Machine Learning Engineer Nanodegree Capstone Proposal

Abhimanyu Rajput

Domain Background

Machine learning (ML) has become an increasingly important part of IT today. This effect is seen both in how IT leverages machine learning to improve operations and in how IT supports and enables the lines of business (LOBs). Still, organizations have limited understanding on its effective use and have made limited progress in associating it with business outcomes. Admittedly , The Companies which will lead in the future are those who will b e interested in implementing the machine learning algorithms on the enormous amount of data base which they have , they will be the pioneers in their field. STARBUCKS is one of flagship Worldwide companies which has been established since 31st March 1971 and have worldwide coffeehouse chain, and has a tremendous database of users , that is why I am interested in implementing m y capstone project for STARBUCKS Capstone Challenge as I believe that I can implement a good Machine Learning Model for one of the most

Worldwide prestigious companies. Customers' Concerns are the goal for all companies all over the world , what people like? , how much they want to pay?, when do they are capable to pay? , what is th e gender and age of those people who are interested and capable to pay? are ver y important questions and the answer comes from Historical data which we have to implement a deep learning algorithms to it , and building machine Learning Algorithms according to those Historical data to maximize Companies s' profits.

Problem Statement

The Problem Statement as mentioned in Starbucks Capstone Challenge ,analysing the data set for STARBUCKS Customers and building a Model that predicts whether or not someone will respond and complete to an offer.

We have an enormous number of users , some of them are making transaction either the received or not received an offer ,others are just viewing the offers without completing it , others responding to specific type of offers and completing it.

We have to make analysis for those who are receiving , viewing and make transaction within the offer period and those customers are our target.

Analysing the demographic feature for the above mentioned customers , their gender , age , income , the membership period and the type of offers which they are interested are the most important step before building our Model to stand on the Features which we will use in our Model.

The customers who are not influenced by the offers , or they purchase without having received an offer or seen an offer are NOT in out target. Cleaning , analysing and Visualizing the data consumes 90 % of the efforts to build a good model.

Datasets and Inputs

We have three JSON 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*

portfolio.json: shape (10 rows x 6 columns)

- *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:shape (2175 rows x 5 columns) with 17000 unique users..

- *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: (306534 rows x 4 columns)

• 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

After Cleaning and Preparation the data looks..

```
]: profile
```

]:

	gender	age	id	became_member_on	income
0	NA	118	68be06ca386d4c31939f3a4f0e3dd783	20170212	65404.991568
1	F	55	0610b486422d4921ae7d2bf64640c50b	20170715	112000.000000
2	NA	118	38fe809add3b4fcf9315a9694bb96ff5	20180712	65404.991568
3	F	75	78afa995795e4d85b5d9ceeca43f5fef	20170509	100000.000000
4	NA	118	a03223e636434f42ac4c3df47e8bac43	20170804	65404.991568
...	...	...	...	...	...
16995	F	45	6d5f3a774f3d4714ab0c092238f3a1d7	20180604	54000.000000
16996	M	61	2cb4f97358b841b9a9773a7aa05a9d77	20180713	72000.000000
16997	M	49	01d26f638c274aa0b965d24cefe3183f	20170126	73000.000000
16998	F	83	9dc1421481194dcd9400aec7c9ae6366	20160307	50000.000000
16999	F	62	e4052622e5ba45a8b96b59aba68cf068	20170722	82000.000000

17000 rows × 5 columns

```
portfolio
```

	reward	difficulty	duration	offer_type	id	mobile	social	web	email
0	10	10	7	bogo	ae264e3637204a6fb9bb56bc8210ddfd	1	1	0	1
1	10	10	5	bogo	4d5c57ea9a6940dd891ad53e9dbe8da0	1	1	1	1
2	0	0	4	informational	3f207df678b143eea3cee63160fa8bed	1	0	1	1
3	5	5	7	bogo	9b98b8c7a33c4b65b9aebfe6a799e6d9	1	0	1	1
4	5	20	10	discount	0b1e1539f2cc45b7b9fa7c272da2e1d7	0	0	1	1
5	3	7	7	discount	2298d6c36e964ae4a3e7e9706d1fb8c2	1	1	1	1
6	2	10	10	discount	fafdcd668e3743c1bb461111dcafc2a4	1	1	1	1
7	0	0	3	informational	5a8bc65990b245e5a138643cd4eb9837	1	1	0	1
8	5	5	5	bogo	f19421c1d4aa40978ebb69ca19b0e20d	1	1	1	1
9	2	10	7	discount	2906b810c7d4411798c6938adc9daaa5	1	0	1	1

```
transcript
```

	person	event	value	time	offer_id
0	78afa995795e4d85b5d9ceeca43f5fef	offer received	{'offer id': '9b98b8c7a33c4b65b9aebfe6a799e6d9'}	0	9b98b8c7a33c4b65
1	a03223e636434f42ac4c3df47e8bac43	offer received	{'offer id': '0b1e1539f2cc45b7b9fa7c272da2e1d7'}	0	0b1e1539f2cc45b7b
2	e2127556f4f64592b11af22de27a7932	offer received	{'offer id': '2906b810c7d4411798c6938adc9daaa5'}	0	2906b810c7d44117
3	8ec6ce2a7e7949b1bf142def7d0e0586	offer received	{'offer id': 'fafdcd668e3743c1bb461111dcafc2a4'}	0	fafdcd668e3743c1b
4	68617ca6246f4fbc85e91a2a49552598	offer received	{'offer id': '4d5c57ea9a6940dd891ad53e9dbe8da0'}	0	4d5c57ea9a6940dd
...	...	...	...	...	...
306529	b3a1272bc9904337b331bf348c3e8c17	transaction	{'amount': 1.5899999999999999}	714	
306530	68213b08d99a4ae1b0dcb72aebd9aa35	transaction	{'amount': 9.53}	714	
306531	a00058cf10334a308c68e7631c529907	transaction	{'amount': 3.61}	714	
306532	76ddbd6576844afe811f1a3c0fbb5bec	transaction	{'amount': 3.5300000000000002}	714	
306533	c02b10e8752c4d8e9b73f918558531f7	transaction	{'amount': 4.05}	714	

306534 rows × 7 columns

Solution Statement

My strategy is to develop a Machine Learning model to predict which is the best type of offer for each customer (with "best" I mean the type of offer that makes the customer more propense to convert). I will develop a model for each offer type and then combine the results to have a "best action" for each app user.

Benchmark Model

- *We will use Logistic regression model as a Benchmark in which to compare ou r models 's performance to , because it is fast and simple to implement.*
- *We will implement the AUC , Precision and Recall Metrics to Compare other Models 's Results.*

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 by seeing the number of correct predictions vs total number of predicitons. 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, 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 easiest (accuracy).

Reference :

First : https://towardsdatascience.com/accuracy-recall-precision-f-score-specificity-which-to-optimize-on-867d3f11124

Second : https://medium.com/thalus-ai/performance-metrics-for-classification-problems-in-machine-learning-part-i-b085d432082b

Cost function :-

Absolute Error :-

Absolute Error is the amount of error in your measurements. It is the difference between the measured value and "true" value. For example, if a scale states 90 pounds but you know your true weight is 89 pounds, then the scale has an absolute error of 90 lbs – 89 lbs = 1 lbs.

This can be caused by your scale not measuring the exact amount you are trying to measure. For example, your scale may be accurate to the nearest pound. If you weigh 89.6 lbs, the scale may "round up" and give you 90 lbs. In this case the absolute error is 90 lbs – 89.6 lbs = .4 lbs.

Reference : https://www.statisticshowto.com/absolute-error/

Mean absolute percentage error :-

The mean absolute percentage error (MAPE), also known as mean absolute percentage deviation (MAPD), is a measure of prediction accuracy of a forecasting method in statistics, for example in trend estimation, also used as a loss function for regression problems in machine learning.

Reference : https://en.wikipedia.org/wiki/Mean_absolute_percentage_error

Project Design

The process of our analysis will be by using the CRISP-DM Process (Cross Industry Process for Data Mining) : Define Business understanding, Data understanding,Analyze the data,Modeling the data,Compare model performance and finally selecting one model and improving it.

The first step is Business understanding : The objective here is to find patterns and show when and where to give specific offer to a specific customer.

The second step is Data understanding : First, *portfolio* dataframe ,Then, *profile* dataframe , Finally, *transcript* dataframe:

The third step is Data preparation and wrangling : Based on what we have seen in the previous step, there needs to be some work to prepare the data for analysis and modeling.

The fourth step is Analyzing the data : For this part, it will be divided into Univariate Exploration and Multivariate Exploration.

The fifth step is Modeling the data : I tried to make a model that can identify which kind of offers we should give a customer. Because my model will guess the *offer_type*, I will only get those transcripts with offer id's. So I will ignore all transactions without offer id's.

The sixth step is compare model performance : Now that we have trained the data, it's time to evaluate their performance based on accuracy.

The seventh step is model improvements : : After using *Grid Search* with *Logistic Regression* we managed to get better results.

Finally,conclusion and Improvements.