Machine Learning Engineer Nanodegree

Capstone Proposal

Mounika Kajjam

April 9th, 2020

Domain Background

Starbucks Corporation is an American coffee company and coffeehouse chain. Starbucks was founded in Seattle, Washington, in 1971. As of early 2019, the company operates over 30,000 locations worldwide.
Once every few days, Starbucks sends out an offer to users of the mobile app. An offer can be merely an advertisement for a drink or an actual offer such as a discount or BOGO (buy one get one free). Some users might not receive any offer during certain weeks.

Not all users receive the same offer, and that is the challenge to solve with this data set.

Every offer has a validity period before the offer expires. As an example, a BOGO offer might be valid for only 5 days. You'll see in the data set that informational offers have a validity period even though these ads are merely providing information about a product; for example, if an informational offer has 7 days of validity, you can assume the customer is feeling the influence of the offer for 7 days after receiving the advertisement.

If the company sends more offers or irrelevant offers to a person then there is no use. That's why they should send only relevant offers to a person who is likely going to use this offer which benefits the company.We need to predict who is more likely going to use a certain offer and send that offer only to them to make the company beneficial.

My motivation behind choosing this problem are, firstly this is a more realistic problem which I see in the day-to-day like where I will be receiving the offers. Second thing is feel this is a challenging task to be completed.

Problem Statement

Starbucks wants to find a way **to give to each customer the right in-app special offer**. Our goal is to analyze historical data about app usage and offers / orders made by the customer to develop an algorithm that associates each customer to the right offer type. We can assess the performance of the project by measuring the correct association by applying the model to past data.

This is a classification problem and here we are trying to build a machine learning model that predicts whether or not someone will respond to an offer, so that the model can be used for predicting the best offer to Starbucks customers.

Datasets and Inputs

For this project, we will be leveraging the data graciously provided to us by Starbucks /Udacity. This is given to us in the form of three JSON files. We have information about 10 offers: 4 BOGO, 4 Discount and 2 Informational. A customer can interact with an offer by receiving it, viewing it or completing it.It is possible for customers to complete some offers without viewing them.

Before delving into those individual files, let us first understand the three types of offers that Starbucks is looking to potentially send its customers:

Buy-One-Get-One (BOGO): In this particular offer, a customer is given a reward that enables them to receive an extra, equal product at no cost. The customer must spend a certain threshold in order to make this reward available.

Discount: With this offer, a customer is given a reward that knocks a certain percentage off the original cost of the product they are choosing to purchase, subject to limitations.

Informational: With this final offer, there isn't necessarily a reward but rather an opportunity for a customer to purchase a certain object given a requisite amount of money. (This might be something like letting customers know that Pumpkin Spice Latte is coming available again toward the beginning of autumn.)

With that understanding established, let's now look at the three provided JSON files and their respective elements:

1. **profile.json**: This file contains dummy information about Rewards program users. This will serve as the basis for basic customer information.(17000 users x 5 fields)

- gender: (categorical) M, F, O, or null
- age: (numeric) missing value encoded as 118
- id: (string / hash)
- became_member_on: (date) format YYYYMMDD
- income: (numeric)

2. portfolio.json: This file contains offers sent during a 30-day test period. This will serve as the basis to understand our customers' purchasing patterns.(10 offers x 6 fields)

- reward: (numeric) money awarded for the amount spent
- channels: (list) web, email, mobile, social
- difficulty: (numeric) money required to be spent to receive reward
- duration: (numeric) time for offer to be open, in days
- offer_type: (string) bogo, discount, informational
- id: (string / hash)

3. transcript.json: This file contains event log information. Complementing the file above, this file will serve as a more granular look into customer behavior.(306648 events x 4 fields)

- person: (string / hash) Unique customer ID

- event: (string) Type of event that occured. Either offer received or offer viewed or transaction or offer completed
- value: (dictionary)Either an offer ID or transaction amount depending on the event
- offer id : (string / hash) not associated with any “transaction”
- amount: (numeric) money spent in “transaction”
- reward: (numeric) money gained from “offer completed”
- time: (numeric) time in hours after start of test

Solution Statement

The solution for this problem will be to combine the three files and make a single file or table which will have all the demographics of the person and offer details and the predicted output (i.e., a person will or not respond to the offer given). I will use Supervised Learning models to determine the propensity for a customer to complete an offer. I plan to implement Logistic Regression, K-Nearest Neighbors, Decision Tree, Support Vector Machine, Random Forest, and Naive Bayes.

Logistic Regression is a common technique used for binary classification problems. Propensity models are a form of binary classification, since they are concerned whether a customer is likely to respond to an offer or not.

Support Vector Machines attempt to find the best dividing hyper planes in the data to determine whether to send an offer or not.

Also, I would check the model with other classification algorithms.

And I plan to use a GridSearchCV for hyperparameter tuning.

Benchmark Model

For a benchmark model, I will first build a Logistic Regression model and compare all other

models against that. This should give me a solid benchmark to compare against since Logistic Regression is “quite efficient in dealing with the majority of business scenarios[for propensity modelling]”

We will use Logistic Regression Model as a benchmark in which to compare our model performance , because it is fast and simple to implement

Evaluation Metrics

Since we have a simple classification problem, I will use accuracy to evaluate my models. We want to see how well our model is by seeing the number of correct predictions vs total number of predictions.

Why choose accuracy? First let's define accuracy, the ratio of the correctly labeled subjects to the whole pool of subjects. Also, accuracy answers questions like: How many students did we correctly label out of all the students? It's similar to our situation right? because we want to see how many customers use Starbucks offers. Furthermore,

$$\text{Accuracy} = (TP+TN)/(TP+FP+FN+TN).$$

Not to forget, that this is a simple classification problem, so this is my opinion and reasoning on why to use the accuracy.

Project Design

First, there is the **data preparation** step: we look at the data sources, understand their content and cleanse the data. For example, we aim at recreating the customer journey (from the *received offer* to the relative *transaction*) through the *transcript* dataset. Moreover, we have to join all the different pieces of information coming from the 3 data sources. Finally, we create the target variable, which is the base of all our analyses.

The next step is **data exploration**. We analyze the newly formed datasets to understand the distributions of the features and their relationship. We have to investigate possible missing values, and categorical features with too many categories. Then, we need to tackle the **data analyzation** part like Univariate Analysis and Bivariate Analysis. After analyzing the data, we transform the starting dataset through different stages like missing imputation, categories encoding. After that, we **develop the model**. We create different Machine Learning models, to predict the best offer, *BOGO* propensity and the *discount* counterpart. For each model, we try different algorithms, such as Logistic Regression, K-Nearest Neighbors, Decision Tree, Support Vector Machine, Random Forest, and Naive Bayes.. Finally, we **measure the performances** of the built process and compare them with the current benchmark, to understand if the proposed solution is viable to implement the current offer attribution process.