**Detailed Project Report**

**Flight Fare Prediction**

|  |  |
| --- | --- |
| Written By | Adil Anwar |
| Version | 1.2 |
| Date | 02-10-2022 |

**Document Change Control Record**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| 1.0 | 02-09-2022 | Adil Anwar | Abstract  Introduction  General Description |
| 1.1 | 21-09-2022 | Adil Anwar | Technical Requirements  Data Requirements  Data Preprocessing Design  Flow |
| 1.2 | 02-10-2022 | Adil Anwar | Data from User and its validation  Rendering the Results  Deployment  Conclusion |

# **Introduction**

1.1 Why this DPR Document ?

The main purpose of this DPR documentation is to add the necessary details of the

project and provide the description of the machine learning model and the written code.

This also provides the detailed description on how the entire project has been designed

end-to-end.

Key points :

• Describes the design flow

• Implementations

• Software requirements Architecture of the project Non-functional attributes

like:

• Reusability

• Portability

• Resource utilization

**General Description**

2.1 Problem Perspective

The flight fare prediction may be a machine learning model that helps users to predict

the price of the flight tickets and help them to understand the price of their journey.

2.2 Problem Statement

After amendment of the new rules, there is changes in the flight fare price from one

location to another. The main goal of the system is to create a model to predict the price

of their flight fare on the basis of bound input provided by user like date of journey,

Source, Destination and many more.

2.3 Proposed Solution

To solve the problem, we have created a User interface for taking the input from the

user to predict the flight fare price using our trained ML model after processing the

input and at last the output ( predicted value ) from the model is communicated to the

User.

2.4 Further Improvements

We also analysis the data used for training the ML model by considering different

occasions such as Weekday, Season or any Social reasons, considering different angle

of business. If we method such information and predict the discounted flight fare price,

it will bring some loss to the airline companies but user can get benefit from that. If we

develop these using Business perspective of Airline, this technique isn’t thought - about.

**Technical Requirements**

As technical requirements, we doesn’t need any specialized hardware for virtualization

of the application. The user should have the device that has the access to the web and

the fundamental understanding of providing the input.

3.1 Tools Used

• Python 3.9 is employed because the programming language and frameworks like

NumPy, Pandas, Scikit - learn and alternative modules for building the model.

• Jupyter - Notebook is employed as IDE.

• For Data visualizations, seaborn and components of matplotlib are getting used.

• For information assortment prophetess info is getting used.

• Front end development is completed victimization HTML/CSS.

• Flask is employed for each information and backend readying.

• GitHub is employed for version management.

• Heroku is employed for deployment

**Architecture**

Following workflow was followed during the entire project.

**Dataset**

Price

Route

R

**FLIGHT**

**FARE**

**PREDICTION**

Arrival Date

Departure Date

Destination

Airline

Arrival Time

Departure Time

No. of Stops

Source

HYPOPITUITARY

Additional Info

R

**Data Analysis Steps**

DATA

COLLECTION

DATA

PREPROCESSING

EXPLORATORY DATA

ANALYSIS

FEATURE

SELECTION

MODEL

CREATION AND

EVALUATION

In step 1, we collect

data which is generally

present in a database

or on internet.

In step 2, we

preprocess the data

which involves data

cleaning by handling

outliers, null values

etc.

In step 3, we explore

the data by performing

univariate and

bivariate analysis on

the features.

In step 4, we use

feature selection

techniques to filter out

the most important

features to perform

model creation

In step 5, we finally

build models on our

dataset and choose

the model which gives

the best accuracy.

**Gradient Boost Model**

**INTRODUCTION:**

* The gradient boost model is a supervised learning algorithm which we can use for regression and classification problems. It is among the most popular machine learning algorithms due to its high flexibility and ease of implementation.
* It gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees.
* Gradient Boost Regressor being ensembled algorithm tends to give more accurate result. This is because it works on the principle i.e., number of weak estimators when combined forms strong estimator. Even if one or few decision trees are prone to noise, overall results would tend to be correct. Even with small number of estimators (=30), it gives us high accuracy.

**MODEL TRAINING AND VALIDATION WORKFLOW**

**Model Creation and Evaluation**

❑ Various regression algorithms like Logistic Regression, Random Forest, Decision Tree, Support Vector Machine and many more were tested.

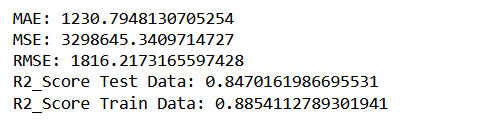
❑ Random Forest, Decision Tree and Gradient Boost Regressor were given better results. Gradient Boost Regressor was chosen for the final model training and testing.

❑ Hyper parameter tuning was performed.

❑ Model performance evaluated based on R2 Score, Cross Val Score, RSME Score.

**Model Prediction Results**

**on Test Dataset**

** Model Deployment**

**Model Deployment**

* The final model is deployed on Azure using Flask framework.



**Conclusion**

* The Flight Fare Prediction system will predict the price for helping the customers with the trained knowledge with set of rules. The user can use this system to recognize the approximate value of its flight fare for his or her journey.

**Frequently Asked Questions**

Q1) What is the source of data?

The data for training is obtained from Kaggle and is provided by Machine Hack.

Kaggle Link: <https://www.kaggle.com/datasets/nikhilmittal/flight-fare-prediction-mh>

Q2) What was the type of data?

The data was the combination of numerical and Categorical values.

Q3) What’s the complete flow you followed in this Project?

Refer Page no 6 for better Understanding.

Q4) After the File validation what you do with incompatible file or files which didn’t pass the validation?

Files like these are moved to the Achieve Folder and a list of these files has been shared with the client and we removed the bad data folder.

Q5) How logs are managed?

We are using different logs as per the steps that we follow in validation and

modeling like File validation log, Data Insertion, Model Training log, prediction

log etc.

Q 6) What techniques were you using for data pre-processing?

• Removing unwanted attributes

• Visualizing relation of independent variables with each other and output variables

• Checking and changing Distribution of continuous values

• Removing outliers

• Cleaning data and imputing if null values are present.

• Converting categorical data into numeric values.

Q 7) How training was done or what models were used?

• Before dividing the data in training and validation set, we performed

preprocessing over the data set and made the final dataset.

• As per the dataset training and validation data were divided.

• Algorithms like Linear regression, SVM, Decision Tree, Random Forest,

XGBoost were used based on the recall, final model was used on the

dataset and we saved that model.

Q 8) How Prediction was done?

• The testing files are shared by the client. We Performed the same life cycle on the provided dataset. Then, on the basis of dataset, model is loaded and prediction is performed. In the end we get the accumulated data of predictions.

Q 9) What are the different stages of deployment?

• First, the scripts are stored on GitHub as a storage interface.

• The model is first tested in the local environment.

• After successful testing, it is deployed on Azure.

Q 10) How is the User Interface present for this project?

• Created form for user input prediction.

• UI is very user friendly and easy to use

**Thank You**