Machine Learning Engineer Nanodegree

Capstone Proposal

Kei Fukutani
May 13th, 2018

Domain Background

This project is based on the Kaggle competition, "Expedia Hotel Recommendations". [1] This competition is about a hotel search application that provides personalized hotel recommendations for its users. The customer data will be used to predict which types of hotels people are going to book.

I choose this competition, as there are a lot of situations in the real world that are similar to this kind of recommendation problems. I also believe that I can explore various machine learning techniques that are able to predict much better than human ability to solve this problem.

One of the early researches on this problem is the research paper by Gourav G. Shenoy et al. [2]

Problem Statement

This is a classification problem. Our goal is to predict the hotel groups for a user event based on their search, such as hotel location, check-in/out date, number of adults, and other attributes associated with that user event, including user location, distance between the user and the searched hotel, whether or not the hotel is booked, etc.

The company has in-house algorithms to form the hotel groups (“hotel cluster”), where similar hotels for a search are grouped together. Using provided logs of customer’s search behaviors as described above, we will predict the likelihood of 100 different hotel clusters that a user will book.

Datasets and Inputs

The datasets are provided by the competition organizer [3].

These datasets are the logs of customer’s search behaviors that include what customers searched for, how they interacted with search results (click/book), which hotel clusters the searched hotel falls in.

The training dataset has the “hotel cluster” label that we are attempting to predict, and test dataset does not. When looking at the common features, we see 13

categorical and 5 numerical features. The training data has 37,670,293 rows, and test data has 2,528,293 rows. The "hotel cluster" has 100 different values. The most frequent cluster has 98,550 samples, and the least frequent cluster has 1,480 samples.

As we cannot verify the test dataset directly, we will further split the provided training dataset into a training dataset and a validation dataset. The validation dataset will be used to validate the models and is about 20% of the provided training dataset in size.

Solution Statement

We will use supervised machine learning algorithms to predict the hotel clusters. Multi-class classification models, such as Gaussian Naïve Bayes, Random Forest Classifier, or any other model, can be trained on the training dataset that have features of a user search event and a hotel cluster label. Then we make predictions about the hotel clusters based on the trained model. This solution includes exploring data, preprocessing data, selecting models, and fine-tuning parameters.

Benchmark Model

Naïve predictor that chooses hotel cluster randomly can be used to compare with our model.

Evaluation Metrics

The model will output a predicted hotel cluster for each data in the validation dataset. Once trained and satisfactory scores are obtained with the validation data, the model will be retrained on the full dataset, and predictions will be made on the test dataset. The result will then be submitted to the Kaggle competition, where a score will be assigned to the model.

The solution score will be evaluated using the Mean Average Precision (MAP) [4].

Additionally, we will track prediction time for the scores achieved, as well as training time in order to quantify the effort needed to use the score in a production environment.

Project Design

This project is broken down into several categories.

- Data Exploration
- Data Preprocessing
- Benchmark Model
- Model Tuning and Prediction

Data Exploration

We will load the datasets and explore the data to see size, columns, and types of the data. Then we will examine which columns have numerical or categorical data, and whether or not the column is needed to predict the hotel cluster.

Data Preprocessing

Unnecessary columns and NaN value can be removed. We will calculate length of stay using 'srch_ci' (check-in date) and 'srch_co' (check-out date) and add it to the data. A logarithmic transformation will be applied on the skewed features so that the outliers do not negatively affect the performance of a learning algorithm. We will then normalize the numerical features. For the categorical features, we will convert them to numerical entries using one-hot encoding.

Benchmark Model

We will use a random model as a benchmark model. The model picks one out of 100 hotel clusters randomly.

Model Tuning and Prediction

We will fine-tune the chosen model using grid search with some important parameters. When the model is trained, whole data will be used if it is not too big. The final model performance will be evaluated and compared with the benchmark model.

Acknowledgement

[1] https://www.kaggle.com/c/expedia-hotel-recommendations
[2] Gourav G. Shenoy, Mangirish A. Wagle, Anwar Shaikh. "Kaggle Competition: Expedia Hotel Recommendations." Indiana University. arXiv:1703.02915 [cs.IR] 6 Mar 2017
[3] https://www.kaggle.com/c/expedia-hotel-recommendations/data
[4] https://www.kaggle.com/c/expedia-hotel-recommendations#evaluation