CS5691 : Pattern Recognition and Machine Learning

Data Contest Report

Team Liche
Chetan Reddy (CS19B012)
Likith Sai G(EE19B080)

1. Data merging

The input is spread over multiple csv files. To analyze this data, a single pandas dataframe containing all the relevant data is more convenient to work on. For this each csv file is loaded into a pandas dataframe using read_csv. And these individual dataframes are appended one by one using the common booking_id column.

2. Missing data handling

Sections #2 and #3 are for data cleaning, which is a very crucial part in making the data usable to train later on. Multiple columns like hotel_category, hotel_name_length, hotel_description_length, hotel_photos_qty, booking_create_timestamp, ,booking_approved_at, , booking_checkin_customer_date,booking_expiry_date have multiple empty elements. To handle this, they(np.nan values) are replaced by their corresponding column's measure of central tendency (mean here).

3. Removing Duplicates

Using drop_duplicates() function, duplicate data has been taken care of, after which only each booking_id appears only once, with the selection criterion among them being the one with the highest value of columns payment_installments & booking_sequence_id, which helps in choosing better features.

4. Features used

1. *booking_status*
2. Time1 = *booking_approved_at - booking_create_timestamp*
3. Time2 = *booking_checkin_customer_date - booking_create_timestamp*
4. Time3 = *booking_expiry_date - booking_create_timestamp*
5. *hotel_id*

6. *seller_agent_id*
7. *price*
8. *agent_fees*
9. *customer_unique_id*
10. *country*
11. *hotel_category*
12. *hotel_name_length*
13. *hotel_description_length*
14. *hotel_photos_qty*
15. Maximum of *payment_sequential*
16. *payment_installments*

NOTE: Some of these features are categorical features, They are encoded using factorize function in pandas library

5. Algorithm

The given train data is split into 80% training data and 20% validation data using train_test_split function from sklearn.

We tested various Classification Algorithms and observed that Random Forest Classifier Algorithm gave better results. The Random Forest Classifier is imported from sklearn library and is used with default hyperparameters.

The algorithm achieved an accuracy of 60.5% on an average and Mean Squared Error of 2.03 on validation data