MACHINE LEARNING ENGINEERING NANODEGREE

PROPOSAL: STARBUCKS CAPSTONE

Domain Background:

Starbucks is a leading name in running coffee-house chains all over the world. It strives to always give its customers the best service and experience and via the mobile app it is quite dedicated to ease the user experience to make the orders online, predict the waiting time and send the customers special offers.

With the advent of the usage of Machine Learning(ML) in todays' world, the company can utilizes the strength of ML to increase its revenue. This project is concerned with associating the various offer types to its customers. The reason is that all users should not receive the same offer type. If we can understand the historical customer behaviour/reaction to these offers, we can create personalized offer predictions which can target these customers to increase the revenue. The data provided is collected from the mobile app.

As a machine learning researcher in real life, Starbucks' use case intrigues me to investigate the machine learning algorithms on this data to see how the offer types can be personalised with respect to the customers.

Problem Statement:

Starbucks collects the customer data to understand their behaviour on the rewards and offers sent to the via the mobile-app. Once every few days, Starbucks sends the personalised offers to its customers. These customers can respond positively/negatively/neutrally. A key thing to note is that not all the customers receive the same offer. The task of this project is to combine transaction, demographic and offer data of the past (which is already provided) to determine which demographic groups respond best to which offer types.

Datasets and Inputs:

The dataset simulates the customer behaviour towards the promotional offers. There are three types of offers that can be sent: buy-one-get-one (BOGO), discount, and informational. In a BOGO offer, a user needs to spend a certain amount to get a reward equal to that threshold amount. In a discount, a user gains a reward equal to a fraction of the amount spent. In an informational offer, there is no reward, but neither is there a requisite amount that the user is expected to spend. Offers can be delivered via multiple channels.

There are three json files provided:

portfolio.json: containing offer ids and meta data about each offer (duration, type, etc.)

- 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: demographic data for each customer.

- 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: records for transactions, offers received, offers viewed, and offers completed.

- 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 - (dict of strings) - either an offer id or transaction amount depending on the record

Further,
We have 3 offer types: BOGO, informational and discount.
We have information of 17000 customers and a transcript containing 306534 records of transactions concerning the three offers.

Solution Statement:

I propose to develop a baseline model(Logistic Regression-LR) to predict the best offer-type for the customer. A best offer-type implies the offer which the customer is more likely to convert or complete. LR is a good choice when the response variable is categorical. It is a simple model which uses the logistic function and the predictions are in the range of 0 and 1. Then, I would implement Support Vector Machine and XGBoost model (these are the advances classifiers) on the same data and compare the results of the three models to pick the best one.

Note: This project will use AWS-Sagemaker to deploy the trained model and to seek predictions.

Benchmark models:

To setup the baseline, Logistic Regression model will be used to compare with the other subsequent-implemented classifiers(SVM, XGBoost). Its evaluation metric will be used to analyse the predictions.

Evaluation Metrics:

- Precision/Recall: For any classifier, it is necessary to calculate the fraction of relevant instances among the retrieved instances(precision) and the fraction of the total amount of relevant instances that were actually retrieved(recall).
- F1: this is a point metric which tells the tradeoff between the precision and recall. Because optimizing precision necessarily does not increase recall and vice versa. We want a classifier which has good precision as well as recall.
- ROC-AUC: gives a trade-off between the true positive rate and false positive rate. It is the area under the ROC curve (Receiver Operating Characteristic).

Project Design:

- Data preparation step: In the first notebook, we will read the data and do some descriptive analysis on that. The goal will be to understand the structure and the content of the data.
- Data cleaning and merging: The goal will be to make a clear sense of the customers and their related completed/ not-completed offers. Later we join the three tables accordingly to construct the combined table.
- AWS-requirements: Next, a session will be established with AWS-Sagemaker.
- Model training: Baseline model and the advanced classifiers will be trained on the data. We will include the hyperparameter tuning as well.
- Evaluation: We will compare the models with the help of accuracy, f1 and roc-auc scores and pick the best one.
- Next, we will deploy the best picked model and generate the predictions via end-point.
- Last, we will delete the end-point and clear up the AWS resources to avoid unnecessary billing.

Final Note: This proposal contains the recommended solutions to the best of the knowledge before starting exploring the data properly. There might be inclusions/exclusions of proposed ML algorithms or evaluation metrics during the course of the project. If this happens, this document will be updated accordingly.