Subhadeep Chatterjee
@schatterjeecs

Starbucks Capstone Project

Udacity - Machine Learning Engineer Nanodegree

Domain Background

Starbucks is one of the most well-known companies in the world. A coffeehouse chain based out of Seattle, Washington which spans the entire globe.
Starbucks sends out personalized offers in such a way that the targeted audience ends up paying attention and not ignoring them. To reach the correct audience with the correct offer is very important in this context. This is where Machine Learning comes into play.

This project leverages Starbucks' offer related data as facilitated by Udacity. This project will aim towards predicting whether a customer is likely to accept the offer and thereby aims at maximizing the profit for Starbucks by using Machine learning.

In this digital world, customer behavior is one of the major driving forces in the retail domain, which has motivated me to take up this as the Capstone project. I thank Udacity and Starbucks for facilitating this to us.

I am an IT professional, with a lot of passion in Technology, and AI ML has always been a key space where I always wanted to contribute.

Problem Statement

Starbucks wants to find a way to give each customer the right in-app special offer. That is, how likely will the customer accept the offer that is sent to them.
Our goal is to analyze historical data about the offers transacted by the customers and develop an algorithm that can improve the conversion rate of the offers and thereby increase the overall profit.
We will use the dataset which has been provided by Starbucks for this task.

Datasets

The data is contained in three files:

- portfolio.json - containing offer ids and meta data about each offer (duration, type, etc.)
- profile.json - demographic data for each customer
- transcript.json - records for transactions, offers received, offers viewed, and offers completed

Here is the schema and explanation of each variable in the files:

portfolio.json

- id (string) - offer id
- offer_type (string) - type of offer ie BOGO, discount, informational
- difficulty (int) - minimum required spend to complete an offer
- reward (int) - reward given for completing an offer
- duration (int) - time for offer to be open, in days
- channels (list of strings)

profile.json

- age (int) - age of the customer
- became_member_on (int) - date when customer created an app account
- gender (str) - gender of the customer (note some entries contain 'O' for other rather than M or F)
- id (str) - customer id
- income (float) - customer's income

transcript.json

- event (str) - record description (ie transaction, offer received, offer viewed, etc.)
- person (str) - customer id
- time (int) - time in hours since the start of the test. The data begins at time t=0
- value - (dict of strings) - either an offer id or transaction amount depending on the record

Solution Statement

I will use the following strategies:

- **Explore** the data, understand the connections in the data. I feel that it is very important for any Machine Learning engineer to understand the data deeply.
- **Data engineering or Preprocessing** the data. Also refers to cleaning up the data and making it suitable for the algorithm to understand the data completely. The better the algorithm understands the data the better is the learning curve. This is the most important stage.
- Build a **model**, and repeat. I think it's important to try out multiple models before coming to a conclusion. Since the data requires supervised learning (classification), I will leverage the algorithms starting with Naive bayes (MultinomialNB), SVM, KNeighboursClassifier. And, I do also have a plan to try some Deep learning algorithms and see the performance.
- Eventually I will make the code generic enough so that it can choose the **best fit** model, which can be used for prediction.
- **Evaluation** - The process will be iterative, which means I will try to bring out the optimal number of features (i.e. tuning the hyperparameters) from the dataset which can give a better accuracy.

Benchmark Model

I will leverage several machine learning models before concluding the optimal model. I will also plan to induce deep learning models. F1-score, accuracy are some of the metrics which will be leveraged for determining the optimal model. Code will compare the scores of each of the ML models and pick the best fit model based on these scores.

Evaluation Metrics

I will mostly leverage the statistical metrics provided by sklearn library and they include the following:

- Precision/Recall - which will help determine the percentage of true positives.
- F1-score
- Accuracy

Mostly 80% of the data will be leveraged for training purposes and the rest will be reserved for testing. On top of that I will also cook up some random data to see if the model is able to classify them correctly.

Project Design

My plan is to create E2E software which will consist of two APIs primarily, leveraging either Flask or Quartz based Python frameworks.
One API will be used for training and generating the ML model, this will be batch processing based API, asynchronous in nature.
The other API will be used for real-time data predictions.
For data processing and cleaning I will also try to leverage real-time frameworks like Apache Spark, if time permits.
For introducing deep learning on the data, I will try to leverage Tensorflow.

Acknowledgement

I thank Udacity & Starbucks for facilitating such a great project. It is a great learning experience.