Machine Learning Engineer Nanodegree

Kaggle Airbnb New User Predictor

Apik Zorian

January 25, 2018

Proposal

Domain Background

Airbnb is an online peer-to-peer property rental service that allows users to book short-term lodging. This includes rooms, apartments, entire homes, vacation rentals, etc. Airbnb brokers reservations between users and landlords for lodging all over the world. As of January 2018, the company had over 3 million listings in 65,000 cities over 191 countries (reference).

One way to immediately pique a new user's interest is to advertise bookings in a city or country the user would first like to visit. By accurately predicting where a new user will book his or her first trip, Airbnb can curate personalized content to showcase that will result in the user completing their booking. For Airbnb, this helps decrease the average time for first booking for a new user and helps personalize content for their community. It also improves the new user's first booking experience by curating content to their travel preferences.

Airbnb has done a fair amount of research on this topic, including initiating a Kaggle Recruitment Challenge to address this very issue. Many of the contestants shared their views on their approach to solving the first booking problem and their findings during implementation (see Pons and Kuroyanagi).

Problem Statement

The challenge then becomes: How can Airbnb predict the country in which a new user will make his or her first booking?

In this project, we will predict a new user's first booking by deploying machine learning algorithms to analyze data about the user that will help predict this first booking. Airbnb has posted this very problem as a Kaggle Recruitment Challenge and has provided New User Booking Data to help participants develop models to predict a new user's first booking. This data includes information about users demographics, web session records, and summary statistics.

Our goal is to analyze the data, build and train a model, and test this model to predict a new user's first booking.

Datasets and Inputs

The dataset provided by Airbnb includes 5 .csv files:

1. train_users.csv
2. test_users.csvThese datasets will be used to train and test our model, respectively. For each use in the dataset, we are provided the following features
   - user id
   - date_account_created: the date of account creation
   - timestamp_first_active: timestamp of the first activity, note that 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
   - age
   - signup_method
   - signup_flow: the page a user came to signup up from
   - language: international language preference
   - affiliate_channel: what kind of paid marketing
   - affiliate_provider: where the marketing is e.g. google, craigslist, other
   - first_affiliate_tracked: whats the first marketing the user interacted with before the signing up
   - signup_app
   - first_device_type
   - first_browser
   - country_destination: this is the target variable you are to predict

   The final feature is our traget feature and is not provided in the testing dataset. These features will be used to train our Machine Learning model and eventually test how well it can predict the new user's first booking
3. sessions.csv - This file contains information on the user's web sessions. Features included in this file contain:
   - user_id: to be joined with the column 'id' in users table
   - action
   - action_type
   - action_detail
   - device_type
   - secs_elapsed

   This information can be used to provide meaningful insight on the user that may help us predict his or her first booking. For example, we could see the different types of actions a user took, if he/she began planning a trip, how long he/she spent deciding on an action (secs elapsed), etc. This information could be used to help predict where the user would want to travel first, as well as things about the trip that are important to the user, which Airbnb could leverage in their first attempt at customizing the trip for the user.
4. countries.csv - Summary statistics of destination countries in this dataset and their locations. This includes:
   - country name
   - latitude and longitutde
   - distance from U.S. (km2)
   - language and language levenshtein distance (how close words in the language are to words in english language)

   This information can be used to help determine if the country would interest the user based in its distance from the U.S. and whether or not the user would feel comfortable given the spoken

language of that country. For example, the user may speak the main language spoken in that country, or the distance is not far enough from the U.S. to where the user would be discouraged from booking a trip.

5. age_gender_bkts.csv - Summary statistics of users' age group, gender, country of destination. This dataset includes the following columns:
   - age_bucket
   - country_destination
   - gender
   - population_in_thousands
   - year
6. This information can be used to help understand what types of other users have chosen select countries. For example, we can see what age range of men booked trips to France in 2015 and compare this to the user's age and gender. This information would help determine whether or not the user would want to book a trip to this destination.

Solution Statement

We believe that features in training dataset can be leveraged to come up with a model that can accurately predict where a new user's first destination will be. We understand that information about past users, including their actions, characteristics, demographics, and personal information, can be used to develop a machine learning model to predict where a new user with similar traits would decide to travel.

We will begin with the 15 features included in the `training_set` as the inputs for the model, and the `country_destination` feature as the label. Along the way, we may find some features less useful than others, while some features combined may help simplify our model. These will be explored more as we test our model's capabilities and determine the pros and cons of using different models.We will be sure to test out a variety of supervised-learning models, including SVM, Decision Trees, and Random Forest. We will use ensemble learning techniques such as Gradient Tree Boosting and XGBoost as our final solution and will prove finalize how boosting will yield a better result than standard supervised learning algorithms. We will also do parameter tuning using Gridsearch.

Benchmark Model

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'.

To determine our benchmark, we will make predictions using traditional supervised learning models, such as SVM and Decision Trees.

Evaluation Metrics

Kaggle assesses submissions based on Normalized discounted cumulative gain (NDCG). NDCG is calculated as:

$$DCG_k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{\log_2(i+1)},$$

$$nDCG_k = \frac{DCG_k}{IDCG_k},$$

where reli is the relevance of the result at position i. We will be making a maximum of 5 predictions per booking (k = 5).

For each new user, we make a maximum of 5 predictions on the country of the first booking. The ground truth country is marked with relevance = 1, while the rest have relevance = 0.

If, for example, the destination for a particular user is France (FR), then the predictions become:

$$[\text{ FR }] \text{ gives a } NDCG = \frac{2^1-1}{log_2(1+1)} = 1.0$$

$$[\text{ US, FR }] \text{ gives a } DCG = \frac{2^0-1}{log_2(1+1)} + \frac{2^1-1}{log_2(2+1)} = \frac{1}{1.58496} = 0.6309$$

As NDCG is the evaluation metric that Kaggle uses, to assess participants for their competition, this will be our metric as well.

Project Design

Our project will be composed of three steps:

```
1. Data Exploration: Along with visualizing our dataset, this step will include
using feature manipulation techniques, such as:
* Removing irrelevant features - We will identify features in the training data
that we think would not be relevant, drop them, and a decision tree regressor to
see if the prediction drastically changes with or without this feature.
* One-hot-encoding categorical features to be combined with numerical features
* Feature generating - We will create our own features by grouping different
features together that may be relevant to each other. This will help reduce the
number of total features and hopefully will lead to better predictions. We will use
visualize this data using scatter plots to see if any pairs of features exhibit
some degree of correlation.
* Detecting and removing outliers - Using [Tukey's
Method](http://datapigtechnologies.com/blog/index.php/highlighting-outliers-in-your
-data-with-the-tukey-method/) for identifying outliers, we will collect outliers
and later test our model on a training dataset with and without outliers to see how
removing the affects prediction.
* Removing null values - We will remove null values in user data that may skew our
predictions

2. Training and evaluating model: In this step, we will consider various supervised
Machine Learning models. Our end goal is to prove that XGBoost will yield the best
results, so we will evaluate results from traditional supervised learning models,
such as SVM and Decision Trees, and then develop an ensemble learning model using
```

XGBoost and compare the differences. We will use cross validation to determine which will be best suited for our task. We will also utilize Gridsearch to optimize our parameters.

3. Testing model: We will utilize our testing set to test our trained model, submit to Kaggle.