Airbnb's New User Bookings Kaggle Competition

Lisa Oshita

April 4, 2018

1 About the Competition

Founded in 2008, Airbnb is an online accommodation marketplace featuring millions of houses, rooms and apartments for rent or lease in over 200 countries. As part of a recruiting competition, Airbnb partnered with Kaggle to host a data science competition starting in November 2015 and ending in February 2016. The task of this competition was to build a model to accurately predict where new Airbnb users will make their first bookings. There are a total of 12 possible destinations to predict: US, France, Canada, United Kingdom, Spain, Italy, Portugal, Netherlands, Germany, Australia, Other and no destination found. No destination found indicates that the user did not make a booking, while Other indicates that a booking was made to a country not already listed. For this competition, Airbnb provided six data sets for participants to use in model building. I used two of these data sets: train_users, which contains information about the users, including where they first booked, and sessions, which contains information about user's web session activity.

2 XGBoost, Random Forest, Stacked Models

Extreme Gradient Boosting (XGBoost) is a fairly new method of supervised learning that performs consistently better than single-algorithm models. It is a form of gradient boosting that introduces a different, more formal form of regularization to prevent overfitting—enabling it to outperform other models. Additionally, XGBoost algorithms are fast. At its core, the algorithm is parallelizable which allows this model to fully utilize the power of computers. As XGBoosts have been used to place highly in Kaggle competitions and were also used by many of the top 100 participants of this Airbnb competition, this was the model I first decided to explore and implement.

The second model I decided to implement was a random forest. Random forests are another form of ensemble modeling that have performed well in Kaggle competitions.

As the top three winners of this Airbnb competition used a form of stacked modeling, I also decided to explore a general form of this. Stacking, also called meta ensembling, is a technique used to combine information from individual

predictive models to create a new model. Stacked models usually outperform its base models—as it's able to correct and build upon the performance of those base models.

3 Exploratory Analysis

4 Feature Engineering

From the train_users data I created a total of ? features. I pulled apart the month, year and day of the week of the date features (date account created, date first active, date of first booking) and created a season variable (winter, spring, summer, fall). I also calculated the difference in days between each date feature: days between the date an account was created and the date of first booking, and days between date first active and date of first booking. I also cleaned the age and gender features.

From the sessions data I created a total of blah features. The data contained five features describing userâĂŹs web session activity: action (193 levels), action_type (9 levels), action_detail (93 levels), device_type (13 levels), and secs_elapsed. I aggregated the data by each user and created features counting the number of unique levels for each of these features. From secs_elapsed, I also calculated summary statistics like the mean and median for each unique user. These features were then joined by user id to the training data.

The following is a list of all features included in the training data:
Features in the original data

- signup_method
- signup_flow
- language
- affiliate_channel
- affiliate_provider
- first_affiliate_tracked
- signup_app
- first_device_type
- first_browser

Features derived or cleaned

- Year, day of the week, month and season of date account created, date first active, date first booking
- days between date account created and date of first booking

- days between date first active and date of first booking
- age
- gender
- count features created from the sessions data
- mean, median, standard deviation, minimum, maximum of secs_elapsed

One-hot encoding was used to convert categorical features to a form that works better with machine learning algorithms. Essentially, a boolean column indicating a 1 or 0 was generated for each level of the categorical feature. Continuous features were left as is. After one-hot encoding and feature engineering, there were a total of 596 features to use in the models.

5 Model Building