Machine Learning Engineer Nanodegree

Starbucks Capstone Challenge Capstone Proposal

Ali Ahmed Nada
January 9th, 2021

Domain Background

This data set contains simulated data that mimics customer behavior on the Starbucks rewards mobile app. Once every few days, Starbucks sends out an offer to users of the mobile app. An offer can be merely an advertisement for a drink or an actual offer such as a discount or BOGO (buy one get one free). Some users might not receive any offer during certain weeks.

Not all users receive the same offer, and that is the challenge to solve with this data set.

Your task is to combine transaction, demographic and offer data to determine which demographic groups respond best to which offer type. This data set is a simplified version of the real Starbucks app because the underlying simulator only has one product whereas Starbucks actually sells dozens of products.

Every offer has a validity period before the offer expires. As an example, a BOGO offer might be valid for only 5 days. You'll see in the data set that informational offers have a validity period even though these ads are merely providing information about a product; for example, if an informational offer has 7 days of validity, you can assume the customer is feeling the influence of the offer for 7 days after receiving the advertisement.

Datasets and Inputs

The data is contained in three files: **portfolio.json**

- 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

- 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

- 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

Solution Statement

we have 3 dataSets : Portfolio, Profile, transcript .

what we would do would be in 3 phases :

1. Data preparation
2. Feature extraction and preparation
3. Modeling

Benchmark Model

RNN : are a class of neural networks that are naturally suited to processing time-series data and other sequential data. Here we introduce recurrent neural networks as an extension to feedforward networks, in order to allow the processing of variable-length (or even infinite-length) sequences, and some of the most popular recurrent architectures in use, including long short-term memory (LSTM) and gated recurrent units (GRUs)

FNN : It consist of a (possibly large) number of simple neuron-like processing units, organized in layers. Every unit in a layer is connected with all the units in the previous layer. These connections are not all equal: each connection may have a different strength or weight. The weights on these connections encode the knowledge of a network. Often the units in a neural network are also called nodes.

Data enters at the inputs and passes through the network, layer by layer, until it arrives at the outputs. During normal operation, that is when it acts as a classifier, there is no feedback between layers. This is why they are called feedforward neural networks.

we will create the 2 Models In Order to measure the Accuracy for each one of them ..

Evaluation Metrics

The accuracy of the models will be measured to evaluate the performance of the networks. the target accuracy for the RNN in this project is about 80%

Project Design

1. Data extraction: You select and integrate the relevant data from various data sources for the ML task.
2. Data analysis: You perform exploratory data analysis (EDA) to understand the available data for building the ML model. This process leads to the following: Understanding the data schema and characteristics that are expected by the model. Identifying the data preparation and feature engineering that are needed for the model.
3. Data preparation: The data is prepared for the ML task. This preparation involves data cleaning, where you split the data into training, validation, and test sets. You also apply data transformations and feature engineering to the model that solves the target task. The output of this step are the data splits in the prepared format.
4. Model training: The data scientist implements different algorithms with the prepared data to train various ML models. In addition, you subject the implemented algorithms to hyperparameter tuning to get the best performing ML model. The output of this step is a trained model.
5. Model evaluation: The model is evaluated on a holdout test set to evaluate the model quality. The output of this step is a set of metrics to assess the quality of the model.
6. Model validation: The model is confirmed to be adequate for deployment—that its predictive performance is better than a certain baseline.
7. Model serving: The validated model is deployed to a target environment to serve predictions. This deployment can be one of the following:

references

https://www.fon.hum.uva.nl/ https://link.springer.com/ https://wikipidea.com
https://cloud.google.com/solutions/machine-learning https://classroom.udacity.com/nanodegrees/nd009-ent/parts/eba3d6e0-7db5-4d92-8877-94c321015ae5/modules/ce5d9de7-9dd1-4f2a-8ffe-0432d118f673/lessons/0346a707-1789-4373-b28a-1714386fc891/concepts/59623bdf-9fdf-4b34-a5f8-c56dc75fc512