Report: Flight Data Analysis and Prediction Project

Abstract

This report presents a comprehensive study on predicting flight prices while integrating ecological metrics to promote sustainable tourism. The primary goal is to develop a machine learning-based tool that empowers travelers to make cost-effective and environmentally responsible decisions. Data was collected from two primary sources: Amadeus API, providing over 180,000 flight records from global destinations, and Kayak, focusing on Paris airports and international routes. Advanced data processing techniques, including feature engineering for carbon emissions and distance calculations, were implemented to enhance predictive modeling. Using XGBoost models, a custom EcoScore metric was developed to balance affordability and sustainability. The project culminates in a user-friendly Streamlit application, allowing travelers to compare flight prices alongside their environmental impact. This work highlights the potential of integrating machine learning and ecological considerations in the travel industry, paving the way for more responsible tourism.

Introduction

Problem Statement

This project addresses tourism-related issues by analyzing and predicting flight prices while considering environmental impact. The goal is to provide an interactive prototype for travelers to choose flights that balance cost and ecological responsibility. The work was conducted in four stages: defining the problem, data scraping, machine learning development, and creating an interactive Streamlit application.

1. Pre-Project: Use Case Definition

Objective

The use case focuses on helping travelers make informed decisions about flights by predicting prices and evaluating the carbon emissions for each flight. This aligns with the tourism domain and incorporates an ecological dimension, enabling users to choose flights with lower environmental impact.

Relevance

This project addresses key challenges in the tourism industry, emphasizing the importance of responsible travel. By incorporating carbon emissions into flight analysis, it aligns with global efforts to combat climate change. Additionally, it demonstrates the practical application of machine learning in solving real-world problems.

Feasibility Testing

- **Source Sites**: Flight information was scraped from the Kayak platform using Selenium and from the Amadeus API.
- **Feasibility Test**:
  - A Python script was developed to generate potential flight routes from Paris airports (CDG, ORY) to 20 popular international destinations over a three-month period.
  - A scraper using **Selenium** was tested to retrieve flight details, proving the feasibility of obtaining useful data for the project.
- The datasets obtained from Kayak and the Amadeus API were combined at the beginning of the notebook. During this process, we ensured that only the required columns were retained to create a unified dataset for analysis and machine learning tasks.

2. Data Collection and Processing

Data Scraping

- **Tool**: Selenium WebDriver/Amadeus API
- **Inputs**:
  - A script generated a list of flight searches with parameters: origin, destination, and date.
  - Inputs included 20 destinations from two Paris airports over 90 days, resulting in thousands of search entries.
- **Output**:
  - CSV file containing flight details such as price, duration, number of stops, airlines, and seat types.
- **Challenges**:
  - **Dynamic Content Loading:** Kayak's web pages load incrementally, leading to incomplete data scraping.
  - **CAPTCHAs:** Frequent interruptions from CAPTCHAs blocked scraping attempts.
  - **API Rate Limits:** Strict rate limits and quick expiration of API keys hindered data collection.
  - **Data Variability:** Inconsistent formats and missing values complicated dataset standardization.
  - **Multi-Carrier Flights:** Data involving multiple airlines was inconsistent and difficult to process.
  - **Combining Affordability and Ecology:** Balancing cost predictions with environmental impact in a single metric was complex.

Solutions

  - **Dynamic Content Loading:** Used incremental scrolling and explicit waits in Selenium to ensure all content loaded before scraping.
  - **CAPTCHAs:** Rotated user agents and introduced random delays to mimic human interaction and avoid detection.

- **API Rate Limits:** Managed smaller data batches and used multiple API keys to maximize the number of requests.
- **Data Variability:** Applied robust data cleaning techniques to standardize formats and handle missing fields with fallback mechanisms.
- **Multi-Carrier Flights:** Engineered a multi-carrier indicator feature and applied one-hot encoding to integrate the data.
- **Combining Affordability and Ecology:** Developed the custom **EcoScore** metric to balance price prediction and carbon emissions, fine-tuning weights for optimal results.

Data Structure

Final scraped data included the following features:

- **Date**: Flight date.
- **Origin**: Departure airport.
- **Destination**: Arrival airport.
- **Price**: Ticket price in GBP.
- **Stops**: Number of stops (0-3).
- **Stop Airport**: Intermediate airport names for layovers.
- **Duration**: Total flight duration in minutes.
- **Company Name**: Operating airlines.
- **Seat Type**: Type of seat offered.

Data Cleaning

Steps taken:

1. **Duplicate Removal**: Ensured unique flight records.
2. **Transformations**:
   - `Stops` converted to numeric values (e.g., "1 stop" → `1`).
   - `Price` stripped of currency symbols and converted to integers.
   - `Departure Time` and `Arrival Time` standardized.
3. **Feature Engineering**:

- **Grouped Seat Types**: Consolidated into "Economy," "Business," etc.
- **Multi-Carrier Indicator**: Identified flights involving multiple airlines.
- **Distance Calculation**: Used geodesic coordinates of airports.
- **Carbon Emissions**: Computed as:

$$\text{Emissions (kg CO}_2\text{)} = \text{Distance (km)} \times 0.115 + \text{Stops} \times 50$$

3. Machine Learning

Problem Definition

The primary tasks were:

1. **Price Prediction**:
   - Using features like duration, stops, airlines, and seat type to predict flight prices.
2. **Eco-Friendly Score Integration**:
   - A custom **EcoScore** metric balanced price and carbon emissions.

Exploratory Data Analysis (EDA)

- **Correlation Analysis**:
  - Strong correlation between `Price` and `Duration` (0.46).
  - Moderate correlation between `Price` and `Stops` (0.36).
- **Feature Insights**:
  - `Seat Type` and `Multi-Carrier` were significant predictors.
  - Specific destinations (e.g., DXB, JFK) influenced prices due to regional demand.

Modeling

- **Algorithms**: XGBoost models were chosen for their efficiency with tabular data.
- **Features**:
  - Categorical: Origin, destination, airlines, and seat type.

    - Numerical: Stops, duration, distance, emissions, and temporal features (day, month, weekday).

- **Custom Metric**:

$$\text{EcoScore} = \alpha \cdot \text{Price Error} + \beta \cdot \text{Carbon Emissions}$$

- Balances affordability($\alpha = 1.0$) with ecological impact ($\beta = 0.1$).

Training Results

- **Classic Model**:
    - Focused solely on price prediction.
    - MAE: 31.51, RMSE: 72.90, R^2: 0.93
- **Eco-Friendly Model**:
    - Optimized for EcoScore.
    - MAE: 27.16, RMSE: 75.53, R^2: 0.93 with EcoScore improvement.

4. Interactive Streamlit Application

Overview

We succefully developed a user-friendly Streamlit application for travelers to input flight details and receive:

1. **Predicted Price**.
2. **EcoScore** and **Carbon Emissions** (Eco-Friendly Model).

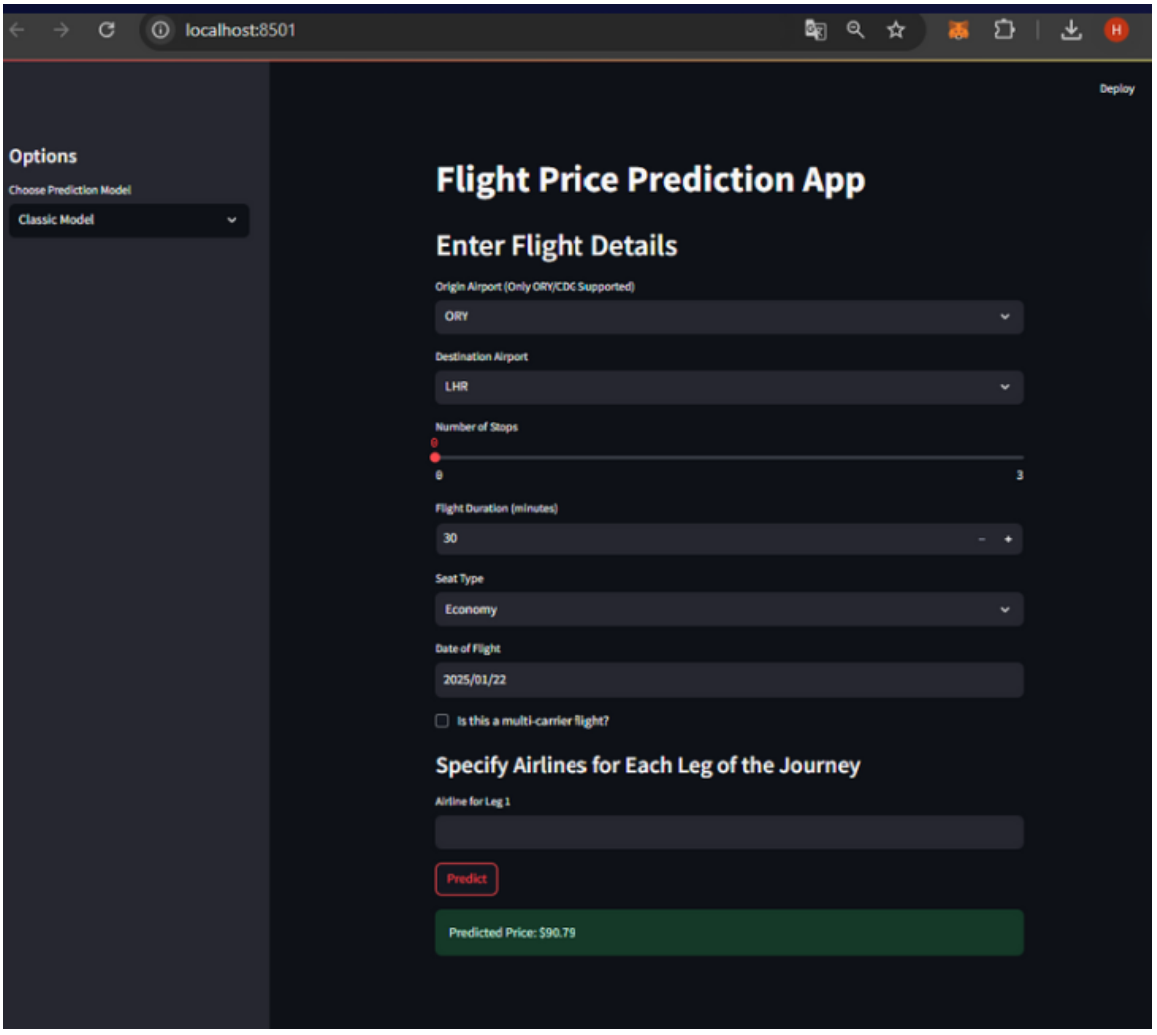
Features

1. **Input Parameters**:
    - Origin: Restricted to Paris airports (CDG, ORY).
    - Destination: 20 major international airports.
    - Stops, duration, seat type, and travel date.
2. **Outputs**:
    - Predicted price.
    - Eco-friendly details (emissions and EcoScore).

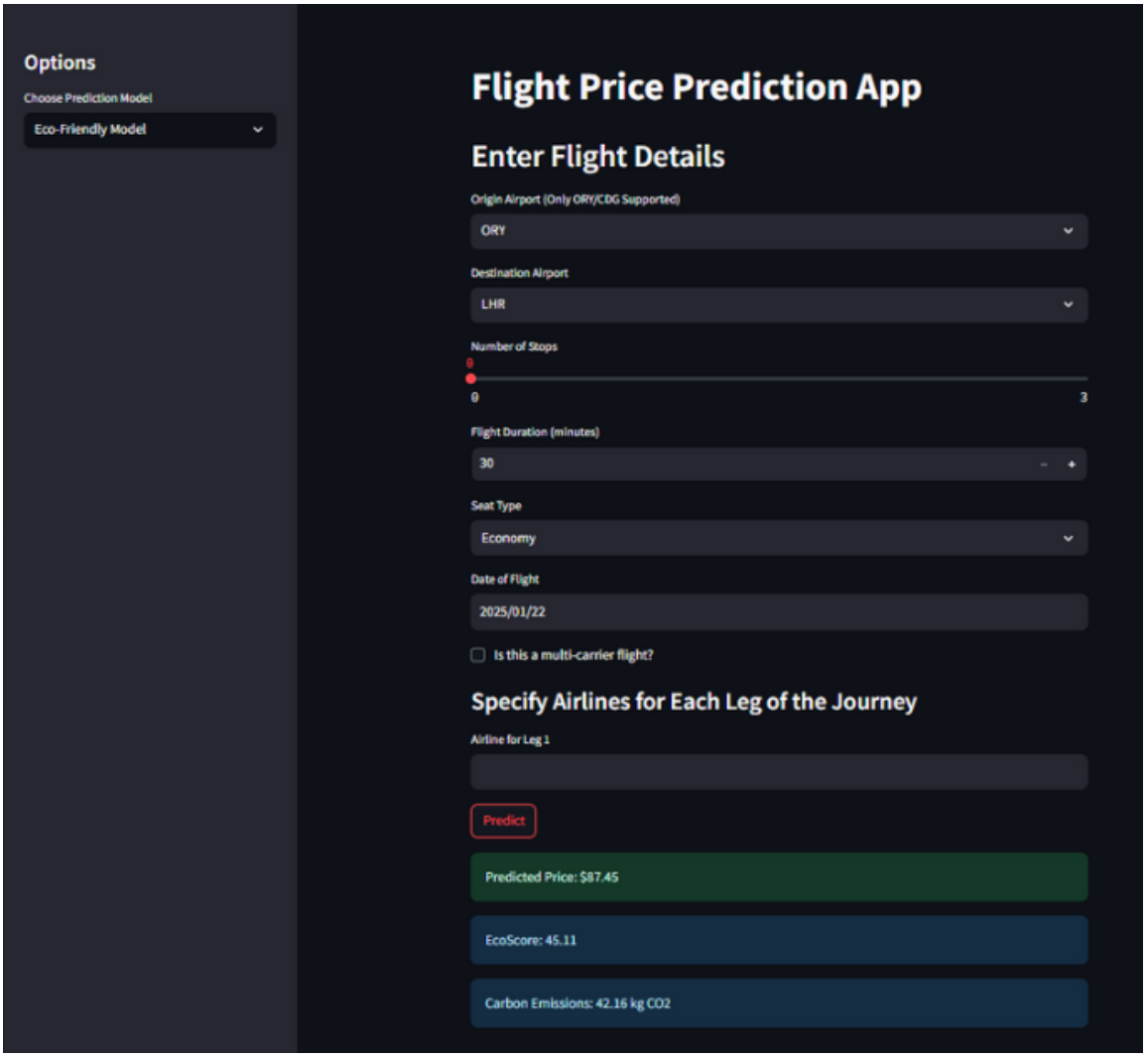
3. **Implementation**:
    - Derived features (e.g., distance, emissions) calculated in real-time.
    - Airline inputs allow multi-carrier scenarios.

    In the interface, we have the choice to choice between the two models:

    -Classic Model:



    -Eco-Friendly Model:



Key Challenges and Solutions

1. **Dynamic Content on Kayak**

   We addressed the challenge of dynamic content loading on Kayak by implementing incremental scrolling and using adaptive element selection to ensure flight details were fully loaded and captured accurately.

2. **Handling Multi-Carrier Flights**

   To manage flights operated by multiple airlines, we split the `Company Name` column into individual airlines and applied one-hot encoding. This allowed us to represent multi-carrier flights as a combination of features in the dataset.

3. **Environmental Considerations**

   We successfully integrated carbon emissions into the prediction model by calculating emissions based on flight distance and the number of stops.

This ensured that our analysis accounted for both affordability and ecological impact.

Conclusion

We successfully combined tourism, machine learning, and ecological considerations into a cohesive project. Through meticulous data collection, cleaning, and modeling, we developed a functional Streamlit application that allows travelers to make informed decisions about flights. The application not only predicts flight prices but also provides insights into the environmental impact of each flight.

Our work highlights the importance of balancing affordability and sustainability in travel. We believe the integration of ecological metrics, such as carbon emissions, into flight analysis is a step toward more responsible tourism. In the future, we could extend this work by incorporating real-time data from airline APIs, expanding the range of destinations, and refining the EcoScore metric to include additional sustainability factors.