INFO 254

DATA MINING AND ANALYTICS

Final Project

Airbnb Destination Country Prediction

I-Chen Lee

Xinyi Li

Yingxin Chen

Erica Chen

Qiaowen Guo

May 9, 2019

Table of Contents

1. Introduction

(Qiaowen/Yingxin/Erica)

In this project, we are going to answer an important question: what are the primary countries that new users who use Airbnb to book accommodation view as destinations.

Nowadays, there are increasing people to prefer booking accommodation at Airbnb rather than at hotels when they go out for traveling. The obvious answer why Airbnb is so popular is because local demand exists. Airbnb provides people with more chances to experience local life, cozy accommodation experience, and flexibility to decide their trip ways. And Airbnb reservation prices are often much lower than the local rates that a traditional hotel to keep its books balanced.

However, the problem is that, for the new users, sometimes it is a little hard to start booking Airbnb houses if they always book hotels before. At the beginning of our project, we checked the complaint posts on the Airbnb community to summarize the users' thoughts. There are many posts in the community which indeed complain that the website is slow and not very friendly to new users. What's more, many posts mention that usually, recommendations on the Airbnb website do not match their interests, which make them confused when logging in the website first. After several attempts, some new users probably give up and turn their attention on other booking websites, such as booking.com, hotels.com, which also take some strategies now to attracts users.

In this way, how to make accurate recommendations for new users is one of the most critical parts Airbnb should consider. The huge advantages of Airbnb are remarkable, while how to guide new users to start the first experience is also of significance. Apparently, it will be much more user-friendly if Airbnb can accurately predict where new users will book as their primary travel experiences. Hence, users could decrease the average time for the first booking. Airbnb could attract new users to book accommodation there and better forecast demand for accommodation at each location.

Based on the information new users provide and also through some recommendation algorithms, Airbnb could recommend accommodation and experience for new users in a targeted manner. We collected raw dataset from Kaggle "Airbnb New User Booking" competition , leveraged data preprocessing and feature engineering skills to deal with user features, account general information, country information, etc. , applied machine learning methods, such as logistic regression, random forest, decision tree, XGboost, to do classification and predict the most likely first destination of new users. Our final objective is to find the 5 top countries that new users will choose as destinations.

2. Dataset

(Erica/Ichen/Qiaowen/Yingxin/Xinyi)

“age_gender_bkts” : 420 rows & 5 features, summary statistics of users' age group, gender, country of destination;
”age_bucket” : the age group, from 0 year old to 100 years old, every 5 years is a group;
”country_destination” : target variable to predict (12 target countries)
“gender”: female, male, unknown
“population in thousands”: population of each age group of each gender in each country
“year”: year of first travelling
“countries”: 10 rows & 7 features, summary statistics of destination countries
“country_destination”: 12 target destination countries
“lat_destination”: latitude of destination country
“lng_destination”: longitude of destination country
“distance_km”: distance between destination country and US
“destination_km2”: area of the destination country
“destination_language”: language used in destination country, such as french,english, spanish,etc.
“language_levenshtein_distance”: levenshtein distance between the language spoken in destination country and US
“sessions”: 10567737 rows & 6 features, web sessions log for users
“user_id”: identity number of users
“action”: web action of users, such as search_results, lookup, show, index, etc.
“action_type”: type of action, such as click, data, view, etc.
“action_details”: more detailed description of action
“device_type”: device used in the session, such as windows desktop, mac desktop.
“secs_elapsed”: time of the session (in second)
“train_users_2” &“test_users”: 213451 rows & 15 features, 62096 rows & 15 features
“id”: identity number of users
“date_account_created”: the date of account creation, includes year, month, day
“timestamp_first_active”: timestamp of the first activity(it can be earlier than date_account_created or date_first_booking because a user can search before signing up)
“date_first_booking”: date of first booking
“gender”: gender of users,such as female, male and unknown
“age”: age of users
“signup_method”: from facebook or basic
“signup_flow”: the page a user came to signup up from
“language”: international language preference of users
“affiliate_channel”: what kind of paid marketing, such as direct, sem-brand,etc.
“affiliate_provider”: where the marketing is, such as google,facebook, direct,etc.
“first_affiliate_tracked”: what’s the first marketing users interacted with before the signing up
“signup_app”: what kind of system does user use, such as web, ios, or other
“first_device_type”: what kind of device does user use, such as mac, windows, or other
“first_browser”: what kind of browser does user use, such as chrome, safari or other

3. Introduction to Approaches

We did not separate pre-processing, EDA, feature engineering, and modeling to exactly different members. Instead, we encourage everyone to contribute part of each step. It is because that we believe everyone should be equipped with end-to-end data analytical skills.

1. **Pre-processing**
   - “train_users_2” &“test_users**”** (Xinyi/Yingxin/Qiaowen)
2. **Explorational Data Analytics**
   - Feature Distribution (Yingxin)
   - Feature Correlation Matrix (Erica/Qiaowen)
   - Descriptional Statistics of datasets (Erica)
3. **Feature Engineering**
   - “sessions” (Ichen)
   - “age_gender_bkts” (Ichen)
   - “countries “ (Ichen)
   - Concatenate dataframes ((Ichen)
4. **Analytics Models**
   - Random Forest (Erica/Qiaowen/Xinyi)
   - Multinomial Logistic Regression (Xinyi)
   - Decision Tree(Yingxin/Qiaowen)
   - XGBoost(Yingxin/Xinyi/Qiaowen)
   - Ensemble Models: max-vote (Xinyi)
5. **Presentation** (Erica/Ichen/Qiaowen/Yingxin/Xinyi)
6. **Final report**
   - Finish individual part in Individual Work (Erica/Ichen/Qiaowen/Yingxin/Xinyi)
   - Introduction (Qiaowen/Yingxin)
   - Dataset (Qiaowen/Yingxin)
   - Conclusion-Real World Implications (Qiaowen)
   - Image and table sequence coding (Qiaowen)
   - Summarize and revise the whole report (Qiaowen)

4. Individual Work

4.1. Erica

In this project, I participated in explorational data analytics, data preprocessing, feature engineering and modeling.

For explorational data analytics, I draw the correlation matrix to get a basic picture of the significant positive/negative correlation between variables; I also use feature importance from baseline random forest model to try to know which variables are important to target label.

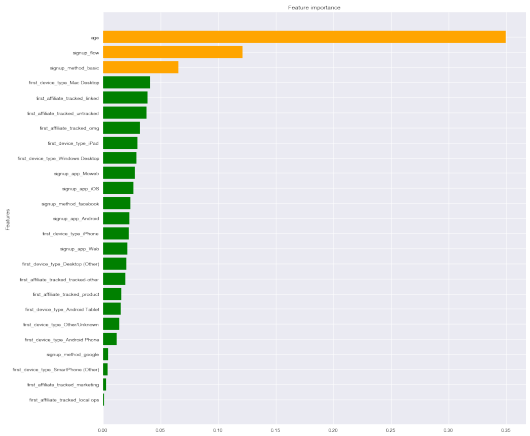

Figure 1 Feature importance from baseline random forest model

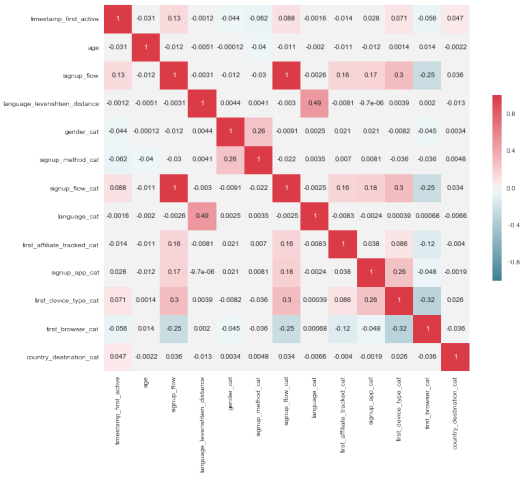

Figure 2 Correlation matrix

Besides, I draw the age-destination distribution box plot to observe if there are outliers or unreasonable values, and we found that there are 750 user's age are over 2014.

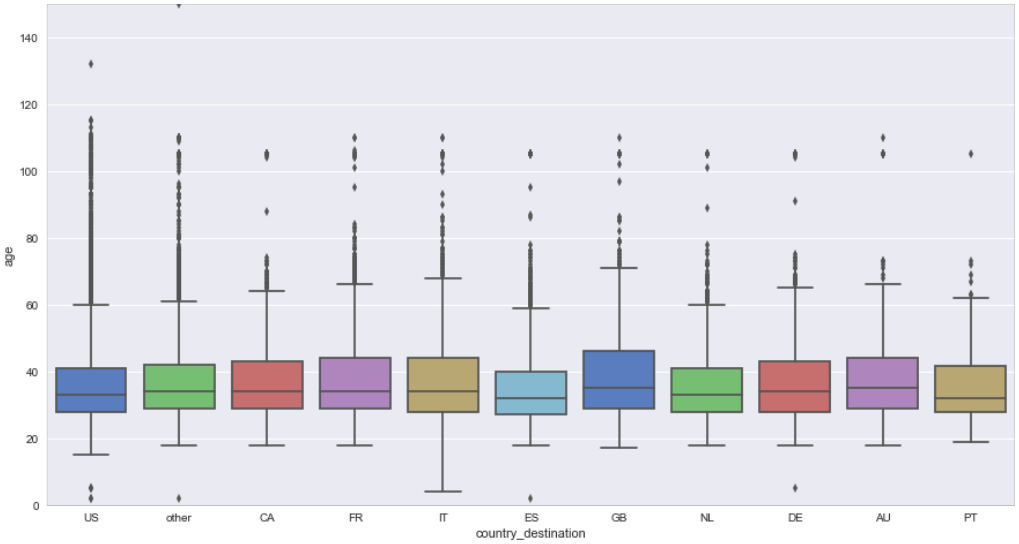

Figure 3 Age-destination distribution box plot

For the data preprocessing part, I first convert age over 100 as NaN, and create 4 age features (age_mean, age_min, age_max, age_std) instead of original age. For gender, I count both 'Other' and 'unknown' as 'Other'; For date, I split the date variable to year, month and day. For other categorical variable, I use dummy variable to represent them.

In feature engineering part, I merge several dataset to train_users_2 dataset like countries (merge by language) and age_gender_bkts (merge by destination country). Although I get more features such as population, country latitude, longitude, destination area and language distance, etc, the accuracy of the following prediction is not improved. Simultaneously, I use feature importance as reference to do the feature selection.

Therefore, in my final model, I only use the features from train_users_2 and countries, which includes 'signup_flow', 'language_levenshtein_distance', 'signup_method', 'age_mean', 'age_std', 'age_max', 'age_min', 'month_account_create', 'month_first_active', 'month_first_booking', 'gender', and apply grid search method to do cross validation and get hyperparameters. Both random forest and XGboost got around 84% when predicting one country, and 86% when predicting 5 countries.

4.2. Ichen

In the project, I participated in data pre-processing, feature engineering, and metric impact.

For data pre-processing and feature engineering, I generated the session info dataset in account ID level, and country info dataset in country level.

For session info, the purpose is to check the actions each user ID took on Airbnb website and the device they used in each session. For example, an user can do 3 times of "search" and 2 times of "compare", all on iPhone. In order to efficiently and effectively transfer these activities and habits from session level to account level, I divided the pre-processing into three parts:

1. **If all session data to be kept? Will it be helpful?**

I checked if account ID in Session are mostly in Train data & Test data, and found that 100% of IDs in session (135,484 IDs) appeared in either Train data or Test data. Also, 99% of IDs in Test data (61,668 IDs) is with sessions, but only 34.6% of IDs in Train data (73,815 IDs) is with sessions. This means that all the session info should be kept (not dropped for efficiency in calculation), while the feature contribution of "session info" will be limited to one third of data we have. However, it may still be effective since it is close to the number of IDs we want to predict in the Test data.

2. **Do all the actions / devices to be kept? Will there be unnecessary sparse data?**

When grouping session info to account ID-level, I decided to group by ID with action types and device types, and then use pivot table to prepare for future ID-level merge within datasets. However, in order to make sure that the columns kept are informative, I checked if there is fat tail of action types and device types. There indeed exists 360 action types and 14 device types, so I kept only those actions used by more than 10,000 IDs, shrinking the number of action types to 37, and guarantee the popularity of actions to avoid too sparse data (At least 10,000/73815 = 13.5% of all Train IDs vary in each action we kept).

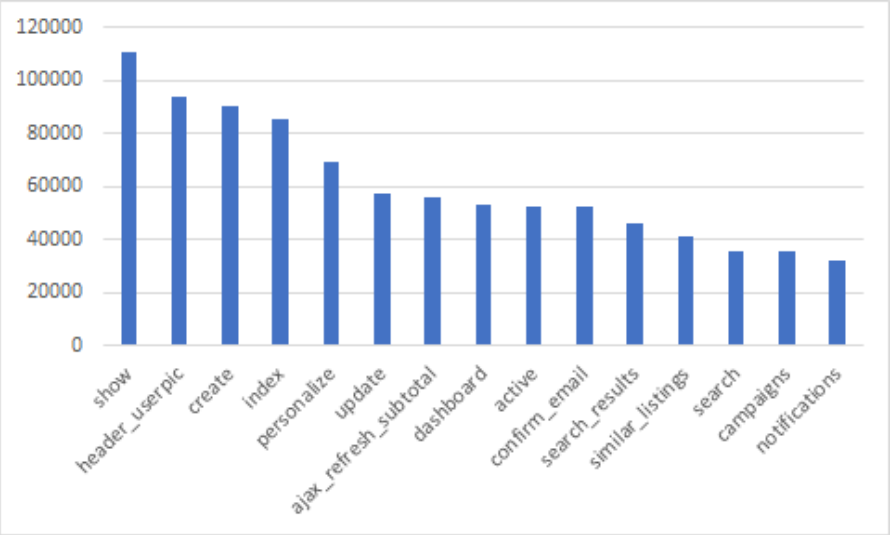

Figure 4 Top 15 actions (user_size)

3. **Besides number of sessions in each action / device, what else to represent the habits of each user?**

I also added the session duration info, which is how long does that account ID stay during that session, to the dataset. It is because I believe the user's habits in each action type / device type may also be informative for prediction. I transferred the duration within sessions per user ID per action (or per device) into their average, minimum, maximum, and standard deviation of time.

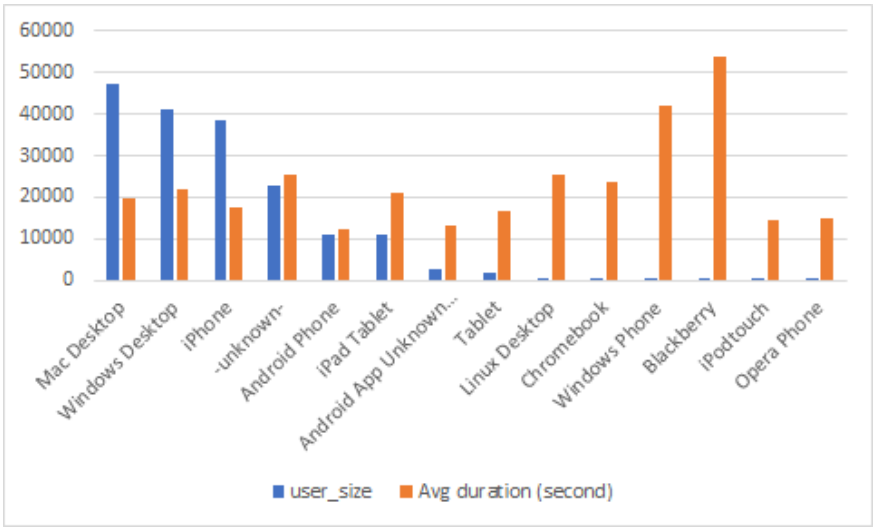

Figure 5 Device Usage

For country info, the purpose is to check if characteristics of each destination country is informative for prediction. However, since we set exactly destination country as our predicted column, we cannot use the correlation trained in Train dataset to Test dataset (because the destination country is blank in Test dataset). Therefore, I used "language set on Airbnb" in both Train and Test dataset to link to "language spoken" in country-level info, which contains:

- The population (age, gender) distribution in each country, and the actual number are transferred to percentage of specific groups within the total population.
- Location, distance from US, area, language distance from English for each country.

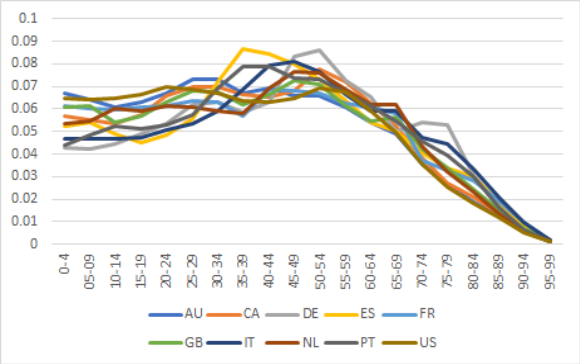

Figure 6 Age Proportion

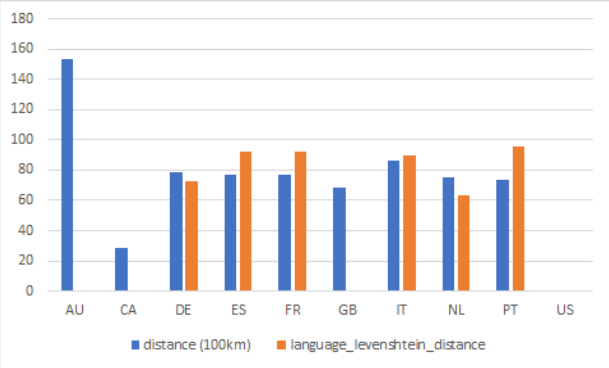

Figure 7 Distance (Location & Language)

However, even though there are four English-speaking countries in our destinations, we decided to take US as the one to match since it is the majority (93.6% out of the four).
For metric impact, since we have the NDCG definition, I compared the situations of final prediction in terms of the priority and number of predictions. I found that only the priority differs for the final score, while the total number of "guess" does not make difference. For example, if in the top-5 countries predicted are Germany, Italy, France, US, UK, and the real answer is Germany, no matter how many countries we predict in total, we will get 1 as the score. In other words, if the real answer is France, we will get 0.5 as the score if we predicted top 3 to 5 countries, while getting 0 as the score if only predicting top 1 to 2 countries. This comparison promised us the confidence of predicting to the maximum length of destinations in total.

4.3. Qiaowen

In this project, I participated in data preprocessing, explorational data analytics, and modeling. Also, I write the introduction, conclusion of the final report, as well as summarize and revise all the individual parts.

In data preprocessing part, I focus on the train&test dataset for three types of data: 1) Numerical data. 2) Categorical data 3) Date data. To start with, I use train.info() to check the missing data in the train dataset. It shows that missing data exists in three features: "date_first_booking" (88908 non-null/ 213451), "age" (125461 non-null/213451), "first_affiliate_tracked" (207386). The number of non-full value in "date_first_booking" is only 88908. In order to find the reason why it is fewer than any other features, I check the number of different country destinations in "country_destination". It turns out that the number of "NDF" is the same with the number of null values in "date_first_booking". So there are 124543 users in train dataset have not start their first booking yet. But Airbnb still view them as one type of country destination after we read the posts in Kaggle. According to Erica, there are some outliers in "age". Hence, I first transform the age greater than 100 into 100, and age less

than 13 into 13. Then I use mean value of age to fill the missing data. For “first_affiliate_tracked”, I replace nan as class called “none”, since this feature is categorical data, it is more appropriate compared with filling missing data. Next, for date data, I first convert “date_account_created” and “date_first_active” to DatetimeIndex. Then I split dates into day, week, month, year. I choose “weekday_account_created”, “month_account_created”, “weekday_first_active” and “month_first_active” as four new features adding to the current feature pool. I drop four features, which are 'timestamp_first_active', 'date_first_booking', 'date_account_created', 'date_first_active' (Note that 'timestamp_first_active' is meaningless). Then I normalize the numerical data and use one-hot to code categorical data, and I obtain the final feature list: 'gender', 'signup_method', 'signup_flow', 'language', 'affiliate_channel', 'affiliate_provider', 'first_affiliate_tracked', 'signup_app', 'first_device_type', 'first_browser' , 'weekday_account_created','month_account_created','month_first_active', ‘age’.

In explorational data analytics part, in order to find the correlation between this features as well as the label, I use the correlation matrix.

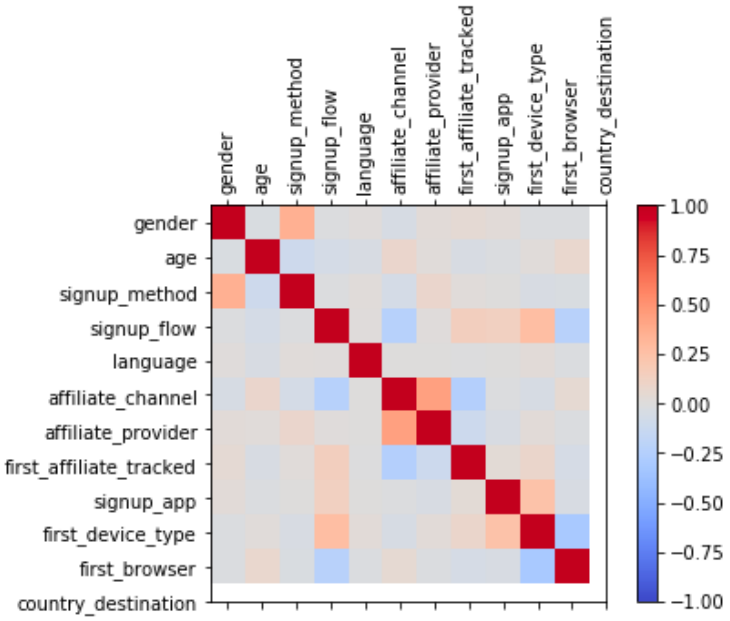

Figure 8 Correlation Matrix

In modeling part, I try two kinds of calculating methods to obtain the accuracy score. Firstly, I want to predict the favorite country of new users. Then I use Decision tree and Xgboost as two machine learning ways to model the data. However, the prediction is really bad since the accuracy for test data is only 0.23230 for Decision tree with max_depth = 10, 0.23464 for Xgboost. As observed in Figure 5 and Figure 6, it is because there exist some specific countries that most users would like to visit, like NDF and US. Hence, for the long-tail users, it is hard for machine learning models to predict what is their favorite country correctly.

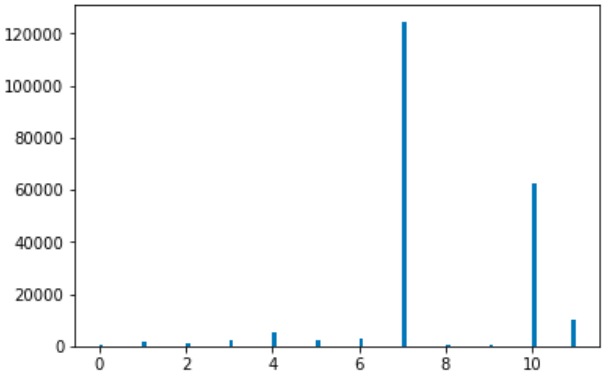

Figure 9 Country destinations histogram

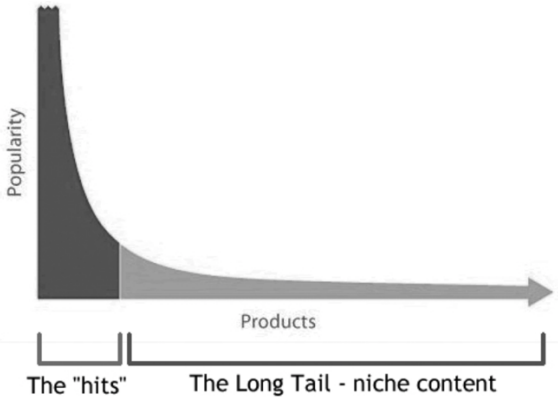

Figure 10 Long tail effect

As the advice of the sponsor in Kaggle, I try to predict the top five country destination for users. Note that if it is to predict the best country, the code is model.predict(), and if it is to predict the top 5 country, the code is model. predict_proba() that is to predict the probabilities of each country destination. Then I use Decision Tree, Random forest and XGboost to predict the probabilities. For Decision Tree with max_depth = 10, the accuracy is 0.86330. For Random Forest with n_jobs=2, n_estimators=50, the accuracy is 0.83567. For XGboost with max_depth=6,eta=0.05,gamma= 0.1, learning_rate=0.1, max_delta_step=0, the accuracy is 0.86506. With respect to the three classic three models, XGboost performs best, but it usually cost much more time than other two models. Consequently, when it comes to turning parameters (especially XGboost has so many parameters), it will be very troublesome. Decision Tree is not a ensemble model, however, its performance is better compared with Random Forest. Even it is not so good as XGboost, its running time is so small that Airbnb could consider using this method to decrease the average booking time. Random Forest's performance is not so good as I thought before, the reason is that it is easy to get into overfitting, even it did perform well in train dataset. The following table summarize the accuracy for each model in two prediction conditions.

Table 1 Prediction accuracy

Model	Favorite country	Top five countries
Decision Tree	0.23230	0.86330
Random Forest	N/A	0.83567
Xgboost	0.23464	0.86506

Besides, in the final report, my major contributes are to summarize and revise all the individual work, including coding each paragraphs, figures and tables; adjusting font size, paragraph space and length to satisfy requirements. I also write the Introduction part with Yingxin as well as real-world Implications in conclusion.

4.4. Xinyi

In this project, I participated in data preprocessing, explorational data analytics, feature engineering and modeling. I tried three different data preprocessing methods to train various models.

My first Data preprocessing is very simple, I only had features -- 'id', 'gender', 'age', and 'language' -- in the training and testing dataset. I combined the training and testing dataset to make it easy to do data preprocessing on both dataset.

I found that there are 'unknown' values in the feature 'gender', then I replace all these 'unknown' values by NA. After doing that, I found that features 'age' and 'gender' have many NA values. I mapped all 'FEMALE' to be 1, and all 'MALE' to be 0, then the gender can be represented as numerics, and I replaced all NA in 'gender' to be the median value of 'gender', which is 1. For 'language', I factorized this feature. Therefore, in my final training dataset, I only had feature 'age', 'gender' and 'language'.

Different from what we did for assignments, in this project, for each user_id, I predicted the probability of 12 different countries to be recommended to the user instead of just predicting only one country. Because based on evaluation metric for this competition, which is NDCG (Normalized discounted cumulative gain) @k(maximum k is 5). NDCG is calculated as:

$$DCG_k = \sum_{i=1}^{k} [(2^{rel_i} - 1)/log_2(i+1)]$$

where rel_i is the relevance of the result at position *i*. The ground truth country is marked with relevance = 1, while the rest have relevance = 0. Therefore, recommending more countries helps to get more scores in this evaluation system. Therefore, for any center user, I will pick the top five country according to the probability to recommend country destination. That means, in my final prediction files, each user corresponds to five countries.

Then I applied my final dataset to train the xgboost model with max_depth = 6, learning_rate = 0.3, n_estimators = 22, objective = 'multi:softprob', subsample = 0.6, colsample_bytree=0.6, seed=0, and get the accuracy, which was close to 85%.

Since in my previous method, my used features are only 'age', 'gender' and 'language' and I got around 85% accuracy, I was thinking of adding more features and doing more complicated data_preprocessing to increase the accuracy.

Therefore, I used the original training and testing dataset, and applied parse_dates function to make features 'timestamp_first_active', 'date_first_booking' and 'date_account_created' to make these date features in a standard format. I dropped the 'country_destination' in the training dataset and saved it as the label for user_ids in the training set. Then I combined the train and test datasets to do data-preprocessing. Since the 'date_first_booking' is a completely missing column in the test set, I dropped this column. I replaced the missing age and missing first_browser to NA. I checked the distribution of 'age' and found that the max age is 2014 which is impossible. To make this feature be more reasonable, I replaced all age, which is greater than 100 and less than 18, to be NA. I splitted 'date_acocunt_created' and

‘timestamp_first_active’ into new columns ‘acc_year’, ‘acc_month’, ‘acc_day’, ‘tfa_year’, ‘tfa_month’, and ‘tfa_day’, and dropped the original ‘date_acount_created’ and ‘timestamp_first_active’ columns.

I did the EDA on ‘gender’ and found that there are four categories and the ‘unknown’ in the majority value. I converted features -- ‘gender’, ‘signup_method’, ‘signup_flow’, ‘language’, ‘affiliate_provider’, ‘affiliate_channel’, ‘first_affiliate_tracked’, ‘signup_app’, ‘first_device_type’, ‘first_browser’ to categorical columns and initialized a MinMax scaler and applied scaler to the numerical features.

After doing the previous data-preprocessing and EDA, I used my dataset to train models and as mentioned in the first method, for each user_id, I picked the top five countries to recommend. Then my xgboost model got the accuracy -- 0.86561. To train the other models, such as Multinomial Logistic Regression and Random Forest, I first replaced all NA values in ‘age’ to be 0, and the accuracy for MLR is 0.85820, the accuracy for Random Forest is 0.82778.

Actually, my last method is only adding more features in the session dataset to my final dataset (after data-preprocessing) in my second method. I outer merged my final dataset and the dataset with session information, which has been preprocessed by I-chen, and I trained the xgboost model on this dataset. My final model accuracy is 0.87729, which is the best accuracy for our project.

Table 2 Prediction Accuracy

Dataset	Model	Accuracy
Train (gender, age, language)	xgBoost	~ 0.85
Train_user.csv	Multinomial Logistic Regression	0.85820
Train_user.csv	Random Forest	0.82778
Train_user.csv	xgBoost	0.86561
Train_user.csv	Max Vote	0.80152
Train_user.csv & Session.csv	xgBoost	0.87729

4.5. Yingxin Chen

In data preprocessing , I mainly focused on cleaning and preprocessing “train_users_2” and “test_users”. For example, I leveraged “pandas.to_datetime” function to transfer “date_account_created”, “timestamp_first_active” into standard form of date and then get year, month, day information from these two features. For “age” feature, I removed weird age values and add missing value with the mean value of “age”. For “language” feature, I replaced language values with low frequency with english. For features with value '-unknown-' , I replaced it with ‘nan’.

In explorational data analytics, I explored distribution of several features. For example, I analyzed the histogram of “age” and found that there are some age values larger than 100 and some age values equal to 0. It is weird because based on realistic experience there is little chance that people who are older than 100 years old or less than 5 years old will book accommodation on Airbnb. And through the histogram of “language”, I find that english is language preference for most of users of Airbnb.

In feature engineering part, I added some new features from “age_gender_bkts” file and “countries” file. First, I leverage “merge” function of pandas to join “age_gender_bkts” and “countries” on “country_destination” features to get a new dataframe “age_gender_country”. And then I left join “train_users_2”,”test_users” with “age_gender_country” on “language” features. After that, I deleted target features. And then, I normalized numeric features with “sklearn.preprocessing.normalizer” transformer and dummy object features with “pandas.get_dummies” .

In modeling part, I applied several machine learning algorithms learned in class to do classification. Before training the model, I split the “train_users” into training data and testing data as what we learned in lab 5(Kaggle competition). Then train different models with training data and applied trained models to testing data. Get accuracy of training data and testing data and make sure that there is no over fitting. Then I trained model with whole data from “train_users” and then applied trained models to “test_users” to predict the probability of 12 target countries to be first destination of each user. Then picked the 5 countries with top 5 probability as the prediction destination countries for each user.

The reason why I picked 5 countries with the top 5 probability rather than picked only 1 countries with the highest probability is: First, based on realistic experience, Airbnb can recommend accommodation of more than one countries to users. And as long as countries we recommend contains the country that the user wants to go, this is a successful recommendation. Second, based on evaluation metric for this competition, which is NDCG (Normalized discounted cumulative gain) @k(maximum k is 5). NDCG is calculated as:

$$DCG_k = \sum_{i=1}^{k} [(2^{rel_i} - 1)/log_2(i+1)]$$

where rel_i is the relevance of the result at position i. The ground truth country is marked with relevance = 1, while the rest have relevance = 0. Therefore, recommending more countries helps to get more scores in this evaluation system.

In logistic regression model, when only picking one country with highest probability, I get 60.32 % accuracy in training dataset and 60.16 % accuracy in testing dataset. And I get score 0.85673 in real test dataset when submit 5 countries with the top 5 probability(the highest one in this competition is 0.88697).

In decision tree model, when max depth equals to 15 and only picking one country with highest probability, I get 67.72 % accuracy in training dataset and 61.43 % accuracy in testing dataset. There exist over fitting. When max depth equals to 10 and only picking one country with highest probability, I get 64.12 % accuracy in training dataset and 62.16 % accuracy in testing dataset. And then I get score 0.86694 in real test dataset when submit 5 countries with the top 5 probability.

In random forest model, when max depth equals to 5 and only pick one country with highest probability, I get 58.48 % accuracy in training dataset and 58.07 % accuracy in testing dataset. when max depth equals to 10 and only pick one country with highest probability, I get 60.77 %

accuracy in training dataset and 60.05 % accuracy in testing dataset. And I get score 0.86097 in real test dataset when submit 5 countries with the top 5 probability.

In XGBoost model, when max depth equals to 6 and only pick one country with highest probability, I get 64.08 % accuracy in training dataset and 63.21 % accuracy in testing dataset with max depth equals to 6. And I get score 0.86432 in real test dataset when submit 5 countries with the top 5 probability.

Table 3 Prediction Accuracy

Model	Accuracy	Top-5 Score
Multi Logistic	58.07%	0.85673
Decision Tree	61.44%	0.86694
Random Forest	60.05%	0.85674
XGBoost	63.21%	0.86432

Also, I finished the introduction, dataset and introduction of approach part of this report.

5. Conclusion

5.1. Findings

Erica Chen: In this project, I participated in explorational data analytics, data preprocessing, feature engineering and modeling. During the process from data exploration to feature engineering, I found that the feature selection and creation are really important in order to make accurate prediction. In this task, over 70% of the target label is US, which means that if we just predict all the destination as US, we could get at least 70% accuracy. But in real world, this does not have real impact for customized targeting. Therefore, how to identify customer online behavior and account information along with the country information to predict 5 destination country is more practical in real world application, and this is what I try and learn in this project.

I-chen Lee: From the process of session merged to account ID-level datasets, and the session helped our model performance increased from 85% to 87.7%, I realized that even only one third of train IDs is with these new features, and even only 36 action types out of 370 were kept, it will still help a lot. Therefore, 20-80 efficiency principle works and we shall never give up developing new features. After exploring and capturing the habits of IDs in different aspects, we can move on to feature selection and modeling, which would be with larger base of data and reach better performance.

Qiaowen: In this project, I participated in data preprocessing, explorational data analytics, and modeling. Also, I write the Introduction and Conclusion part of the final report, as well as summarize and revise all the individual parts. In the Introduction and Conclusion part, I found that it is really very important to make accurate recommendations in Airbnb website, especially for new users. Actually most time we can draw the conclusions that how to attracts new users and turn their attention from hotel booking websites decides the market share to some degree. In data preprocessing, explorational data analytics, and modeling part, I found that Date data is always hard to deal without DatetimeIndex. After transformation, date time can be extracted into some new features which indeed help to improve accuracy prediction. And XGbbost always performs best but it costs much more time than other machine learning methods, which have trouble with tuning parameters. Besides, predicting top 5 country destinations is reasonable and critical. Due to long tail effect, it is almost impossible to predict the best answer for users correctly when there are many choices in candidate pool. Thus, it makes “top 5 prediction” more reasonable to attract long-tail users.

Yingxin Chen: In this project, I took part in the whole data analysis process, from data preprocessing to feature engineering and finally building models to get prediction in classification. I find that more features doesn’t mean more accuracy in prediction. To get the best performance model, I need to select the most importance features and leveraged feature engineering methods like normalization and dummy. Data preprocessing is significant in data analysis and requires not only skills but also experience. When building models, I find that usually ensemble models performance better than single model. And for this Airbnb project, when combined with real situation, more than one output for each example is allowed, because Airbnb could recommend more than one destination countries to new users.

Xinyi: In this project, I did data preprocessing, feature engineering and modeling to predict country destination for each user. I found that sometimes max vote the predictions made by multiple models could not improve the accuracy. For example, for each user, random forest, multinomial logistic regression and xgboost recommend 5 countries respectively. I was thinking about to improve the accuracy by max vote the predicted countries according to their frequencies. However, after doing the max vote, the final accuracy decreased and even became less than any individual model's accuracy. The reason might be when I did the max vote, I only concerned about the predicted countries' frequency but ignored their corresponding predicted probabilites. Therefore, I think when we max vote the predictions generated by multiple models, we need to combine with real situation or model's internal machanism.

5.2. Real-World Implications

1. **Leverage the Long Tail Effect** (Qiaowen)

The long tail is a statistical pattern of distribution that occurs when a large share of occurrences occur farther away from the center or head of distribution. This means that a long tail distribution includes many values that are far away from the mean value. In the context of our project, this signifies that many favorite country destinations of users are different from the most mainstream destinations. This implement is of considerable importance since there exists enormous number of users who prefer travelling to some unpopular destinations. Each group of users with same unpopular destination is small, but total of them is large.

Creating a lot of content is a popular way to leverage Long Tail Effect. It is surprising that sites with the most content also attract the most visitors, such as Craigslist.org, eBay.com, Amazon.com, etc. In this way, Airbnb could display some specific countries and accommodations to attract not only main users, but also long tail users. Another way to leverage Long Tail Effect is to improve forecast accuracy, which is the main task in our project. With some feature engineering techniques as well as effective machine learning models, the accuracy could be up to 87%. The accuracy is pretty high in view of the real applications. Consequently, Airbnb could improve the recommendation display on Airbnb Home Page. In addition, some promotional emails and messages could be sent to customers to remind them of checking out their ideal destination housing

2. **Partner with real estate brokers in cities of emerging needs** (Qiaowen/Ichen)

Improve prediction accuracy to provide appropriate housing for users is one thing in case of demand at the downstream end of the supply chain. Another thing is to analyze the needs for different countries and cities at the upstream. After relative accurate users' needs prediction, Airbnb could provide more Airbnb sources for future demand trend. In this way, Airbnb collaborates with users and real estate brokers. The whole chain will be more effective through better collaboration with users and real estate brokers. When successfully understanding users' plans and forecasts, Airbnb can build promotions and seasonality into the forecast and then provide more insight to their real estate brokers to help prevent the unnecessary cost of buildup and maintenance. What's more, the visibility and insight Airbnb provide will enable fast decisions with visibility and insight for real estate brokers, which will be definitely beneficial to them in reducing costs, including fix cost, sink cost and human cost.