Data Intensive Computing CSE 587

Project Phase – 3

TITLE: FLIGHT SATISFACTION CLASSIFICATION

Name	UB ID
Sriinitha Chinnapatlola	sriinith
Shravani Soma	ssoma

Problem Statement: We aim to consider various factors in predicting the passenger satisfaction for their flight. These factors include flight details like flight distance, inflight Wi-Fi services, food and drinks, seat comfort, departure, and arrival delays, and many more. We also take customer's details into consideration for evaluating the model. These considerations include gender, age, and their loyalty. This results in a classification model which predicts if a customer would either be 'satisfied' or 'dissatisfied' based on all the above given factors.

AIM: For the Phase-3 of the project, we aim to develop a User Interface (UI) for the client's (Airline Company) customers who can provide their details and give feedback on various aspects of their travel experience. This information can be used by the client to get an understanding of the overall satisfaction rate of their customers and using the visualizations that we provide, they can also infer the aspects that play an important role in deciding a passenger's overall satisfaction. From this analysis, the airline company executives can take informed decisions regarding the services that need to be improved and come up with various market strategies.

1. (a) Working instructions to demo/use the finished product.

1. Extract the zip file that has been uploaded. The python file consisting of the code for phase – 3 can be found inside the folder named "phase3"

2. Now, as we have developed using streamlit, we run the python file from the terminal.

3. Open the terminal, change the directory to the directory that consists of our python file (this folder also consists of train and test datasets).

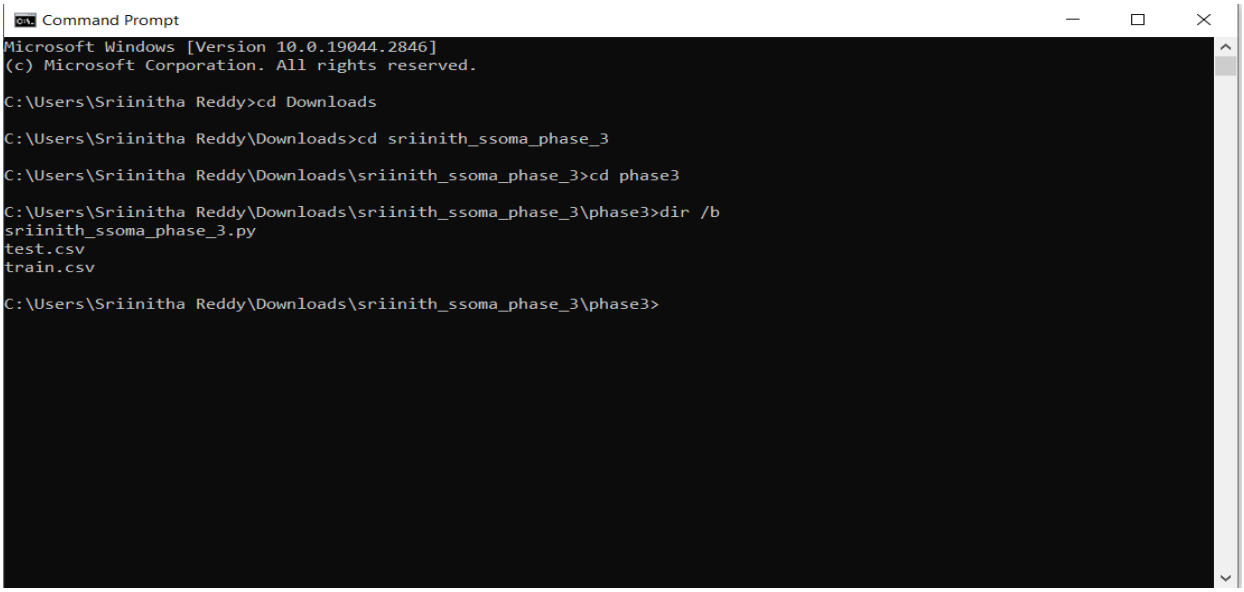

Fig: Example demonstrating the changing of current directory to the directory that consists of the python file.

Here, "**sriinith_ssoma_phase_3.py**" is the main python file that consists of the code for phase3.

4. The following python packages need to be installed before running the file:

Streamlit, numpy, pandas, matplotlib, seaborn, and sklearn.

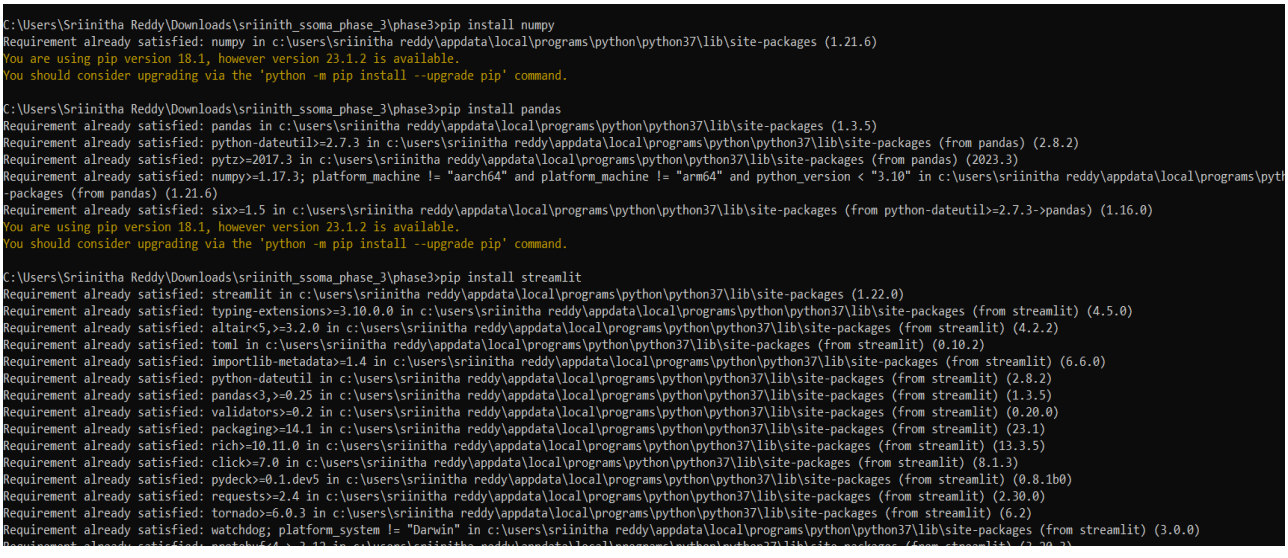

Fig: Downloading the above-mentioned libraries using the command "pip install package_name"

5. Once, we have changed the current working directory to the directory that consists of our python file, and after installing all the required packages, run the following command: "**streamlit run sriinith_ssoma_phase_3.py**"

```
C:\Users\Sriinitha Reddy\Downloads\sriinith_ssoma_phase_3\phase3>streamlit run sriinith_ssoma_phase_3.py

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://192.168.1.188:8501
```

Fig: localhost address

On pressing enter, it redirects to the default web browser where the UI (output) is visualized.

6. This is how the UI looks like that prompts the user to input for various fields.

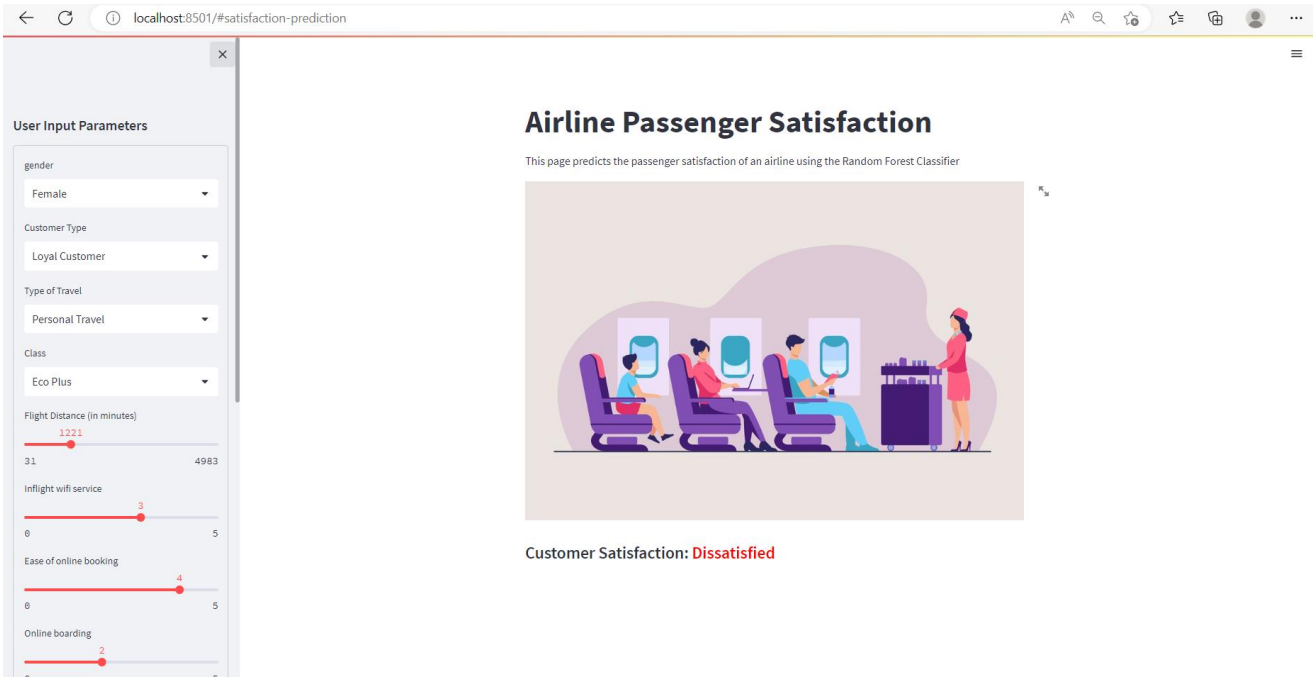

Fig: Screenshot of the UI

\

1. (b) Model Implemented from Phase 2:

In the phase – 2 of the projects, since ours is a classification problem, we have implemented 5 classification algorithms namely: Logistic Regression, K-nearest neighbor, Naïve Baye's algorithm, Support vector machine (SVM) and Random Forest.

Classification Algorithm	**Testing Accuracy**
Logistic Regression	87.52%
K-nearest Neighbour	91.6%
Naïve Baye's	86.23%
Support Vector Machine (SVM)	87.98%
Random Forest	95.61%

Fig: Table representing the various models and respective accuracies achieved

Out of these, we implemented our model using the **random forest algorithm** as the accuracy achieved was the highest among all the others. For our dataset, after performing various pre-processing steps, when the data has been fit to a random forest model, the accuracy achieved is **95.63%**.

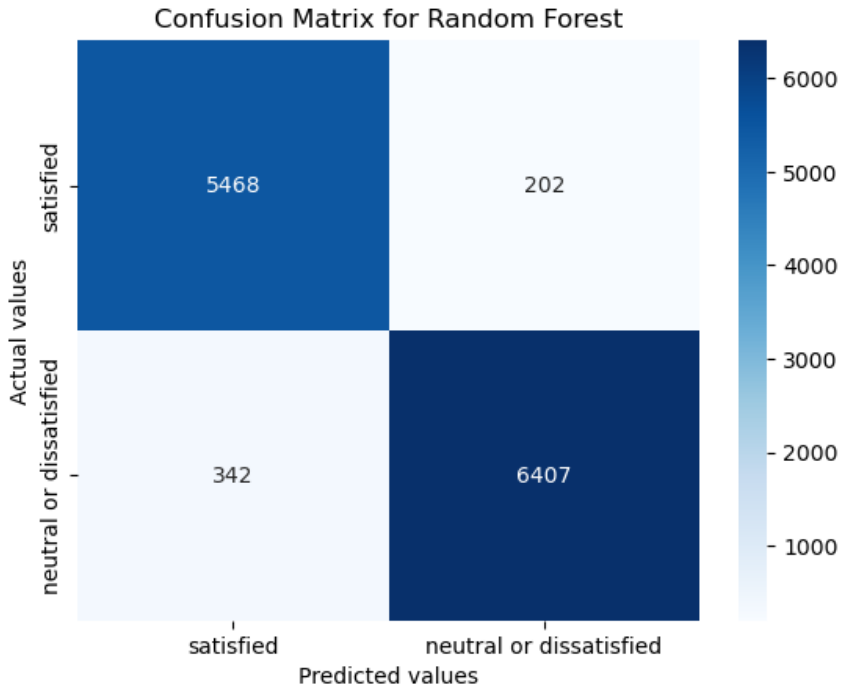

Fig: Confusion matrix for the Random Forest Model.

We have also plotted the confusion matrix as a comparison metric during the phase – 2 of the project, and from that we inferred that out of 5670 of the entries consisting of the target variable as "satisfied", 5480 values are correctly predicted as "satisfied" while the rest 190 are classified as "neutral or dissatisfied". And out of 6749 of the entries consisting of the target variable as "Neutral or Dissatisfied", 6416 values are correctly predicted as "Neutral or Dissatisfied" while the rest 333 are classified as "satisfied".

Feature Selection:

Here, after training the model with the random forest algorithm, we identified the 15 most import ant features that contributed to the overall satisfaction rate of the passenger. The 15 categories tha t are considered important based on the results obtained from the **random forest model are: ‘Gender’, 'Customer Type', 'Type of Travel', 'Class', 'Flight Distance', 'Inflight Wi-Fi servi ce', 'Ease of Online booking', 'Online boarding', 'Seat comfort', 'Inflight entertainment', 'O n-board service', 'Leg room service', 'Check-in service', 'Inflight service', 'Cleanliness'.**

On the UI, the user is prompted to input all the features so that the airlines can get an understanding of the overall satisfaction rate. And then, using various visualization techniques, we display the features that significantly contributed to the overall satisfaction rate of the customer so that the airline company can come up with strategies specifically designed to improve in those areas.

1. (c) Recommendations related to the problem statement:

What can users infer from our product:

Once, the customer gives feedback on the various aspects, as we have embedded a ML model, the client (Airline Company) can know if the passenger is satisfied or dissatisfied with their overall flight experience. Once, the user gives the feedback, along with knowing the satisfaction, the client can also visually see how much contribution each factor is contributing to the overall experience. Based on this, the client (Airline Company) can formulate specific strategies that target these aspects.

How does it help them to solve their problem?

From our product, our client can understand, what all aspects are affecting the overall satisfaction rate of the customer. Now that they know what the areas are they have to work on, it helps them come up with plans specifically designed for the areas they are lacking in. By implementing these strategies, the number of customers satisfied with their overall flight experience also increases.

Ideas on how to extend the product:

This can be treated as a basic version, as there can be more enhancements done to the UI. For suppose we can add more visualizations that show the percentage of customers satisfied with their flight experience out of all the customers who filled out the form. Similarly, this UI can be extended to show real time flight information such as departure and arrival times, and delays if any. The same UI can be turned into mobile application so that they can directly open from their mobile and access it anytime from anywhere. Another great extension to the current version can be the addition of space where users can share in detail about their experiences.

Conclusion:

In the 1st phase of the project, we have started with taking the dataset from Kaggle, preprocessing the dataset as it included a lot of null values and outliers, performed various visualization on the data to get a better understanding of the data. Later, in the 2nd phase of the project, we have implemented various classification algorithms on the dataset and chose the "random forest" algorithm as it gave the highest accuracy. In the 3rd phase of the project, we have developed an UI, where the customer can give feedback, and the client can understand if the user is satisfied or dissatisfied with their flight experience and come up with strategies to improve the overall satisfaction rate. The same webpage can be extended to implement various functionalities like providing real time flight updates, users describing about their personal experiences etc.

References:

1) scikit-learn: scikit-learn: machine learning in Python — scikit-learn 1.2.2 documentation

2) User guide and tutorial — User guide and tutorial — seaborn 0.12.2 documentation (pydata.org)

3) Plotly Python Graphing Library

4) Kaggle dataset: Airline Passenger Satisfaction | Kaggle

5) Streamlit: streamlit · PyPI