AWS Machine Learning Engineer Nanodegree

Udacity

Capstone Project Proposal

Salsabeel Tantoush

January 1st, 2022

Table of Contents

I. Domain Background

This project derives from the direct marketing system which Starbucks uses to keep in touch with its customers.

Aiming to incentivize and reward the customers registered in its platform, Starbucks periodically sends individual messages containing offers related to its products.

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: e-mail, social media, on the web, or via the Starbucks's app.

Nonetheless, marketing campaigns have associated costs. Hence, to be considered a successful campaign, it must generate profit higher than that initial cost. That means, companies expect to have a return on investment (ROI) as high as possible.

Thus, companies do not want to spend money sending offers to customers that are not likely to buy their products. On the other hand, new customers need to be attracted, so it is necessary to identify who are the people with a higher probability to respond to a marketing campaign.

Sometimes, recurrent consumers deserve some reward so they can feel appreciated and not forgotten. However, some customers keep coming back even if they do not receive offers. To give an example, from a business perspective, if a customer is going to make a 10-dollar purchase without an offer anyway, you would not want to send a "buy 10 dollars, get 2 dollars off" offer, unless this relative short-term loss means a more satisfied customer who will consume more in the future.

Another case is those customers who only buy products when receiving some reward, while other ones are opposed to marketing campaigns and do not want to be contacted at all.

Those are a few examples that illustrate how complex is the marketing decision process that has been faced by companies for years.

Considering the recent advances of artificial intelligence and the massive amount of data gathered over the years, this is a topic that could be widely improved by intelligent systems owing to the fact that they can analyze a large amount of data and understand patterns sometimes hidden for the human perception.

Personally, this is a very appealing subject since proposing better marketing campaigns may benefit not only the companies by raising their profit, but also the customers who receive more relevant offers in accordance with their consumer behavior. Perhaps, this win-win situation is the key to maintain the economy growing while people can afford better services and products.

The last section of this document – *VIII. Reference* – holds a list of the papers I have found helpful to formulate this project proposal.

II. Problem Statement

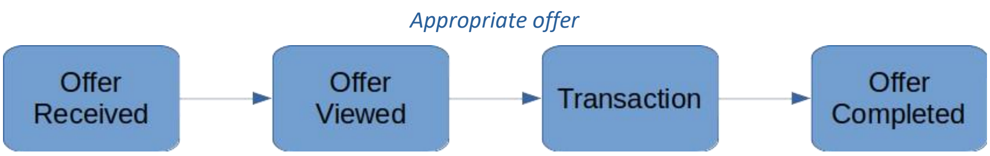
Starbucks, as well as any other company, invests money in marketing campaigns expecting to have a profit higher than the assumed before. So, identifying the most relevant offer to the correct customers is crucial for a successful campaign.

However, some targeted customers do not even see the offer sent to them, which may be a problem with the channel chosen. Other ones do not buy anything, despite seeing the offer, what might be a problem with the offer type sent, or maybe that is not a customer to be considered as a target.

There are other cases where the customers identify themselves with the offer, which leads them to try new products or spend more money than usual. Those are the situations to be identified and pursued.

The problem this project proposes to solve is finding the most appropriate offer for each one of the customers, which means finding the offer that is more likely to lead the customer to buy Starbucks products.

In the context of this project, an appropriate offer is that one where the customer sees the offer received and buys products under its influence, completing the offer lifecycle.



If a customer does not see an offer, it is not an appropriate one. If he or she sees the offer but does not complete it, it is not appropriate as well, since it did not lead the consumer to buy products. Similarly, if the customer buys some products, completes an offer, and receives a reward before visualizing that offer, it is not considered effective because the customer was not under the influence of that offer when decided to make a purchase.

III. Datasets and Inputs

a. Dataset overview

The data set used in this project is provided by Udacity and Starbucks as part of the AWS Machine Learning Engineer Nanodegree program. It contains simulated data that mimics customer behavior on the Starbucks rewards mobile app.

The program used to create the data simulates how people make purchasing decisions and how those decisions are influenced by promotional offers.

Each person in the simulation has some hidden traits that influence their purchasing patterns and are associated with their observable traits. People produce various events, including receiving offers, opening offers, and making purchases.

As a simplification, there are no explicit products to track. Only the amounts of each transaction or offer are recorded.

b. Data Dictionary

The data is contained in three files:

- **profile.json**
  Demographic data for each rewards program users
  Size: 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)

c. Brief dataset classes analysis:

Having started with the exploratory data analysis part of the project, I noticed that there are 3 main demographic features that distinguish the dataset classes and could help in choosing the right model and evaluation metrics to solve this problem:

1) **analysis for Gender and Age features:**

Age is normally distributed among the population, there is a local peak around the age interval 20 ~ 25 for both male and female genders. Apparently, this deviation is not a problem to be handled beforehand. However, this is a point to be taken into consideration if networks have difficulty to converge. Hence, simply standardization for this feature seems to be good enough.

2) **analysis of income feature:**

income is not a well-distributed feature. The density plot was bimodal, including some other local peaks. Splitting the income by gender, we note that the number of men presenting income below $80,000 is considerably higher than women.

- Graphs will be shared in the final report.

- **portfolio.json**
  Offer ids and meta data about each offer sent during 30-day test period Size:
  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)

- **transcript.json**

Event log containing records for transactions, offers received, offers viewed, and offers completed

Size: 306648 events x 4 fields

- person: (string/hash)
- event: (string) offer received, offer viewed, transaction, offer completed
- 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"
- time: (numeric) hours after start of test

IV. Solution Statement

In order to face the problem stated above, this project proposes to apply machine learning techniques to study customers' behavior by analyzing the transcriptions of their relationship with Starbucks.

More specifically, a neural network will be trained to predict how customers may react when receiving each one of the available offers: if they will complete the offer cycle or not. So, it will be possible to identify which one is more suitable for each customer.

Since consumers' behavior is not a feature isolated in the time, the next actions are affected by past experiences. Regarding this time-dependency, this project proposes to build and train a Recurrent Neural Network (RNN) as the central piece of the solution, aiming to analyze the customer behavior through the time.

The final result of this project is a direct marketing system that, given a customer, is able to predict the likelihood of each offer be completed.

V. Benchmark Model

A more traditional model will be trained in the same dataset used by the intended Recurrent Neural Network, so that the results are comparable. In this case, it means training a Feedforward Neural Network (FNN).

Basically, an FNN analyzes a static input and makes predictions not considering the customer history. Differently, RNN is able to make predictions based on past events, instead of analyzing the current moment as an isolated situation.

A naive network model might understand an offer as adequate because it produced a good result in the past, and tends to repeat that action every time. However, it may not be able to detect whether the same offer becomes inadequate when sent a second time to the same customer, perhaps owing to the fact that the customer does not want to repeat the same purchase forever.

In another case, the customer is conditioned to buy products, so no offer sending is necessary anymore. However, that naive network keeps suggesting the same offer over again.

This relationship between past experiences and future behavior is what the Recurrent Neural Network is supposed to recognize.

Building and training both models allows us to compare the predictions made considering only the static user state (FNN) and those made based on the customer history (RNN). Then, we will be able to evaluate whether the problem stated is better addressed with a Recurrent Neural Network model.

VI. Evaluation Metrics

The accuracy of the models will be measured to evaluate the performance of the networks. However, since some dataset classes are imbalanced, by using this metric alone, we will be unable to quantify and compare both the benchmark and the final models results precisely, as the models might develop bias towards the more common classes, therefore, accuracy, recall and precision will be used all together as the evaluating metrics, favoring precision over recall since it is the most relevant metric to the project's aim. Precision indicates how many positive labels were correctly classified against those classified as positive, but it is a negative case.

Considering that customers might have variations in their standard behavior, having an accuracy very close to 100% might indicate that the network just memorized the customers' behavior instead of understanding their consumption patterns.

On the other hand, having too low accuracy also might indicate that the network was not able to learn general patterns.

Hence, the target accuracy for the RNN in this project is around the range of 70% ~ 80%.

$$\texttt{accuracy}(y, \hat{y}) = \frac{1}{n_\text{samples}} \sum_{i=0}^{n_\text{samples}-1} 1(\hat{y}_i = y_i)$$

$$\text{precision} = \frac{tp}{tp + fp}$$

$$\text{recall} = \frac{tp}{tp + fn}$$

Where:

- $\hat{y}$ is the predicted value of the i-th sample and y is the corresponding true value.
- tp (True Positive) is the number of correctly classified sample when

the category is *positive.*

- *fp* (False Positive) is the number of wrongly classified sample when the category is *negative*.
- *fn* (False Negative) is the number of wrongly classified sample when the category is *positive*.

VII. Project Design

The theoretical workflow for approaching the solution stated includes several machine learning techniques, following the guideline sections below.

a. Data loading and exploration

Load files and present some data visualization in order to understand the distribution and characteristics of the data, and possibly identify inconsistencies.

b. Data cleaning and pre-processing

Having analyzed the data, handle data to fix possible issues found.

c. Feature engineering and data transformation

Prepare the data to correspond to the problem stated and feed the neural networks. The transcription records must be structured and labeled as appropriate offer or not.

d. Splitting the data into training, validation, and testing sets

Prepare three datasets containing distinct registers within each one.

The largest dataset is employed to train the networks, while the validation set, to evaluate the models during the training phase.

The testing set contains data never seen before by the networks, so it will be possible to consider this dataset as being new interactions between Starbucks and customers. By using this dataset, it will be possible to measure the final performance and compare the results of the trained models.

e. Defining and training a Feed-Forward Neural Network

Training of the benchmark model.

f. Defining and training a Recurrent Neural Network

Training of the proposed model.

g. Evaluating and comparing model performances

Comparison between the accuracy of both network models to verify each one is more suitable to solve the problem stated.

h. Presenting predictions for offer sending

Present the resulting predictions, along with discussions on how this system should be employed.

VIII. References

[1] D. Silver, L. Newnham, D. Barker, S. Weller, and J. McFall,"Concurrent Reinforcement Learning from Customer Interactions," in *ICML*, 2013.

[2] G. Theocharous, P. S. Thomas, and M. Ghavamzadeh, "Personalized Ad Recommendation Systems for Life-Time Value Optimization with Guarantees," in *IJCAI*, 2015.

[3] X. Li *et al.*, "Recurrent Reinforcement Learning: A Hybrid Approach," *ArXiv*, vol. abs/1509.03044, 2015.

[4] X. Zhao, L. Zhang, Z. Ding, D. Yin, Y. Zhao, and J. Tang, "Deep Reinforcement Learning for List-wise Recommendations," *ArXiv*, vol. abs/1801.00209, 2017.

[5] J. Jin, C. Song, H. Li, K. Gai, J. Wang, and W. Zhang, "Real-Time Bidding with Multi-Agent Reinforcement Learning in Display Advertising," in *CIKM '18*, 2018, doi: 10.1145/3269206.3272021.

[6] G. Zheng *et al.*, "DRN: A Deep Reinforcement Learning Framework for News Recommendation," in *WWW*, 2018, doi:10.1145/3178876.3185994.

[7] S.-Y. Chen, Y. Yu, Q. Da, J. Tan, H.-K. Huang, and H.-H. Tang, "Stabilizing Reinforcement Learning in Dynamic Environmentwith Application to Online Recommendation," in *KDD '18*, 2018, doi: 10.1145/3219819.3220122.

[8] F. Liu *et al.*, "Deep Reinforcement Learning based Recommendationwith Explicit User-Item Interactions Modeling," *ArXiv*, vol. abs/1810.12027, 2018.

[9] J. Zhang, J. Yin, D. Lee, and L. Zhu, "Deep Reinforcement Learning for Personalized Search Story Recommendation," *ArXiv*, vol.abs/1907.11754, 2019.