic as attainable and also the user should not be needed to understand any of the workings.

2.8 Assumptions

The project's goal is to put the utilisation scenarios into practise for the brand-new dataset that the user gives through the software. A machine learning model is used to process the computer file on top of it. Additionally, it is presumable that all project components will be able to function in the manner that the creator intends.

3. Design Flow

3.1 Model building and Deployment Process



Each step is being logged within the system that runs internally, that shows the date time and therefore the processed that has been performed, work is completed in several layers as information, DEBUG, ERROR, WARNINGS. this provides US the perceive of the logged info.

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 Evaluation

4.1. Reusability

Since some of the written code is familiar to other applications, the remainder is altered and reused.

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

The model is being deployed on AWS.

5. CONCLUSION

The flight fare prediction uses training information for forecast the value. As a result, the user will be able to estimate the cost of their travel.