Airbnb New User Bookings

Team : Gen Z

Dev Patel
IMT2018021
International Institute of Information Technology Bangalore
dev.patel@iiitb.org

Prateek Kamboj
IMT2018057
International Institute of Information Technology Bangalore
prateek.kamboj@iiitb.org

Ishaan Sachdeva
IMT2018508
International Institute of Information Technology Bangalore
Ishaan.sachdeva@iiitb.org

INTRODUCTION

Going back to pre-digital era when there was no internet, planning a travel excursion was much more complicated and opaquer. People were dependent on advertisements on print or electronic media to explore places. Travel agents were the one who had all the knowledge.

But over years, Technology has been the game changer in every sector and more and more people are coming to the internet to explore holiday destinations and make travel bookings. Now users can book places of their choice and plan things they like, to many different countries and places all over the world which can be done in a matter of a few clicks. With all these bookings being done there is a lot of user data being generated which can be used to make better predictions for a user and decrease the overall average time taken by a user. In this competition we have to use the Airbnb users data to accurately predict in which country a new user will make his or her first booking by doing analysis on the competition dataset.

We use machine learning to predict where a new user will book his /her first destination with 2 other most probable destinations. This report presents a methodology for analysis of user data and how that is being used to make predictions.

These report proceeds as follows:

1. *Dataset*
2. *Observations*
3. *Pre-processing and Feature extraction*
4. *Model selection*

I. DATASET

The dataset used in this project is a list of users along with their demographics, web session records, and some summary statistics, which was taken from Kaggle competition.

We are asked to predict which country a new users first booking destination will be. All the users in this dataset are from the USA. There are 12 possible outcomes of the destination country: 'US', 'FR', 'CA', 'GB', 'ES', 'IT', 'PT', 'NL', 'DE', 'AU', 'NDF' (no destination found), and 'other'. Here 'NDF' i.e., destination not defined is different from 'other' since it represents that a booking was made to some country which is not in the given list.

The data on which our model is being trained is set back to year 2010 and the test set consists of users joining after 01/04/2014, while the session data dates back to 2014.

The dataset contains 4 files:

'Age_gender_bkts.csv': It has 420 data points with each datapoint containing 5 columns which tell us about the age, gender, and country destination.

'Countries.csv': It has 10 data points with 7 columns each. These 10 data points are the destination country, and the dataset tells us about the geographic location and language spoken in that country.

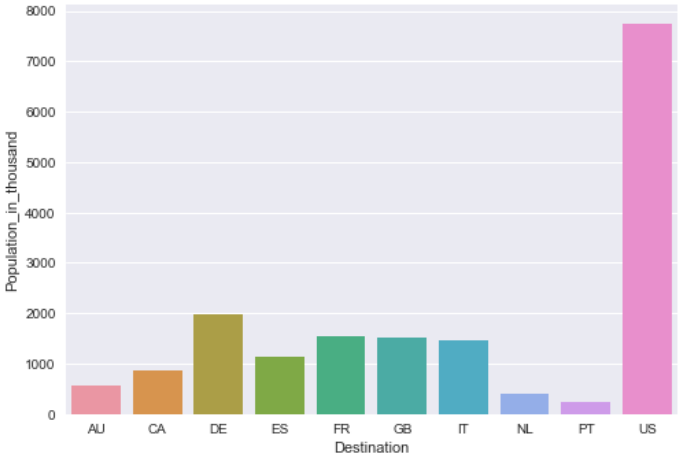
'Sessions.csv': There are 10567737 data points in the data set with each datapoint containing 6 columns. These columns contain the information about the user session for e.g. time spent by the user, device type of the user.

'Train.csv': It has 170137 data points where each datapoint contains 16 columns. These columns are mainly the date of account created, time of first active, age, gender, sign up method, browser, device etc.

II. OBSERVATIONS

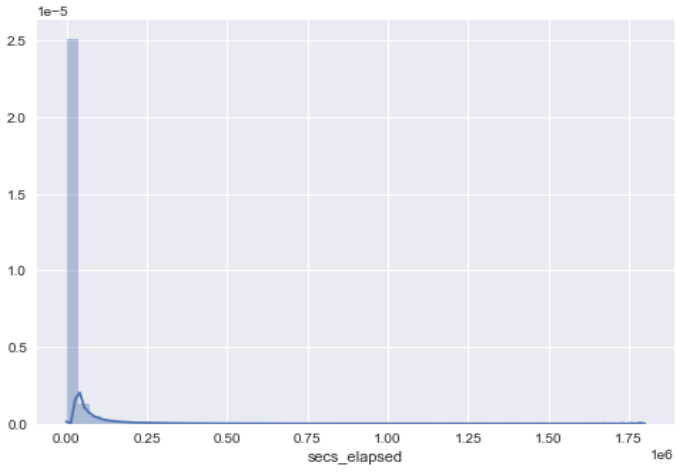
Before we proceed with pre-processing and feature extraction it is important to know the distribution of data and its features.

1) Age, Gender and Population Data:



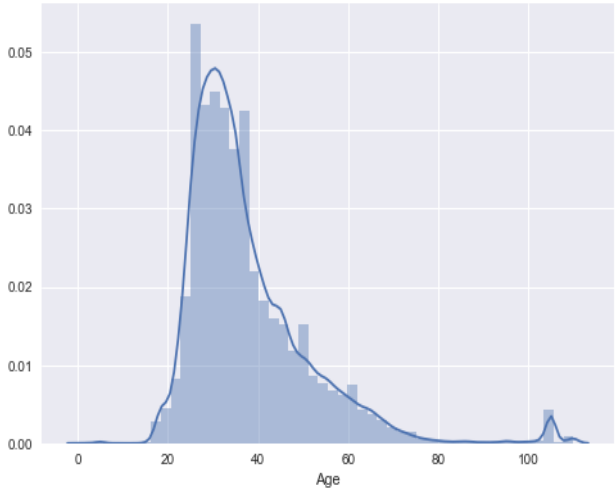
As we can see from the above graph majority of the population travels to USA.

2) Sessions Data:



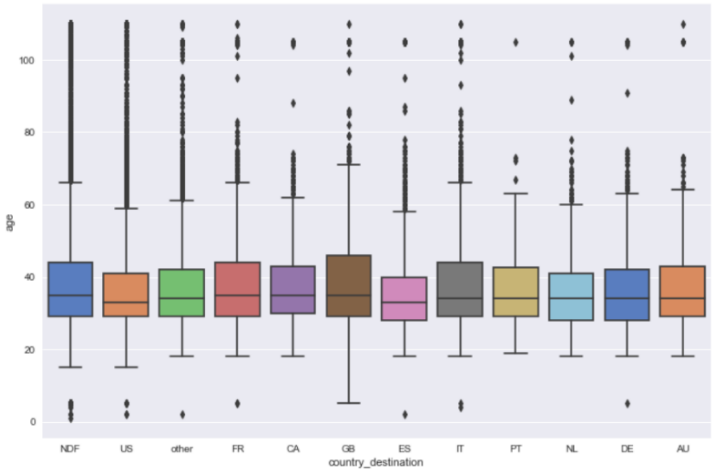
The above plot represents frequency plot of sec_elapsed. We can clearly see that it is highly skewed data, hence we interpolate null values with median rather than mean. After interpolating null values mean changes from 19450 to 19170 seconds and median remains the same around 1147 seconds

3) Train Data:

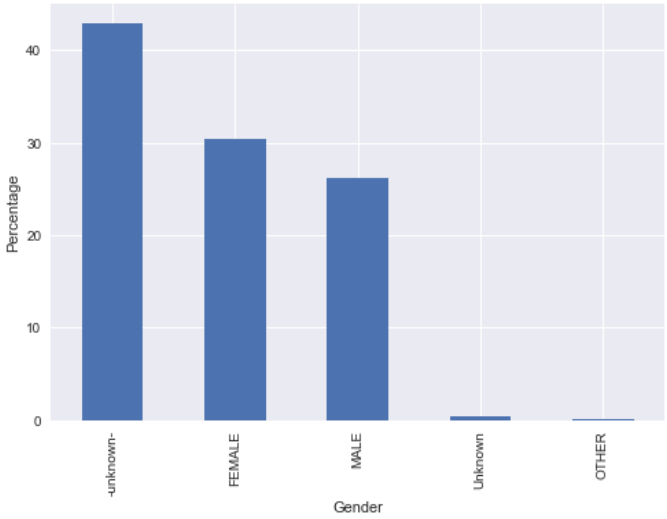


The above plot represents Age frequency distribution. As we can see common age of travelers is between 25 and 45 years.

Then we plot country destination percentage for users who have provided their age and for those where age is missing. We see intresting fact only 25% of users with missing ages book while 55% of users with age provided book. It could be that taking the step of providing age is a sign a user is more serious about making a booking. We discover that people who have not disclosed their ages are least likely to book an Airbnb.



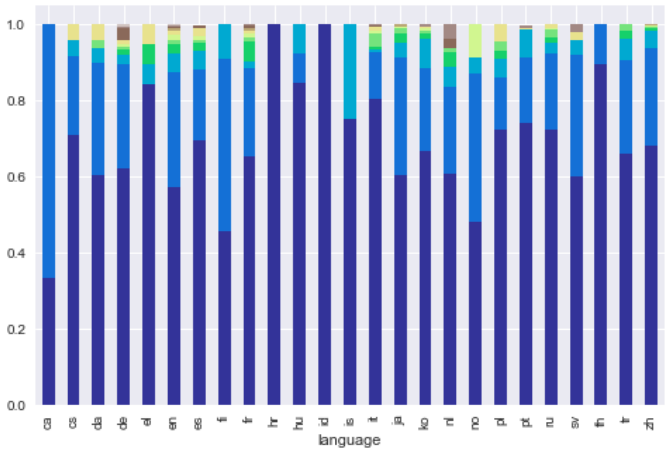
From the above box plot for age vs country_destination, we get to know users who book their trips to Spain and Portugal tend to be younger while those that books trips to Great Britain tends to be older.



From the above plot of gender count in percentage. We see that majority of users have not provided their gender. Out of the users whose gender is known, there are more females than males. This can happen if there are more female users than male users or female are more likely to provide their gender than male.

Then we have plot stacked bar plot for gender and came to known that people who have not marked their gender are less likely to book an Airbnb. Also, people who have marked themselves as 'other' are more likely to make a booking. While female and male ratio of booking is almost similar.

The stacked bar chart extends the standard bar chart from looking at numeric values across one categorical variable to two. Each bar in a standard bar chart is divided into a number of sub-bars stacked end to end, each one corresponding to a level of the second categorical variable. We have created a function to plot such stacked bar chart.



From the above stacked bar plot for language, we can see users who speak Catalan most likely to book, while users who speak Croatian or Indonesian have not booked at all. Also, from the counts plot of languages we infer than 98% of the users speaks English, now we can relate this, that why most of the users prefers USA for their destination.

For the remaining categorical features: signup_method, signup_app, signup_flow , affiliate_channel, first_device_type, affiliate_provider, first_browser. We plot both counts in percentage and stack bar plot and infer the following observations:

i) Users with Basic signup method are little bit more likely to book an Airbnb than users with Facebook signup method.

ii) Users using Web as signup app are most likely to book whereas Android users are least likely to book.

iii) Users with 3 signup flow are most likely to book a trip whereas users with 4 signup flow have never booked any Airbnb.

iv) The Direct Channel has the most number of conversions to bookings whereas the Content Channel has the least.

v) Direct and Google are the most popular affiliate provider among the users.

vi) Among all the users there are 44% Mac Desktop users and 35% Windows Desktop users.

vii) Users with Mac or Desktop are most likely to book Airbnb and users with Others/Unknown are least likely to book.

viii) Chrome, Safari and Firefox are the most popular first browsers among the users.

These observations contribute to the understanding of our model and how we proceed to feature extraction of the dataset. We can infer from these which features contribute maximum towards the result and of these features what parts are majorly influencing the result and which parts are not worth keeping and removing these errors from data and overall gives a view how to proceed with our important features and encodings.

III. PRE-PROCESSING AND FEATURE EXTRACTION

To train our model we have to pass data as numeric form which means we have to convert the raw data into more sophisticated numeric data.

We have done the following pre-processing on the given train and test data:

i) From date_account_created, time_first_active and date_first_booking we extracted day, month, year and dayofweek and make new columns for the each split.

ii) Then we create a new feature called 'diff_in_days' which is the difference between date_account_created and time_first_active.

iii) "Has_booked" is another feature extracted for the date_first_booking. It tells whether the user has previously made some booking in the past.

iv) The age of the user provided in the dataset has a lot of outliers. Ideally the age should be between 10-100 but we have values given to us in thousands E.g., 2000. The reason for this is that the user may have misinterpreted age as birth year and so values about the age in the dataset are not their ages but their birth year. From these values the user's age is calculated and added to the dataset. And another variable "age_group" is created which based on the age.

The categorical data present in the data can be encoded using various encoding techniques like frequency, label, one-hot encoding. We have choose to label encoding for all the categorical data present in the dataset.

With these basic feature engineering and pre-processing and training a XGBClassifier with best hyperparameters we got a descent score of 0.92183 on public leaderboard. But then we extracted some advance feature from the session.csv which helps us break ties in score board and get us on top of the leaderboard with 0.92206 score.

For the advance feature extraction, we first created 4 pivot tables: action, action_type, action_detail and device_type. Each of these tables contains counts corresponds to action, action_type, action_detail and device_type respectively for each user available in session.csv.

Action and action_detail have 360 and 156 columns respectively, if we directly concatenate these 4 tables with our main data it tends to over fit and make our model accuracy even worse. So we decided to extract only those features which increase our model accuracy. Our main idea is to take the top 10 most important features from each of this pivot tables and used a forward selection

technique i.e. we add 1 feature with main data, train the model and checks if it increases the score if it does then we add it to the main list. We repeat this process for all top 10 features for all the 4 pivot tables. And get the best features which will help us improve our accuracy.

Finally got below 8 features as best fit to our model after applying forward selection to 4 pivot tables:

action_type = ['booking_response', '-unknown-']
action = ['requested', 'confirm_email', 'update', 'cancellation_policies']
action_detail = ['pending', 'at_checkpoint']

IV. Model Selection

We have use multi-class classifiers approach in all our models. Firstly we have created a helper NDCG function which takes model, x_test and y_test as input parameters and returns corresponding ndcg score. We have first trained our data with 5 model with basic hyperparameters.

Model	Precision score	NDCG score
LogisticRegression	0.572	0.7802
Decision Tree	0.872	0.9144
Random Forest	0.873	0.8999
LGBMClassifier	0.872	0.9137
XGBClassifier	0.873	0.9150

As we can see from the above table LGBMClassifier and XGBClassifier are the best performing model with near same precision score, but as we can see ndcg score for the for XGBClassifier is higher than LGBMClassifier, here our helper function comes handy in distinguishing between good and best model. So we have use XGBClassifier as final model.

To find best set of hyperparameters for XGBClassifier, we have made our custom Grid search which compares model with our custom scoring metric 'NDGC Function'.

We have tried ensemble techniques with previous 5 mentioned models. Stacking and Blending didn't give promising results as these techniques generally require hundreds and thousands of models to work with for achieving extraordinary scores. Hence, rather than wasting out time to work with such complex classifiers and tunning them we stick with single XGBClassifier and try making model as best as it can. Finally we got our best score of 0.92209 in public leaderboard and 0.92281 in private leaderboard.

V. Conclusion

We would like to conclude that we were able to come up with an efficient model to predict new user destination. We at last settled with XGBClassifier which shows comparatively better results then all the other models and ensemble technique we tried.

VI. Acknowledgment

We were delighted to work under the guidance of Professor G Srinivas Raghavan and Professor Neelam Sinha.

This was our first machine learning project. We have always been impressed by machine learning as a concept but at same time we were also having apprehension and fear of unknown. This is where we would like to acknowledge role of our Profs and TAs to act as a mentor and help us understand this great subject in very simple terms. The facts

brought over by them are our true learning and shall always stay with us. The best part of this project was that the entire learning happened in a competitive but at the same time in a friendly environment. All the 13 teams were competing with each other, but the underlying spirit was to gain knowledge. The teaching assistant Tanmay Jain introduced a good concept of having healthy interaction among all the teams and had monthly interactions with us to guide us with different approaches and keep us on track.

Once again we would like to thank our Professors and Teaching Assistants for giving us this opportunity.

PROJECT LINK

Our project is available at below given link:

https://drive.google.com/file/d/1XQCzVbvLfGMKqtQdEkrRbiJ64-nPK9ya/view?usp=sharing

REFERENCES

[1] Towards Data Science: https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d

[2] Analytics Vidhya: https://www.analyticsvidhya.com/blog/2017/06/which-algorithm-takes-the-crown-light-gbm-vs-xgboost/

[3] Seaborn: https://seaborn.pydata.org/tutorial/distributions.html

[4] Machine Learning Mastery:

https://machinelearningmastery.com/stacking-ensemble-for-deep-learning-neural-networks/

[5] Machine Learning Mastery:

https://machinelearningmastery.com/blending-ensemble-machine-learning-with-python/

[6] Analytics Vidhya: https://www.analyticsvidhya.com/blog/2020/08/types-of-categorical-data-encoding/