Machine Learning Engineer Nanodegree

Capstone Proposal

Predict which products will an Instacart consumer purchase again

Vinay K

August 17th, 2019

1. Domain Background

We are shopping for things all the time, either it's from store or from online. In the latter case shopping experience has to meet customer expectation and retailer who is selling products must look into the current trend which in-turn helps their business adopt and grow.

A good understanding of customer buying pattern helps to address various business problems such as:

- Retailer stocking up right amount of items and thus avoiding overspending on products which aren't sold and underspending on products which are in demand

Instacart is an online grocery delivery service provider, customer can access the service via mobile app or website, then adding groceries to their digital cart. It has partnerships with regional and national local retailers. Instacart personal shoppers pick, pack and deliver the order within the customer's designated time frame.

2. Problem Statement

Same day delivery service for online orders are becoming more common, in order to meet the consumer demand there is a need for better understanding of:

- Where the customer are located
- From which store orders are being placed
- What items are being bought
- How many items are repeated in subsequent orders
- Frequency of orders

The main objective of this project is to use data on customer orders over time to predict which previously purchased products will be in a user's next order.

Ref: https://www.kaggle.com/c/instacart-market-basket-analysis/overview

3. Datasets and Inputs

The dataset consists of relational set of files describing customer's orders over time. The dataset is anonymised and contains a sample of over 3 million grocery orders from more than 200,000 Instacart users. For each user there is between 4 and 100 of their orders, with the sequence of products purchased in each order.

The entity that are provided includes:

- Aisles
- Departments
- Order_Products_Prior
- Order_Products_Train
- Orders
- Products

Order_products_* specify which products were purchased in each order

4. Solution Statement

The proposed solution to this problem involves two steps:

- Extracting new features
- Apply deep neural network to make prediction

For our model to work better - provide data in right format, entities listed in dataset are raw in nature but data is expected to be in format that it explain the correlation between data in some way, feature extraction is one such way to build values intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps.

Once we have the desired feature set, fine tune our neural network for better accuracy by changing parameters and hyperparameters.

Evaluate the model for accuracy and f-score in the last step.

5. Benchmark Model

Since the problem statement is taken from kaggle competition, we will use the Leaderboard and the winners score as benchmark.

F-Score of 0.40972 on Public Leaderboard is observed as best score for the above said problem statement.

Ref: https://www.kaggle.com/c/instacart-market-basket-analysis/leaderboard

6. Evaluation Metrics

Accuracy score and F-Score are used as evaluation metrics.

7. Project Design

Data Exploration:

Observe the data of each entity, check for duplicates and missing fields.

Plotting:

Before manipulating data, plot different observations that can be inferred by simple descriptive statistics.

Feature Extraction:

New features are extracted by combining data from different entities.

Scale the data using min-max normalization.

Create Labels for new feature set.

Model Architecture:

Use tensorflow sequential model with different hidden layers.

Change parameters and hyperpameters such as optimizer, metrics, number of hidden layers, and activation function until desired accuracy is achieved.

Training:

Training dataset along with generated labels is fed into model.

Output:

Save the result to file.

F – Score is calculated using the below formula:

F = ((1+beta**2)*truePositive) / (((1+beta**2)*truePositive) + beta**2*falseNegative+falsePositive)

Where,

truePositive = correctly predicted the product in next order

falsePositive = wrongly predicted the product in next order

falseNegative = failed to predict the product in next order