IIITB ML Project: Airbnb New User Bookings
Team: Devils_Triangle

Ameya Kurme
IMT2018007
ameya.kurme@iiitb.org

Manan Bansal
IMT2018039
manan.bansal@iiitb.org

Avik Bhatnagar
IMT2018505
avik.bhatnagar@iiitb.org

Abstract- **As an alternative to the conventional lodges, many people choose to use airbnb to travel to different places and stay in one of the less "conventional" places and truly make their travel experience one of a kind. In this project we had been given data of thousands of users who had made their first booking on the airbnb site. All the details about the user like their age, gender and details about the country they visited for the first time were given to us. We had to understand the data and predict on the basis of the user's details what top three country destinations of theirs will be.**

Keywords - Data Preprocessing, Feature Extraction, GridSearchCV, Logistic Regression, One vs Rest classifier, Naive Bayes, SGD Classifier, XGBoost Classifier, CatBoost Classifier, Random Forest Classifier, Voting, Stacking

I. INTRODUCTION

Airbnb is one of the best options for a traveler to save their expenses on hotels. Airbnb is a trusted marketplace that allows people to enlist, explore, and book some unique places all over the world. New users on Airbnb can book a place to stay in 34,000+ cities across 190+ countries. The users can choose their travel destination from a list of countries. Airbnb is available to use both as a mobile phone app and also on the web browser.

By using these predictions, Airbnb is able to decrease the time required by a particular user to book their destination by giving accurate probable destinations which also makes the experience of the user on airbnb better. We used various different machine learning algorithms to predict the country destination based on the details of the user. We used a variety of algorithms like random forest and many gradient boosting methods as well.

The rest of the report is as follows:

Section 1 is on the work related to the predictions of destination for users. Section 2 describes our dataset and different attributes present in our data. Section 3 describes our observations and visualizations. Section 4 is about feature engineering. Section 5 describes our logic and factors which led to model selection and different models that we used. Section 6 is about the conclusion that we drew from our experience in training all the models and what could be the scope of this work in any future avenues.

II. RELATED WORK

Using the enormous dataset given by Airbnb to Kaggle, we can check how effectively to augment user experience and rectify total bookings by utilizing big dataset remains a question to answer for us. That question can be resolved after going through some research papers. In [1], the author Narayanan Ramamirtham has concluded that the Random Forest classifier algorithm is superior to the Decision Tree model. In [1], the Random Forest Classifier model had an 87% accuracy compared to 79% for the decision tree model. Also, he had expected that pruning the dataset given to the Random Forest Classifier will improve the results. This proves that the classifier will not over-fit the data since it generates multiple decision trees and chooses the best one, even if we had a redundant feature, there won't be any degradation in the prediction accuracy. In [2],Tianqi Chen and Carlos Guestrin provide greater insights about XGBoost and Gradient Boosting Machines (GBMs), where both are ensemble tree methods that apply the principle of boosting weak learners using the gradient descent architecture. However, XGBoost improves upon the base GBM framework through systems optimization and algorithmic enhancements.

III. DATASET

We have got datasets of Airbnb from the IIITB Kaggle competition. We were given 5 datasets to help with the prediction of travel destinations as a list of 3 top most visited countries to be traveled by first time users in Airbnb.

(1) train: This dataset contains data on 170,137 users of Airbnb. It contains user id, date account created, date first booking, gender, age, language, country destination etc.

(2) test: This dataset contains 43314 users on which the prediction will be performed. It contains the same data fields as train.

(3) sessions: This dataset contains device type, action, action type, user id and the number of seconds between actions were recorded etc.

(4)age_gender_bkts:This dataset contains user's decided country information in groups age of 5 years difference i.e. range of age with 5 years gap in between, travel destination as a country and it shows information about each group's gender. It comprises age bucket,country destination, gender, Population in thousands, and year with a total of 420 entries.

(5) Countries: This dataset contains a summary of the different country destinations and various data on those countries' locations and which common language is spoken in the relative country.

In these datasets, we have twelve unique countries (CA, DE, AU, FR, IT, GB, PT, US, NL, ES, NDF, other). Here, NDF means no destination has been found which means users have not been at any travel destination yet. We found that in the train data the country destination distribution was as follows 57% No Destination Found (NDF) and 30% US. It was an interesting and challenging part that users have browsed destinations but have not made any actual trip.

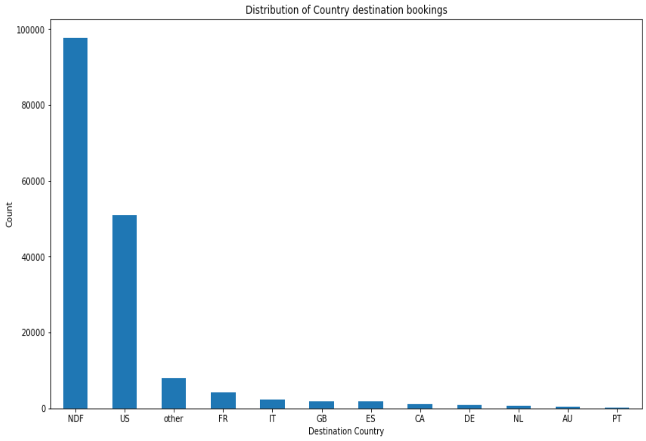
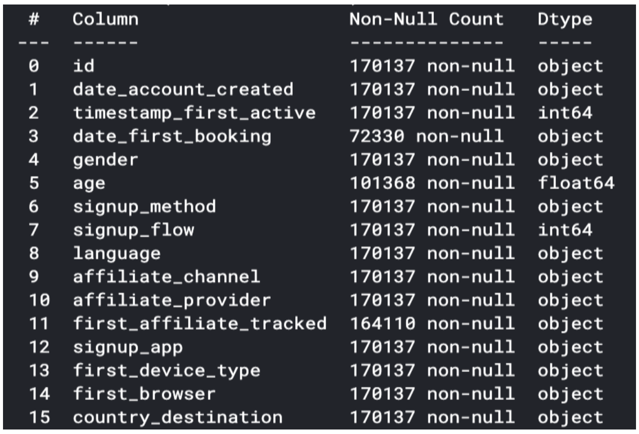


Figure 1: Cuntry destination distribution

IV. OBSERVATIONS

Let's look into how the Data is and the relations and the conclusions we can draw from the dataset and data files given –

1)Firstly when we look at the data in the train datafile, we can see a total of 16 columns including the user id column and the country destination column. Excluding these two columns, there are various types of columns like categorical columns and numerical columns. Even in numerical columns some are like age and signup flow which are just numbers but others are dates like date_first_booking or date_account_created. The column timestamp_first_active gives us the timestamp when the user was active accurately to the second.



```
 #   Column                   Non-Null Count   Dtype
---  ------                   --------------   -----
 0   id                       170137 non-null  object
 1   date_account_created     170137 non-null  object
 2   timestamp_first_active   170137 non-null  int64
 3   date_first_booking       72330 non-null   object
 4   gender                   170137 non-null  object
 5   age                      101368 non-null  float64
 6   signup_method            170137 non-null  object
 7   signup_flow              170137 non-null  int64
 8   language                 170137 non-null  object
 9   affiliate_channel        170137 non-null  object
 10  affiliate_provider       170137 non-null  object
 11  first_affiliate_tracked  164110 non-null  object
 12  signup_app               170137 non-null  object
 13  first_device_type        170137 non-null  object
 14  first_browser            170137 non-null  object
 15  country_destination      170137 non-null  object
```

Figure 2: Training dataset info

2) Relation between the categorical columns and the target columns -

There are a total of 8 categorical columns in the train datafile. One of them is the gender column.

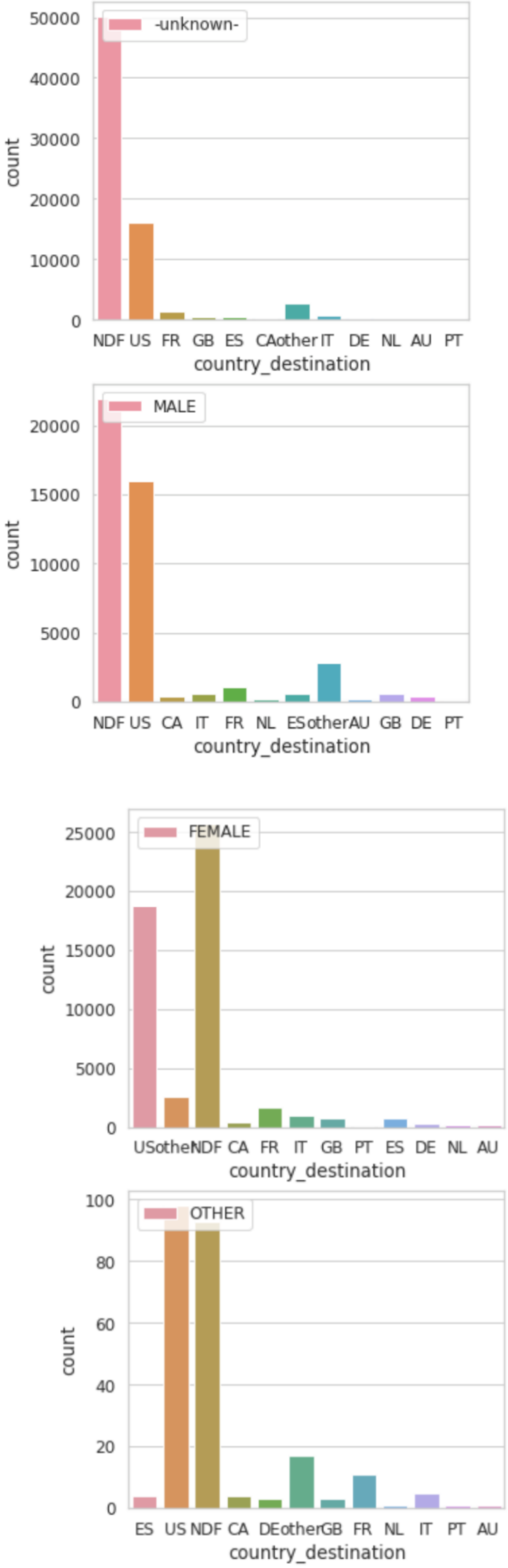


Figure 3: Country destination count based on gender

Looking at these plots we can see that the distribution is different for each category in the gender column. Similarly, we drew a plot for each category in the categorical columns against the target column and then decided whether to keep or drop the attribute or to make any changes.

3) Relation between the numerical columns and the target columns -

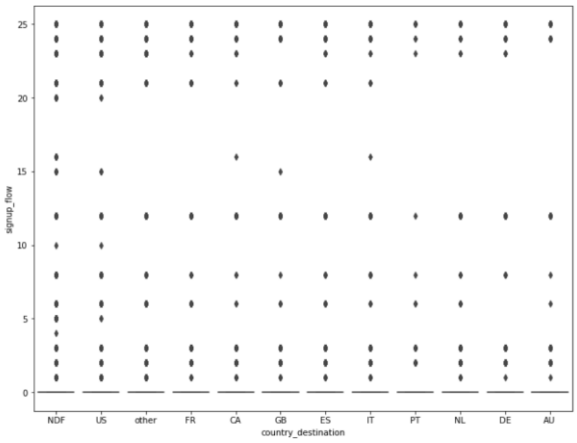


Figure 4: Signup_flow vs Country_Destination

Considering the age and signup_flow columns, plots are drawn against the target column.

For the other numerical columns like date_first_booking, the columns were dissected into day, month and year. Like for the date_first_booking column, new columns are made namely dfb_day, dfb_month and dfb_year, and then again plotted against the target column.

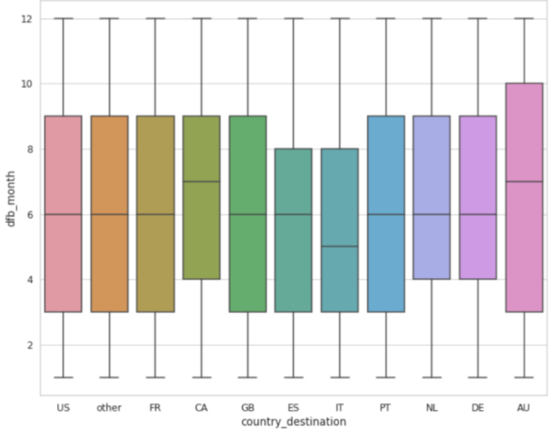


Figure 5: Dfb_month v/s Target column

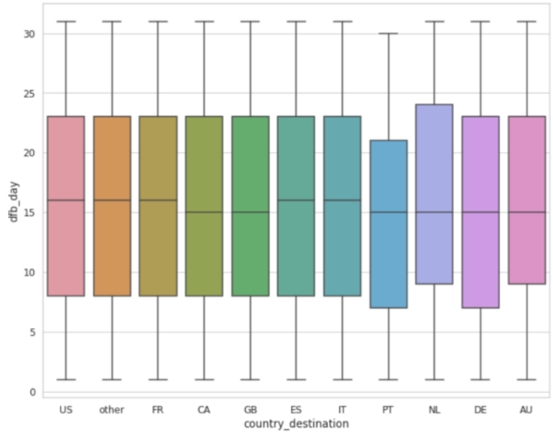


Figure 6: Dfb_day v/s Target column

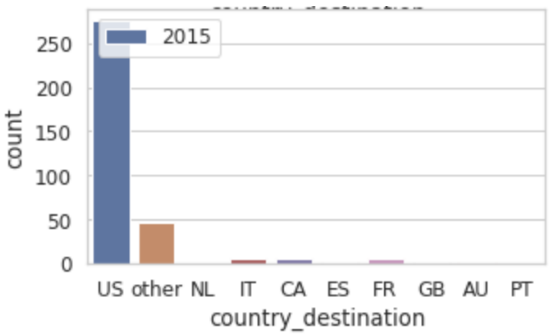


Figure 7: Dfb_year = 2015 v/s Target column

Count of the instances of occurrences of each category in the country_destination column for the category "2015" in dfb_day column. The distribution is then compared for each of the countries and if it is same then the column is dropped.

5) Correlation Matrix-

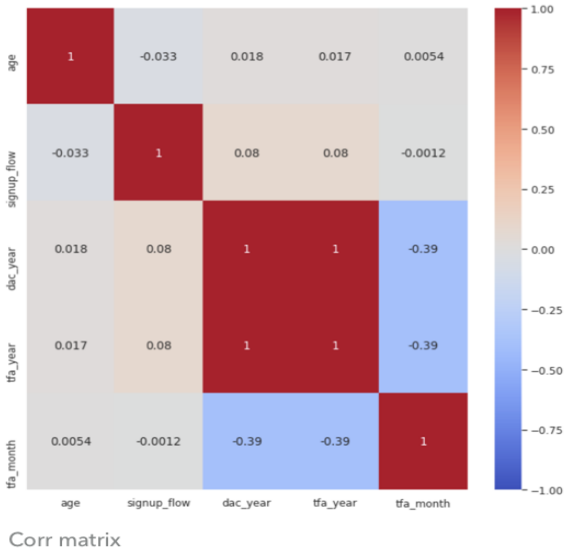


Figure 8:Correlation Heatmap

The columns tfa_year and dac_year have 1 as their correlation value which means both of them have the same magnitude. So that means that one of them can be dropped.

6)Number of null values for the date_first_booking column was the same as the number of times NDF was predicted in the country destination column. After removing all the null values from the date_first_booking column, there were no more NDF predictions in the target column left. Which led to the conclusion that a null value in the date_first_booking column meant NDF prediction.

V. DATA PREPROCESSING AND FEATURE EXTRACTION

Based on our observation regarding NDF prediction we decided on dropping the rows in the training data where the date_first_booking was null.

In the categorical columns, the "first_affiliate_tracked" column had null values which we replaced by "null". This created an additional category in the "first_affiliate_tracked".

For the age column, we replaced the outliers by null values. Values for age less than 18 or greater than 100 were considered to be outliers. The null values in the column were replaced by -1.

The columns "date_account_created", "timestamp_first_active", "date_first_booking" were dissected into days, months and years like mentioned in the observations. And then these 3 parent columns were dropped. After plotting each of the newly made columns with the target column, and analyzing their relation, the following columns were dropped - "dac_day", "dac_month", "tfa_day", "dfb_day", "dfb_month", "dfb_year". Based on the observations from the correlation matrix, "tfa_year" was dropped.

The id column was stored for the test data and dropped separately for both train and test data. All the columns except country_destination , i.e., the target column were one-hot encoded. The country_destination column was label encoded.

For every row in the test data which had the value in the "date_first_booking" column null, the three most probable predictions were exclusively predicted as NDF.

VI. MODEL SELECTION

In the initial stages of our submissions we focused upon logistic regression as our base model which provided us with the leaderboard score of 0.67726. We learnt that logistic regression was a go to model for binary classification and since we had a multiclass classification problem we moved on to the next set of classifier models like random forest, svm, onevsrest classifier etc.

Out of these the random forest classifier gave us the best score of 0.79790. When proceeding to the hyperparameter tuning of the random forest model using GridSearchCV turned out to be a very time and memory intensive task. We then decided to tune the n_estimators and max_depth which boosted our score to 0.92183 and an accuracy of 70%.

To tackle the problem of imbalance dataset we tried oversampling (SMOTE) and undersampling techniques, however these resulted in an even worse classification and score.Some of the models also had the hyperparameter of class_weights so as to tackle the class imbalance issue but the reduced score forced us to remove this too. To tackle this issue we decided the elementary approach to split the train data with the stratify hyper-parameter so as to get a better evaluation and understanding on our models and the obtained metrics (accuracy, precision, recall and f1 score).

The models Naive bayes, SVM and SGD classifier provided us with a very generic classification of an equal number of predictions for the country destinations US, other and France. Keeping aside the NDF that we predicted using the date first booking we saw these results as inappropriate and we didn't pursue them further.

While testing out these models we also came to know that when the models provided an accuracy in the range of 75 to 85% the model overfitted and we obtained a reduced leaderboard score .

One vs rest Classifier gave us the most overfitted model.

We then pursued the gradient Boosting Classifier models.Firstly, XGB classifier model which is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework[3].The model provided a score of 0.92183.We then came across hyperparameter tuning techniques by reading Analytics Vidhya Article [4] to obtain a better score of 0.92185.

The Catboost Classifier for our multiclass classification gave a relatively overfitted model as observed by the submission file country destination value counts and reduced score of 0.92137.

Then we came across the Light GBM, which is a gradient boosting framework that uses a tree based learning algorithm.It can handle the large size of data and takes lower memory and time to run. Another reason why Light GBM is popular is because it focuses on accuracy of results [4]. When hyperparameter tuned this model placed us on the top of the leaderboard with a score of 0.92188.

Having tried out various models and the best accuracy score of these individual models got to a saturation point of around
70.3% and a best score of 0.9188. Then we looked into various ensembling techniques [6] like VotingClassifier[7] and StackingClassifier[8].

We have finalised the three models to try out the ensembling techniques: LGBMClassifier,
XGBClassifier and RandomForestClassifier which gave the best results of 0.92188, 0.92185 and 0.92183 respectively.

We implemented the stacking classifier with initial estimators as random forest and xgb and a final estimator as the LGBMClassifier along with a 10-fold cross validation so as to create our meta model. Having tried all the variations in this ensembling technique we were still unable to boost our score.

We then moved on to the VotingClassifier with our aforementioned models and tried assigning the best weights to these models so as to obtain the best results. This Boosted our score significantly to 0.92198 placing us on the top of the leaderboard once again at that time.

The Kaggle competition used Normalized
Discounted Cumulative Gain (NDCG) evaluation method, that is used for giving rank to the most visited countries.[9]

VII. CONCLUSION

We would like to conclude that we were able to come up with an efficient model for the country destination predictions for a new user. Which can be further used to provide a time-efficient and a better booking experience by giving accurate probable destinations for all the Airbnb users.

VIII. CHALLENGES AND FUTURE SCOPE

The main challenges are to deal with the Inconsistencies in data as most of the users prefer guest sessions resulting in some missing values. Development and implementation of better Machine learning models to tackle the imbalanced dataset caused by the majority preference to visit a set of particular countries also needs to be tackled so as to get a better prediction in regard to the minority counts countries.

Since, we know which destination countries are more popular with the users, Airbnb can implement targeted marketing. Airbnb can plan ahead which countries they should scout more to get accommodation-providers onboard as they can clearly see the users inclination to visit those countries. In regards to future bookings depending on the choice of the destination country of a particular user, Airbnb can possibly think of similar destination countries (in terms of climate, topography, choices of recreation etc.) to offer as other viable travel options to that user.

ACKNOWLEDGMENT

We would like to thank Professor G. Srinivas Raghavan, Professor Neelam Sinha and our Machine Learning Teaching Assistants for providing us with the opportunity to work on this project. They were a great help whenever we got stuck somewhere and needed some guidance.They gave us invaluable insights in the tutorial sessions and we are ever grateful to them for that.

We had a great learning experience while working on this project. This competition helped us up into reading more and more articles which gave us some really cool ideas and improved concepts for implementing in our project.

The Leader board was a great motivation to push ourselves into trying out more machine learning concepts and techniques so as to flourish in the highly competitive environment established by our fellow competing teams.

PROJECT LINK

Our project files are available at: https://github.com/Avik1007/ML-Project-Airbnb-New-user-Bookings.git

REFERENCES:

[1] Srinivas Avireddy, Sathya Narayanan Ramamirtham, Sridhar Srinivasa Subramanian “Predicting Airbnb user destination using user demographic and session information”, UC San Diego https://cseweb.ucsd.edu/classes/wi15/cse255-a/reports/fa15/040.pdf

[2] XGBoost algorithm was developed as a research project at the University of Washington. Tianqi Chen and Carlos Guestrin presented their paper at SIGKDD Conference in 2016 and caught the Machine Learning world by fire. http://dmlc.cs.washington.edu/data/pdf/XGBoostArxiv.pdf

[3] https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d

[4] https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python

[5] https://medium.com/@pushkarmandot/https-medium-com-pushkarmandot-what-is-lightgbm-how-to-implement-it-how-to-fine-tune-the-parameters-60347819b7fc

[6] https://github.com/frenzytejask98/ML_TA_IIITB_2020/blob/master/November_27/Ensemble_Techniques.ipynb

[7] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.VotingClassifier.html

[8] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.StackingClassifier.html

[9] https://www.kaggle.com/c/airbnb-new-user/overview

[10] https://www.kaggle.com/krutarthhd/airbnb-eda-and-xgboost