FLIGHT DELAY PREDICTION FOR AVIATION USING MACHINE LEARNING

Submitted in partial fulfillment of requirement for the award of the Degree

Bachelor of Computer Science

In the faculty of Computer Science of Bharathiar University, Coimbatore

Submitted by

TEAM ID: NM2023TMID18664

S.ANGALAPARAMESHWARI

S.SINDHU

M. R.SUVETHA

K.GAYATHRI

Under the guidance of

Ms..N.SUNDARAVALLI MCA.,M.Phil.,., Ph.D.

Lecturer , Department of Computer Science

DEPARTMENT OF COMPUTER SCIENCE

LRG GOVERNMENT ARTS COLLEGE FOR WOMEN

(Affiliated To Bharathiar University)

TIRUPUR-4

APRIL-2023

LRG GOVERNMENT ARTS COLLEGE

NAAN MUDHALVAN PROJECT WORK

(AFFILIATED TO BHARATHIAR UNIVERSITY)
TIRUPUR-641602

TITLE : Flight Delay Prediction for Aviation using machine learning

This is to certify that this is a bonafide record of work done by the above students of III B.Sc (CS) Degree **NAAN MUDHALVAN PROJECT** during the year 2022-2023

Submitted for the Naan Mudhalvan project work held

on………….20

CLASS MENTOR **HEAD OF DEPARTMENT**
[Ms.N.Sundaravalli] **[Dr.R.Parimala]**

INDEX

PROBLEM SELECTION

1. Binary Classification: Predicting whether a flight will be delayed or not. This is a common approach where the problem is framed as a binary classification task, where the model predicts whether a flight will be delayed (1) or not delayed (0) based on historical data, such as weather conditions, airline schedules, and previous flight delays.

2. Multiclass Classification: Predicting the severity of flight delays. This approach involves categorizing flights into multiple classes based on the severity of delays, such as mild delay, moderate delay, and severe delay. This can provide more detailed insights and allow airlines to take appropriate actions based on the predicted severity of delays.

3. Regression: Predicting the duration of flight delays. In this approach, the problem is framed as a regression task, where the model predicts the duration of flight delays in minutes or hours. This can provide more precise information about the expected delay time, allowing airlines to better manage their operations and communicate with passengers.

4. Time Series Forecasting: Predicting flight delays in a time series format. This approach involves analyzing historical flight data in a time series format, taking into account temporal dependencies and patterns, to predict future flight delays. This can be useful for airlines to proactively plan for potential delays and make operational adjustments accordingly.

5. Anomaly Detection: Identifying unusual events or anomalies that may cause flight delays. This approach involves detecting unusual patterns or events in historical data, such as abnormal weather conditions, airport

IDEATION

1. Weather-based Prediction: Develop a model that predicts flight delays based on weather conditions, such as temperature, precipitation, wind speed, and visibility. The model can use historical weather data in combination with flight data to identify patterns and correlations between weather conditions and flight delays.

2. Airline Performance Prediction: Build a model that predicts flight delays based on the historical performance of airlines, such as on-time performance, previous delays, and cancellations. The model can leverage data from various airlines to identify trends and patterns in airline performance that may impact flight delays.

3. Time of Day and Seasonal Patterns: Develop a model that captures time of day and seasonal patterns in flight delays, such as peak travel times, holiday seasons, and weather-related patterns. The model can help airlines proactively manage operations during high-impact periods to reduce delays.

4. Airport Congestion Prediction: Build a model that predicts flight delays based on airport congestion levels, such as runway capacity, air traffic control delays, and gate availability. The model can help airlines anticipate potential congestion-related delays and make operational adjustments accordingly.

5. Aircraft Maintenance Prediction: Develop a model that predicts flight delays based on aircraft maintenance schedules, repair history, and equipment availability. The model can help airlines identify potential maintenance-related delays and optimize maintenance scheduling to minimize disruptions.

6. Social Media and Customer Feedback Analysis: Build a model that analyzes social media and customer feedback data to predict flight delays. The model can capture information from social media posts, customer reviews, and complaints to identify potential issues that may cause flight delays, such as equipment malfunctions, service disruptions, or weather-related concerns.

7. Hybrid Models: Explore the use of hybrid models that combine multiple data sources and machine learning techniques, such as combining weather data with airline performance data, or using a combination of time series forecasting and anomaly detection approaches. These hybrid models can provide a more comprehensive and accurate prediction of flight delays.

REQUIREMENTS ANANLYSIS

1. Data Availability and Quality: Assessing the availability, quality, and relevance of data is crucial for building an accurate and effective machine learning model. Identify the types of data needed, such as historical flight data, weather data, airline performance data, and maintenance data, and evaluate their availability, completeness, accuracy, and timeliness. Determine if any data gaps need to be addressed and plan for data collection, preprocessing, and integration.

2. Prediction Accuracy and Precision: Define the desired level of prediction accuracy and precision based on the specific needs of the aviation industry. Consider factors such as the acceptable margin of error, the impact of false positives and false negatives, and the trade-offs between accuracy and model complexity. Set clear performance metrics and evaluate the model's performance against these metrics during model development and validation.

3. Real-time or Batch Prediction: Determine whether real-time or batch prediction is required for the flight delay prediction model. Real-time prediction involves predicting flight delays in real-time as new data becomes available, which may require low-latency and high-throughput processing capabilities. Batch prediction, on the other hand, involves predicting flight delays in batches or batches of data, which may allow for more extensive data processing and analysis.

4. Interpretability and Explainability: Consider the need for interpretability and explainability of the model's predictions, especially in the aviation industry where decisions can have significant safety and operational implications. Decide whether the model needs to provide interpretable and explainable results, such as feature importance rankings or decision rules, to gain trust and acceptance from stakeholders.

5. Scalability and Deployment: Determine the scalability requirements of the model, considering the volume and velocity of data in the aviation industry. Decide on the deployment environment, such as on-premises or cloud-based, and consider factors such as computational resources, latency requirements, and operational constraints. Plan for the deployment, monitoring, and maintenance of the model to ensure its continuous performance and effectiveness.

6. Regulatory and Compliance Requirements: Consider the regulatory and compliance requirements of the aviation industry, such as data privacy, security, and industry standards. Ensure that the developed model complies with relevant regulations, laws, and standards, and plan for necessary data handling, storage, and processing practices to meet these requirements.

7. Stakeholder Involvement: Involve relevant stakeholders, such as airline operators, aviation authorities, and end-users, in the requirement analysis process. Understand their perspectives, needs, and feedback to ensure that the developed model aligns with their expectations and requirements.

PROJECT DESIGN

1. Define Project Objectives: Clearly define the project objectives, such as predicting flight delays accurately, improving operational efficiency, minimizing passenger inconvenience, or optimizing resource allocation. Establish a well-defined problem statement that outlines the scope, goals, and expected outcomes of the project.

2. Data Collection and Preprocessing: Collect and preprocess the relevant data required for building the machine learning model. This may include historical flight data, weather data, airline performance data, maintenance data, and other relevant data sources. Clean, preprocess, and transform the data as needed, including handling missing values, outliers, and categorical variables, and ensure data quality and integrity.

3. Feature Engineering: Conduct feature engineering to identify and extract relevant features from the collected data that can be used as input features for the machine learning model. This may involve selecting relevant features, normalizing or scaling features, and creating new features that may capture important information related to flight delays.

4. Model Selection and Development: Choose appropriate machine learning algorithms based on the project objectives and data characteristics. Train and evaluate different models using appropriate techniques such as cross-validation to select the best-performing model. Fine-tune the model hyperparameters to optimize its performance. Consider using ensemble methods, such as stacking or bagging, to improve the model's accuracy and robustness.

5. Model Evaluation: Evaluate the performance of the developed model using appropriate evaluation metrics, such as accuracy, precision, recall, F1-score, and AUC-ROC, depending on the problem type (binary classification, multi-class classification, regression, etc.). Validate the model's performance on a separate test set to ensure its generalization ability. Perform thorough model analysis, interpretability, and sensitivity analysis to gain insights into its behavior and limitations.

6. Model Deployment: Deploy the trained machine learning model in a production environment, taking into consideration factors such as computational resources, latency requirements, and operational constraints. Implement necessary APIs or interfaces to enable the model to be integrated into the existing systems or workflows of the aviation industry. Ensure proper monitoring and maintenance of the deployed model to ensure its continuous performance and effectiveness.

7. Interpretability and Explainability: Consider the interpretability and explainability of the model's predictions, especially in the aviation industry where transparency and accountability are important. Develop techniques or methods to interpret and explain the model's predictions, such as feature importance rankings, decision rules, or visualizations, to gain trust and acceptance from stakeholders.

8. Documentation and Reporting: Document the entire project, including data collection and preprocessing steps, feature engineering, model selection and development, evaluation results, and deployment details. Create reports or dashboards to communicate the project findings, insights, and recommendations to relevant stakeholders, such as airline operators, aviation authorities, and end-users.

PROJECT PLANNING PHASE

1. Define Project Objectives: Clearly define the objectives of the project, such as predicting flight delays accurately, improving operational efficiency, minimizing passenger inconvenience, or optimizing resource allocation. Establish a well-defined problem statement that outlines the scope, goals, and expected outcomes of the project.

2. Identify Stakeholders: Identify the key stakeholders involved in the project, such as airline operators, aviation authorities, data providers, data scientists, IT personnel, and end-users. Understand their needs, expectations, and requirements, and involve them in the project planning process to ensure alignment and collaboration.

3. Define Scope and Timeline: Clearly define the scope of the project, including the specific aspects of flight delays to be predicted, the data sources to be used, and the timeframe for the project. Break down the project into smaller tasks and define the timeline for each task, including milestones, deadlines, and dependencies. Consider factors such as data availability, model development time, evaluation time, and deployment time when defining the timeline.

4. Allocate Resources: Identify and allocate the necessary resources for the project, including personnel, hardware, software, data, and budget. Consider the expertise and skill set required for the project, and ensure that the team has the necessary capabilities to carry out the tasks effectively. Define the roles and responsibilities of team members, and establish communication channels and reporting mechanisms.

5. Risk Assessment and Mitigation: Identify potential risks and challenges that may arise during the project, such as data quality issues, model performance limitations, technical challenges, or changes in requirements. Assess the impact and likelihood of each risk, and develop strategies to mitigate or minimize the risks. Include contingency plans and backup options in case unexpected issues arise during the project.

6. Define Deliverables: Clearly define the expected deliverables of the project, such as the trained machine learning model, evaluation results, deployment plan, documentation, and reports. Specify the format, quality, and timeline for each deliverable, and ensure that they align with the project objectives and stakeholder requirements.

7. Define Evaluation Metrics: Define the evaluation metrics to be used to assess the performance of the developed model, such as accuracy, precision, recall, F1-score, and AUC-ROC, depending on the problem type (binary classification, multi-class classification, regression, etc.). Clearly define the evaluation criteria and thresholds for success, and establish a plan for conducting thorough model evaluation.

8. Define Communication and Reporting Plan: Establish a communication and reporting plan to ensure effective communication among team members and stakeholders. Define the frequency, format, and channels of communication, and establish a reporting mechanism to provide regular updates on the progress, challenges, and results of the project.

PROJECT DEVELOPMENT PHASE

1. Data Collection and Preparation: Collect and gather the relevant data for flight delay prediction, which may include historical flight data, weather data, airline schedules, airport data, and other relevant data sources. Clean, preprocess, and transform the data to ensure its quality, consistency, and suitability for the machine learning models. Handle missing values, outliers, and inconsistencies in the data, and ensure that it is in the right format and structure for model training.

2. Feature Engineering: Extract and engineer relevant features from the collected data to represent the input variables for the machine learning models. Consider factors such as flight departure/arrival times, airport location, weather conditions, airline information, and other relevant variables that may impact flight delays. Perform exploratory data analysis (EDA) to gain insights and understand the patterns and relationships in the data, and select the most relevant features for model training.

3. Model Selection and Training: Select appropriate machine learning algorithms for flight delay prediction, such as decision trees, random forests, support vector machines, or deep learning models, depending on the specific problem requirements and data characteristics. Split the data into training and validation sets, and use the training data to train the selected models. Tune the hyperparameters of the models to optimize their performance, and evaluate their performance using the defined evaluation metrics.

4. Model Evaluation: Evaluate the performance of the trained machine learning models using the defined evaluation metrics. Use techniques such as cross-validation, holdout validation, or time-series validation, depending on the nature of the data and the problem requirements. Compare the performance of different models and select the best-performing model(s) for further refinement and deployment.

5. Model Refinement: Refine the selected model(s) based on the evaluation results and feedback. Fine-tune the model hyperparameters, adjust the feature engineering techniques, or explore ensemble methods to improve the model performance. Repeat the model training and evaluation process iteratively until the desired performance level is achieved.

6. Model Deployment: Deploy the trained and refined machine learning model(s) in a production environment, such as a web application, mobile app, or cloud-based service, depending on the project requirements. Ensure that the model is integrated seamlessly into the operational workflow and follows the necessary security and privacy measures. Conduct thorough testing and validation of the deployed model(s) to ensure its accuracy, reliability, and stability in real-world scenarios.

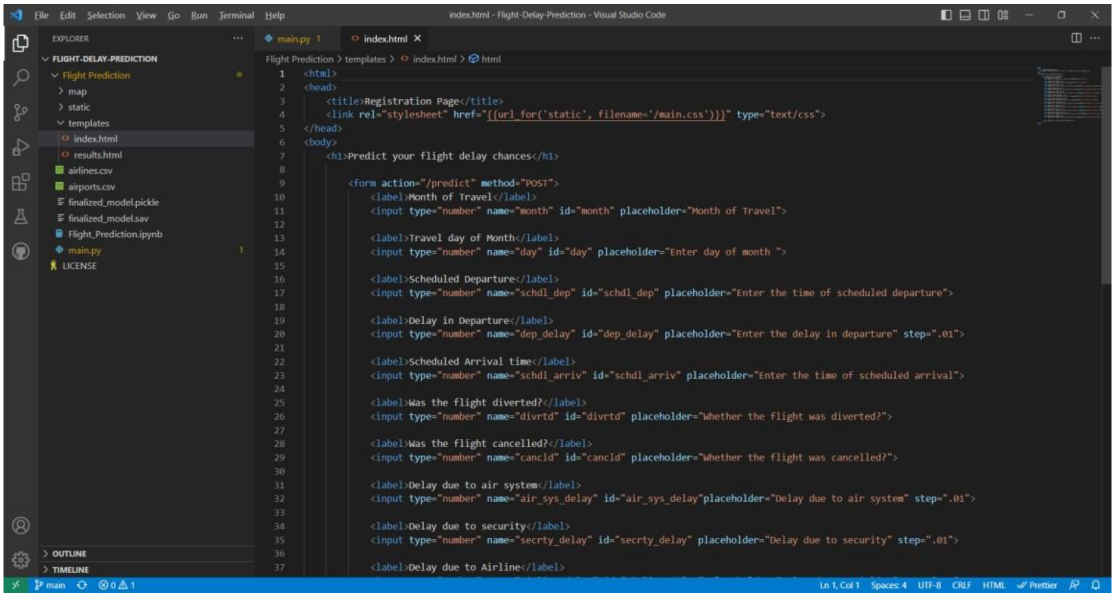
7. Documentation and Reporting: Document the entire project development process, including the data collection, preprocessing, feature engineering, model selection, training, evaluation, refinement, and deployment steps. Document the findings, insights, and lessons learned during the project, and prepare comprehensive reports and documentation for stakeholders and future reference.

RESULT

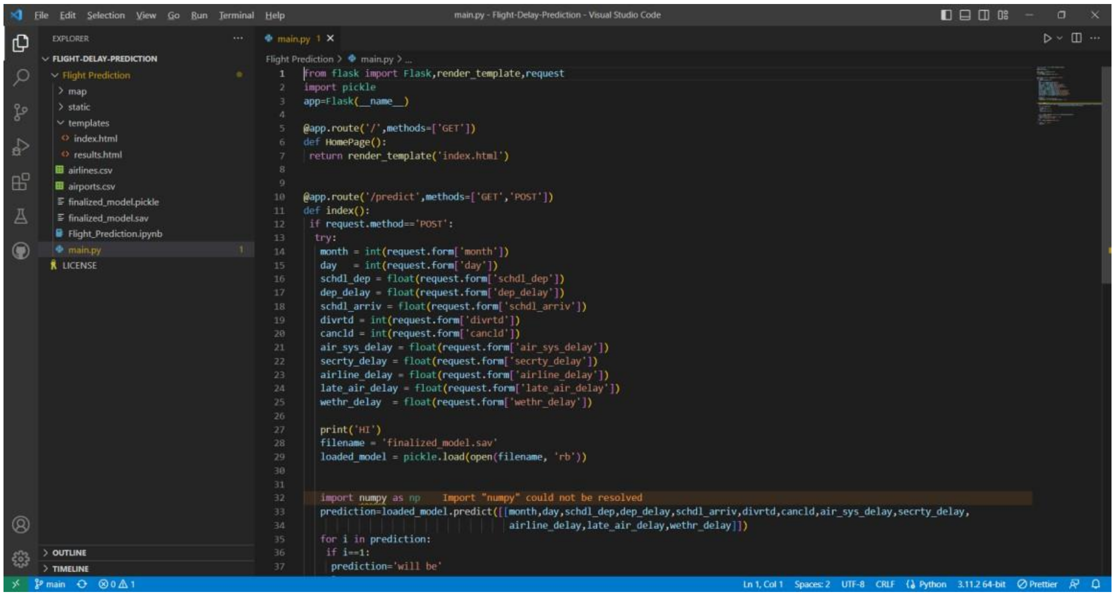

127.0.0.1:5000
Prediction of Flight Delay
Enter the Flight Number :
Month :
Day of Month :
Day of Week :
origin MSP
destination MSP
Scheduled Departure Time :
Scheduled Arrival Time :
Actual Departure Time :
SUBMIT


127.0.0.1:5000
Prediction of Flight Delay
Enter the Flight Number : 1399
Month : 2
Day of Month : 4
Day of Week : 5
origin JFK
destination SEA
Scheduled Departure Time : 1
Scheduled Arrival Time : 22
Actual Departure Time : 1
SUBMIT

SAMPLECODING



```html
<html>
<head>
    <title>Registration Page</title>
    <link rel="stylesheet" href="{{url_for('static', filename='/main.css')}}" type="text/css">
</head>
<body>
    <h1>Predict your flight delay chances</h1>

        <form action="/predict" method="POST">
            <label>Month of Travel</label>
            <input type="number" name="month" id="month" placeholder="Month of Travel">

            <label>Travel day of Month</label>
            <input type="number" name="day" id="day" placeholder="Enter day of month ">

            <label>Scheduled Departure</label>
            <input type="number" name="schdl_dep" id="schdl_dep" placeholder="Enter the time of scheduled departure">

            <label>Delay in Departure</label>
            <input type="number" name="dep_delay" id="dep_delay" placeholder="Enter the delay in departure" step=".01">

            <label>Scheduled Arrival time</label>
            <input type="number" name="schdl_arriv" id="schdl_arriv" placeholder="Enter the time of scheduled arrival">

            <label>Was the flight diverted?</label>
            <input type="number" name="divrtd" id="divrtd" placeholder="Whether the flight was diverted?">

            <label>Was the flight cancelled?</label>
            <input type="number" name="cancld" id="cancld" placeholder="Whether the flight was cancelled?">

            <label>Delay due to air system</label>
            <input type="number" name="air_sys_delay" id="air_sys_delay"placeholder="Delay due to air system" step=".01">

            <label>Delay due to security</label>
            <input type="number" name="secrty_delay" id="secrty_delay" placeholder="Delay due to security" step=".01">

            <label>Delay due to Airline</label>
```



```python
from flask import Flask,render_template,request
import pickle
app=Flask(__name__)

@app.route('/',methods=['GET'])
def HomePage():
 return render_template('index.html')


@app.route('/predict',methods=['GET','POST'])
def index():
 if request.method=='POST':
  try:
   month = int(request.form['month'])
   day   = int(request.form['day'])
   schdl_dep = float(request.form['schdl_dep'])
   dep_delay = float(request.form['dep_delay'])
   schdl_arriv = float(request.form['schdl_arriv'])
   divrtd = int(request.form['divrtd'])
   cancld = int(request.form['cancld'])
   air_sys_delay = float(request.form['air_sys_delay'])
   secrty_delay = float(request.form['secrty_delay'])
   airline_delay = float(request.form['airline_delay'])
   late_air_delay = float(request.form['late_air_delay'])
   wethr_delay  = float(request.form['wethr_delay'])

   print('HI')
   filename = 'finalized_model.sav'
   loaded_model = pickle.load(open(filename, 'rb'))


   import numpy as np
   prediction=loaded_model.predict([[month,day,schdl_dep,dep_delay,schdl_arriv,divrtd,cancld,air_sys_delay,secrty_delay,
                                     airline_delay,late_air_delay,wethr_delay]])
   for i in prediction:
    if i==1:
     prediction='will be'
```