Machine Learning Engineer Nanodegree

Udacity

Starbucks Capstone Project Proposal

Karim Atef Henry

December 19th, 2020

I. Domain background

Starbucks was established in 1971 by three local businessmen to sell high quality whole beans coffee. Starbucks focus on consumer habits and share its speciality of coffee with the buyers.

As we are now living in a new era where tons of data are generated every day which grow and scale dramatically. Applying businesses analytics concepts enable us to build **Descriptive Reporting, Predictive Reporting** and **Prescriptive reporting.** Additionally, we can enhance the business throughout different stages through the application of Machine learning.

One of the most applications of ML takes place in marketing and sales domain in order to provide more personalized, effective promotions, recommend products, increase profitability, studying new products and its effect on markets.

Customer targeting has changed the whole idea of marketing whereby you can be able to reach the most likely customer through patterns generated from search history. We are having many case studies applying this concept in their businesses including real estate, telecom, e-commerce also many research papers are discussing how can we make better use of customer data to reach better results and launching successful campaigns.

II. Problem Statement

Our project aims at identifying a criterion upon which Starbucks will be sending offers to its customers and predicting how they will respond to different offers depending on hidden traits of customers and which cluster is most likely to accept each offer from the following: buy-one-get-one (BOGO), discount, and informational. This helps perform better prediction of which offer is most likely to excite each of our clients individually.

III. Datasets and Inputs

This data set contains simulated data that mimics customer behavior on the Starbucks rewards mobile app whereby it simulates how people decide which product to purchase. The data is contained in three files:

portfolio.json: includes 10 offers with the following features:

- 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: includes 17000 users with the following features:

- age (int) - age of the customer, missing value encoded as 118.
- became_member_on (int) - date when customer created an app account
- gender (str) - gender of the customer (note some entries contain 'O' for other and rather than M or F)
- id (str) - customer id
- income (float) - customer's income

transcript.json: includes 306648 observations with the following features:

- event (str) - record description (ie transaction, offer received, offer viewed, etc.)
- person (str) - customer id
- time (int) - time in hours since start of test. The data begins at time t=0
- value - (dictionary) different values depending on event type
    - ✓ offer id: (string/hash) not associated with any "transaction"
    - ✓ amount: (numeric) money spent in "transaction"
    - ✓ reward: (numeric) money gained from "offer completed"

We will combine user profiles along with observations in transcript to help identify whether the user took the offer or not and time taken. Our target label is to determine whether or not the user takes the offer.

IV. Solution Statement

It's obvious that we will be dealing with a classification supervised learning model using SVM, k-nearest neighbour and other simple models in addition to a more complicated model as XGBoost and compare among them or use a combined model. We will be segmenting our clients based on different features individually and combined looking for meaningful patterns describing our client and determine which offer is most likely to excite him among the offers we have. Applying these machine learning concepts will enable us reach better results and will make our clients satisfied. It's very appealing when you find a certain company understanding your needs and sending you the offer that makes you relate to the company.

V. Benchmark Model

We will be deploying different classification models including logistic regression, support vector machine (SVM), k-nearest neighbour and compare between all of them to reach an optimum simple solution to our problem. We will be deploying neural networks in case these models fail to come up with great results. We will keep refining our model till we reach a reliable model with a certain accuracy and f1-score. We would refer to logistic regression model as benchmark model which can be used to give us an indication of the minimum score that our model can achieve.

VI. Evaluation Metrics

We will evaluate our classification models using precision, recall and f1-score after calculation of each of true positive (TP), true negative (TN), false positive (FP) and false negative (FN) and then construct confusion matrix. Additionally, we will look for accuracy which will be a valid choice in case of well-balanced and no skewed data. Precision is calculated through the formula: (TP)/ (TP+FP) and the recall through the formula: (TP)/ (TP+FN) and finally, the f1-score through: (2*((precision*recall)/(precision+recall))

VII. Project Design

Our project passes through different stages which start with:

1. **Importing data and libraries**: whereby we libraries used to read our data frame pandas and libraries to facilitate dealing with arrays numpy and libraries needed for visualization seaborn and matplotlib.
2. **Data cleaning**: which involves dealing with nan and missing data through imputation of the average into these values or removing observation in case it doesn't affect the output, checking incorrect data formats, generating visualizations to determine distribution of data and dealing with outiers which are defined as points that don't follow distribution of data. Outlier detection can take place through different methods, among them determining outlier score through proximity-based approaches or classification based model.
3. **Feature engineering**: through which we create some transformations to our features aiming at identifying features which will be most effective in our output also, it includes scaling our data using normalization technique.
4. **Splitting our data**: we split our data into training 80% and testing data 20% where we will be using training data to train our model and testing data which the model haven't seen before to evaluate our model and simulate our model in real life.
5. **Training model**: we start by training a simple model and more complicated ones.
6. **Tuning hyperparameters**: change some of default parameters in our trained model to enhance its accuracy.
7. **Testing model:** use our model to predict values of testing data.
8. **Evaluating model**: Using evaluation metrics to compare between different models.
9. **Model selection**: choosing the model with best results.