Capstone Proposal: Starbucks Capstone Challenge

Domain Background

The project is about creating an offer recommendation engine for the Starbucks marketing team. Every few days, Starbucks is sending out offers to users of the mobile app. There are mainly three main types of offers: discount, BOGO (buy one get one) and informational offers. Not all Starbucks customers receive the same offer.

Problem Statement

The problem of the project is to find a solution that will take customer attributes and offer attributes into account and suggest if the offer will be successful or not. By doing so Starbucks team will be able to check to which offer a customer will respond. By finding the right offer for the right customer Starbucks will be able to increase the marketing ROI. Also Starbucks will be able to target only customers that will respond to the offers, instead of sending the same offer to everyone and customers may become annoyed by Starbucks's advertising.

Solution

Based on historical data of customer responses a model will be created that will predict / suggest which type of offer a customer should get. The input for the model to make a prediction will be customer demographic data and offer attributes. Respectively the two datasets can be found in portfolio.json and profile.json files, see below. Based on the input the solution will provide if the offer is going to be successful or not.

The Datasets and Inputs

The dataset is the simplified version of the real Starbucks app data. The dataset contains three files: portfolio.json, profile.json, and transcript.json. The task is to combine transaction data with the demographic and offer data to identify which type of customers fit well to which type of offers.

A Benchmark Model

As a benchmark model, a Naive Classifier will be used. A Classifier that will simply mark all offers as successful. By using the Naive Classifier we can see how well the new model will perform in comparison to a random classification. The dataset will be combined and cleaned for a binary classification approach. Mainly the goal will be to predict if the user should receive the BOGO or the discount offer.

Evaluation metrics

As the main evaluation metric, the accuracy score will be used. For this type of problem, the accuracy score is perfectly fine. There is no huge imbalance in the dataset, also the imbalance will be fixed by oversampling. To fine-tune the model even further the f1, precision and recall metrics will be used to dig deeper into models performance.

Project Design

There will be 5 steps in the project.

1. Exploratory Analysis I: The first step will be to explore the 3 provided datasets in JSON format. The main point is to understand the general structure of the data, to explore the distribution of the data to see if any relationship can be found. It's more a general overview of the project.
2. Data Cleaning: During this step files from the step above that contain any missing rows will be dropped from the datasets. Also, columns that have ambiguous data such as in the profile.json, age column with values of 118 will be removed. Because there is no way someone is 118 years old. Also during this part, data will be merged into a meaningful structure, such as a combination of transactions with the profile data.
3. Exploratory Analysis II: Now the data is combined into a single data frame that can be analyzed to find patterns. During this part metrics such as conversion rate by offers, avg. spent by offers will be calculated and visualized to get a better understanding of the offer performance by other dimensions.
4. Feature Engineering: During this part features some of the features such as offer_names, or customer_ids will be dropped. Also, other columns that can cause problems for the algorithms will be dropped. Columns such as gender, offer_type or age_categories will be hot encoded. Features like age, income, duration and so on will be scaled for better performance.
5. Algorithm Selection: As already described above, as a baseline algorithm a Naive classifier will be used. To find the optimal algorithm, a brute-force approach will be used. To see which algorithm performs best. Also, the plan is to use ensembles, they seem to be reasonable for such a type of problem.
6. Model Training & Tuning: The plan is to train the model locally and to use a grid search to find the optimal hyper-parameters. The optimization with grid search will be the last step once the best algorithm was found.