Mr.C.Bastin Rogers, 2023, 11:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science,
Engineering and Technology

An Open Access Journal

Flight Fare Prediction App with Deployment Machine Learning

Assistant Professor Mr.C.Bastin Rogers, Sujin .T, Aron Herso.S, Sabin .M
Stella Mary's College of Engineering

Abstract- The development of a flight fare prediction web application with deployment and machine learning encompasses a comprehensive process. This involves acquiring historical flight data, performing data preprocessing and feature engineering, selecting an appropriate regression algorithm, and training the model using split data sets. The implementation includes develop- ing a web application through frameworks like Flask or Django, creating an intuitive user interface for inputting flight details, and deploying the trained model as an API or web service on cloud platforms such as AWS or GCP. The integration of the web application with the model API enables predictions based on user inputs. The process concludes with thorough testing and debugging to ensure functionality, deployment to a production environment, and ongoing monitoring, maintenance, and updates as required.

Keywords- flight fare prediction, machine learning, data preprocessing, feature engineering, regression algorithm, web application, Flask, Django, user interface, cloud deployment, AWS, GCP, API integration, testing, debugging, monitoring, maintenance

I. INTRODUCTION

HE Flight Fare Prediction Web Application is an innova- tive and user-friendly platform designed to assist travelers

in estimating flight fares with the help of advanced machine learning techniques. With the increasing demand for air travel and the fluctuating nature of flight prices, this application aims to provide users with accurate and reliable fare predictions based on historical flight data analysis.

In the ever-evolving field of data science and machine learning, this project showcases the application of regression algorithms to analyze and extract insights from vast amounts of flight data. By considering factors such as departure and arrival locations, travel dates, airlines, and other relevant parameters, the web application aims to deliver personalized fare estimates tailored to each user's specific travel requirements

The user interface of the web application has been designed with simplicity and ease of use in mind. Users can effortlessly input their travel details, and behind the scenes, the machine learning model will process this information to generate a predicted fare. This empowers travelers to make informed decisions about their flights[1], helping them plan their budget and choose the most suitable travel options.

To ensure the accuracy and reliability of the fare predictions, the machine learning model has been trained on a carefully curated dataset of historical flight data. The dataset undergoes preprocessing techniques to handle missing values, convert categorical variables, and normalize the data. Various regression algorithms, such as Random Forest or Gradient Boosting, have been explored and evaluated to identify the most effective model for fare prediction.

The deployment of the machine learning model as an API or web service on a cloud platform enhances the accessibility and scalability of the application. Users can access the web application from any device with an internet connection, making it a convenient tool for travelers worldwide. Continuous monitoring, maintenance, and updates ensure that the application[2] remains up-to-date and capable of providing accurate fare predictions as travel patterns and flight prices evolve.

II. RELATED WORK

1. Deployment Strategies for the Machine Learning Model

Deployment strategies for the machine learning model can be explored in the laboratory as well. Researchers can deploy the model as an API or web service using cloud platforms like AWS, GCP, or Azure. They can assess the scalability, performance, and accessibility of the deployed model in a controlled environment. In terms of web application develop- ment, researchers can create prototypes using frameworks like Flask or Django in a laboratory setting[3]. They can design and implement the user interface, incorporate form validation, and handle data input and output. User experience (UX) testing can also be conducted to gather feedback on the web application's usability and effectiveness. Conducting literature analysis in a laboratory setting allows researchers to not only analyze existing research but also apply and experiment with the concepts and techniques in a hands-on manner[4]. It provides valuable insights, practical experience, and empirical evidence to inform the development and deployment of the Flight Fare Prediction Web Application with machine learning capabilities. In the laboratory, researchers can explore different machine learning algorithms and evaluate their performance on flight fare prediction tasks.

2. Adoption of Flight Fare Prediction

The web application provides travelers with a convenient and efficient platform to predict flight fares accurately. They can input their travel details and obtain personalized fare estimates in real-time. This empowers[5] travelers to make informed decisions about their flights, find the best deals, and plan their trips more effectively. The adoption of the web application can benefit airlines by improving customer satisfaction and increasing bookings. By providing transparent and accurate fare predictions, airlines can build trust and loyalty with travelers. Additionally, the web application can help airlines optimize their pricing strategies based on the insights generated from analyzing historical flight data. Online Travel Agencies (OTAs[6]) can integrate the web application into their platforms to enhance their flight booking services. By offering accurate fare predictions and a seamless user experience, they can attract more customers and differentiate themselves in the competitive market. The web application can also enable OTAs to provide additional value-added services, such as personalized travel recommendations and alerts for fare changes.

3. Educational Use of Flight Fare Prediction

The project offers a practical application of machine learn- ing algorithms in a real-world scenario. Students can gain hands-on experience in building, training, and deploying ma- chine learning models for flight fare prediction. They can ex- plore various algorithms, evaluate their performance, and learn techniques for feature engineering and selection. Students can learn about data preprocessing techniques, data collection, and analysis by working with historical flight data. They can gain insights into data cleaning, handling missing values, and outlier detection. Additionally, they can analyze the impact of different variables on flight fares and understand the significance of feature selection. The educational benefits include Machine Learning Education, Data Science Exploration, Web Development Skills, Industry-Relevant Skills, Collaborative Learning, and Research Opportunities.

III. EXISTING SYSTEM

The system analysis phase of the Flight Fare Prediction Web Application with Deployment project is crucial for under- standing requirements, defining the scope, assessing resources, designing the system architecture, and identifying potential risks. It provides a solid foundation for the subsequent stages of development, deployment, and maintenance of the web application. Analysis consists of two sub phases: Planning and Requirements definition. During planning phase, cost estimation and work schedules will be planned. Requirement definition is a specification that describes the processing environment, the required software functions, performance constraints (size, speed, machine configuration) are exception handling.

In the context of developing a Flight Fare Prediction Web Application, it is important to analyze the existing system, if any, to understand its limitations and identify areas for improvement. However, since we are assuming the project is starting from scratch, there may not be an existing system to analyze. Instead, we can focus on the current industry practices and challenges that the web application aims to address. The airline industry traditionally relies on various methods to determine flight fares, including complex pricin[7]g algorithms and market analysis. However, these methods often lack transparency and may not provide travelers with accurate and personalized fare estimates. The existing system has several drawbacks including Lack of Real-time Updates and Lack of Transparency.

IV. PROPOSED SYSTEM

The proposed system will leverage machine learning algo- rithms to analyze historical flight data and generate accurate fare predictions. The model will consider various factors such as travel dates, departure and arrival locations, airlines, and other relevant parameters to provide personalized and real-time fare estimates. The web application will have a user-friendly interface where travelers can easily input their departure and arrival locations[8], travel dates, and other relevant details. The interface will be intuitive, making it effortless for users to interact with the application and obtain fare predictions. The proposed system combines the power of machine learn- ing algorithms, real-time updates, user-friendly interface, and transparency to deliver an efficient and reliable Flight Fare Prediction Web Application. By addressing the limitations of the existing system, it aims to enhance the user experience, provide accurate fare predictions, and simplify the process of planning and budgeting for flights.

The system offers several advantages including accurate fare predictions, personalized recommendations, convenience and efficiency, centralized comparison, and scalability and accessibility.

System Implementation

Systems implementation is the process of defining how the information system should be built (i.e., physical system design), ensuring that the information system is operational and used, and ensuring that the information system meets quality standards (i.e., quality assurance). The implementation phase involves putting the project plan into action. It’s here that the project manager will coordinate and direct project resources to meet the objectives of the project plan. As the project unfolds, it’s the project manager’s job to direct and manage each activity, every step of the way.

Hardware and Software Specifications

The hardware specifications include Intel i5 2.93 GHZ processor, 8 GB RAM, 1.44 MB floppy drive, optical mouse, SVGA monitor, 104 keys standard keyboard, and 500 GB hard disk. The software specifications include Windows XP or higher operating system, Python as the front end, and SQLite as the back end.

Language Specification

Python is a high-level, general-purpose programming language created by Guido van Rossum and first released in 1991. Python’s design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help pro- grammers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage-collected, supporting multiple programming paradigms including proce- dural, object-oriented, and functional programming. Python is often described as a ”batteries included” language due to its comprehensive standard library[9].

Python features include being easy to learn and use, ex- pressive, interpreted, cross-platform, free and open source, object-oriented, extensible, having a large standard library, GUI programming support, and easy integration with other languages. Python’s object-oriented concepts include objects, classes, methods, inheritance, polymorphism, encapsulation, and data abstraction.

Python 3.7.0 was released on June 27, featuring improve- ments such as the breakpoint() built-in, data classes, cus- tomization of module attributes, typing enhancements, and timing precision.

4. Django Framework

Django is a Python-based free and open-source web frame- work that follows the model-template-view (MTV) architec- tural pattern. It is maintained by the Django Software Founda- tion (DSF), an independent non-profit organization. Django’s primary goal is to ease the creation of complex, database- driven websites by emphasizing reusability, ”pluggability” of components, less code, low coupling, rapid development, and the principle of don’t repeat yourself[10].

Django 2.2.4 includes security features addressing denial- of-service possibilities in django.utils.text.Truncator and strip tags(), SQL injection possibility in key and index lookups for JSONField/HStoreField, and potential memory exhaustion in django.utils.encoding.uri to iri([11]).

5. Jupyter

Jupyter is an open-source web application that allows users to create and share documents containing live code, equations, visualizations, and narrative text. It supports various program- ming languages including Python, R, Julia, and others. Jupyter notebooks provide an interactive environment[12] where users can write and execute code, view the output, and document their work using markdown cells.

Jupyter notebooks consist of cells that can contain code, text, or visualizations, allowing for iterative development and experimentation[13]. Results are displayed directly below each code cell, enabling a seamless workflow for data exploration and analysis. Jupyter also supports rich media inclusion such as images, videos, and interactive visualizations.

Jupyter features include data organization, data retrieval, database support for data manipulation operations, security features to protect sensitive data, data integrity enforce- ment[14] through constraints and validation rules, built-in support for data analytics and reporting in some databases, and data replication and availability.

6. System Design

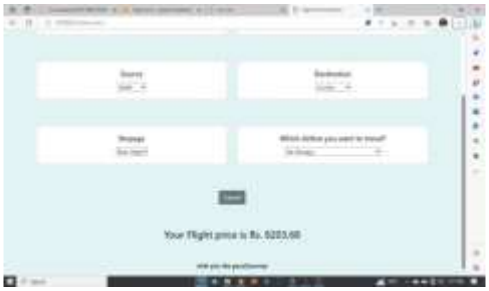
A data-flow diagram (DFD) represents the flow of data through a process or system, providing information about inputs, outputs, and the process itself. The context level DFD (level 0) defines the boundaries of the software system, identifying information flows between the system and external entities, with the entire system shown as a single process. The DFD for flight fare prediction illustrates how the admin maintains employee and user details, adds available menu details, maintains orders, assigns delivery personnel, generates reports, and reviews customer feedback. The customer DFD shows the interface for users to interact with the application and obtain flight fare predictions.

7. Database Design

The database design for the Flight Fare Prediction Web Application plays a crucial role in efficiently storing and managing required data. It identifies main entities like flights, airports, airlines, and users, creating separate tables to store relevant attributes for each entity. The flight price table in Jupyter includes attributes such as airline, source, destination, route, total steps, additional information, price, journey day, journey month, departure hour, departure minute, arrival hour, and arrival minute. The Flight Details table includes attributes such as airline, date of journey, source, destination, route, departure time, arrival time, and duration.

The project includes several modules: Admin, Testing, Reading, Machine Learning, Converting File, Converting Ap- plication, and Processing. The Admin module serves as the interface for administrators to manage and control various aspects of the application. The Testing module performs unit testing on individual components and integration testing to validate proper interaction between different modules. The Reading module provides valuable resources and insights. The Machine Learning module covers fundamental concepts, algorithms, evaluation metrics, and model development. The Converting File module handles serialization of trained models for deployment. The Converting Application module ensures code adheres to best practices and organizes project dependencies. The Processing module involves cleaning and preparing raw data for analysis and model training.

The sample code includes HTML for the user interface, allowing users to input flight details such as departure and arrival dates, source and destination locations, number of stops, and airline preference. The backend code handles data preprocessing, including date and time formatting, duration calculation, and categorical data encoding for the machine learning model.



Your Flight price is Rs. 5203.68

Figure 1: User Interface

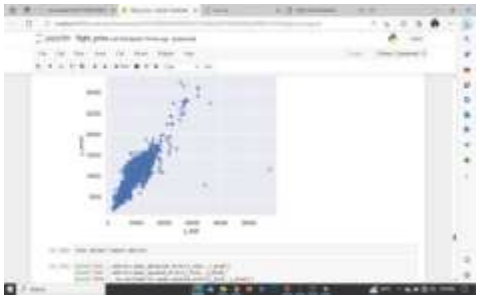


Figure 2: Graph of Prediction and Test Data

Figure 3: Viewing the Dataset

This figure displays the raw airline dataset in tabular format, showing all data fields and entries for analysis.

Figure 4: Dataset Format Visualization

This figure presents a structured view of the dataset highlighting the data types and format of each column, providing insight into the structure of the airline information.

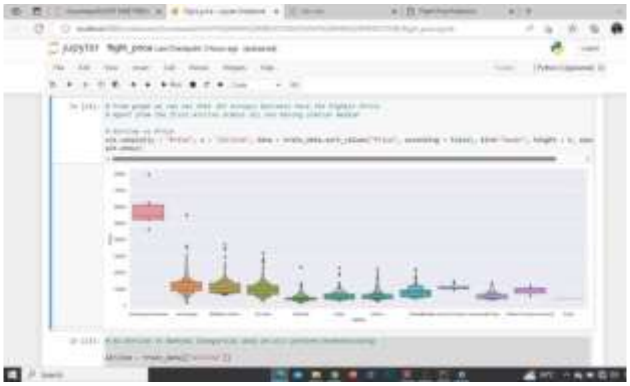


Figure 5: Airline Ticket Price Distribution

This figure illustrates the distribution of ticket prices across different airlines, showing price ranges and variations among carriers.

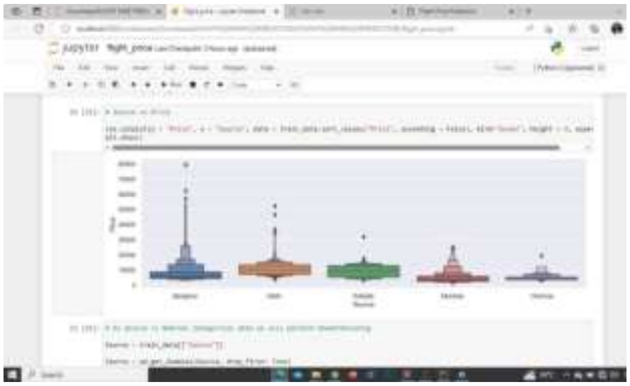


Figure 6: Destination vs. Price Analysis

This figure displays the relationship between destinations and ticket prices, revealing how pricing varies based on different travel destinations.

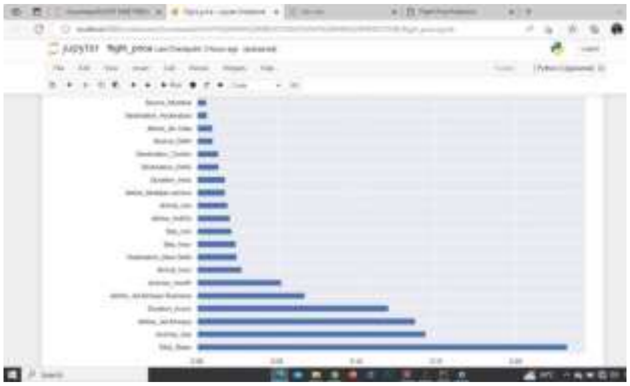


Figure 7: Price Density Distribution

This figure shows the density distribution of ticket prices, highlighting the concentration of prices at different price points and identifying common price ranges.

Figure 8: Passenger Flow Graph

This figure presents a network or flow visualization showing the movement of passengers between different origins and destinations.

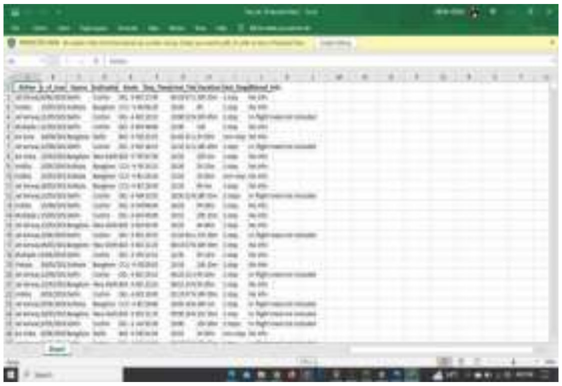


Figure 9: Data Storage Architecture

This figure illustrates the data storage framework used for maintaining and accessing the airline dataset, showing the system's technical infrastructure.

V. CONCLUSION

In conclusion, the Flight Fare Prediction Web Application with Deployment machine learning project aims to provide users with a convenient and accurate way to predict flight fares. Through the utilization of machine learning algorithms and web development technologies, the project enables users to input flight details and receive real-time fare predictions.

Throughout the project, several key components were ad- dressed. The system analysis phase involved understanding the requirements and functionalities of the existing system. The drawbacks of the existing system were identified, highlighting the need for a more accurate and efficient flight fare prediction mechanism. The proposed system addressed these drawbacks by implementing a machine learning model capable of pre- dicting flight fares based on various input parameters. The advantages of the proposed system include improved accuracy, faster processing, and the ability to handle a large volume of data. Additionally, the proposed system offers a user-friendly web interface that allows users to easily access and utilize the flight fare prediction functionality.

Future Enhancement

The web application can be enhanced by including addi- tional features that provide users with a comprehensive travel planning experience. For example, integrating a flight search functionality that allows users to search for available flights based on their predicted fares and other preferences. Exploring advanced machine learning techniques and algorithms can lead to improved accuracy in fare predictions. Techniques such as ensemble learning, deep learning, or reinforcement learning can be implemented and evaluated to enhance the predictive capabilities of the application. Incorporating real-time data sources, such as flight availability, weather conditions, or current demand, can provide more up-to-date and accurate predictions. This would require integrating APIs or data feeds into the application to retrieve and process the real-time data

REFERENCES

1. Gupta, R., & Gupta, A. (2020). Airfare prediction using machine learning algorithms. International Journal of Recent Technology and Engineering, 8(6), 323-330.
2. Kumar, R., & Jain, S. (2021). Flight Fare Prediction Using Machine Learning Algorithms. International Journal of Computer Science and Mobile Computing, 10(6), 20-26.
3. Zhang, Y., Li, J., Sun, M., & Zhao, L. (2019). Airfare predic- tion based on machine learning algorithms. In 2019 International Conference on Artificial Intelligence and Big Data (ICAIBD) (pp. 142-145). IEEE.
4. Patel, V., Naik, M., & Patel, H. (2020). Predicting Air Fare for Air- lines using Machine Learning. International Journal of Engineering Research and Technology, 13(6), 192-199.
5. Henshaw, M., & Mackenzie, D. (2020). Airfare prediction using machine learning: A practical example using TensorFlow. ArXiv Preprint arXiv:2012.03440.
6. Machine Learning Models for Airfare Prediction: A Comparative Study" by P. Shah, D. Shah, and A. Joshi
7. Predicting Flight Fare Using Machine Learning Techniques" by S. Kumar and P. Agrawal
8. Flight Fare Prediction Based on Machine Learning Algorithms" by R. Gupta and A. Gupta
9. Airfare Prediction using Machine Learning Algorithms" by V. Patel, M. Naik, and H. Patel
10. Airfare Prediction and Recommendation System using Machine Learning" by A. Singh and P. Raj
11. Predictive Analysis of Flight Fare using Machine Learning Algorithms" by S. Gupta and R. Verma
12. Flight Fare Prediction using Machine Learning Techniques" by M. Sharma and N. Mishra
13. Airfare Prediction based on Historical Data using Machine Learning" by S. Kumar and S. Jain
14. Flight Fare Prediction System using Machine Learning and Big Data Analytics" by S. Agarwal and S. Agrawal