Consumer Behaviour in Retail: Next Logical Purchase using Deep Neural Network

Ankur Verma

Indian Institute of Technology BHU
ankur.verma.phe09@iitbhu.ac.in

Abstract

Predicting consumer purchase pattern is one of the most challenging problems for large scale retail firms. Retailers spend a lot of money and resources to ensure smooth and delightful shopping experience. This includes recommending relevant items to consumers and maintaining right inventory. The later helps minimise the instances of Out of stock and Excess Inventory. Consumer purchase prediction problem has generally been addressed by ML researchers in conventional manner either through recommender systems or traditional ML approaches. To our knowledge none of the models have generalized well enough in predicting the consumer purchase pattern. In this paper we present our study of consumer purchase behaviour at the intersection of consumer, item and time using e-commerce retail data. For each of the relevant consumer-item combination, we create a sequential time-series data. We then build generalised non-linear model to predict propensity of consumer to purchase the item for given time horizon. We demonstrate robust performance by experimenting with different neural network architectures, ML models and their combination. We showcase the benefits that neural network architectures like Multi-Layered Perceptron, Long Short Term Memory and Convolution Neural Network bring over ML models like Xgboost and RandomForest.

Introduction

Consumer behaviour insights have always been one of the key business drivers for retail, specially given fast changing consumer needs. Existing trend, competitor pricing, item reviews and marketing are some of the key factors driving todays consumer world in retail. While very little information is available on future variablities of the above factors, what retailers have is large volumes of transactional data. Retailers use conventional techniques to model transactional data for predicting consumer choice. While these help in estimating purchase pattern for loyal consumers and high selling items with reasonable accuracy, they don't perform well for the rest. Since multiple parameters interact non-linearly to define consumer purchase pattern, traditional models are not sufficient to achieve high accuracy across thousands to millions of consumers.

In many of the retail brands, short term (4-6 weeks ahead) inventory planning is done on the basis of consumer pur-

---

chase pattern. Given that every demand planner works on a narrow segment of item portfolio, there is high variability in choices that different planners recommend. Also, given their busy schedule, they have very less interaction moments to discuss their views and insights over their recommendations. Hence, subtle effects like item cannibalization, item affinity, pricing remains unaccounted correctly. Such inefficiencies lead to gap between consumer needs and item availablity, resulting in loss of business opportunities in terms of consumer churn, out of stock and excess inventory.

In this paper, we apply multiple deep learning architectures along with tree based machine learning algorithms to predict the next logical purchase at consumer level. We showcase the performance of individual models with varying hyper configurations. We also show the performance of stacked generalization ensemble and F1-maximization which involves combining predictions from different models and fine tuning purchase probability cut-off at consumer level respectively. The following section explains the overall methodology adopted to solve the problem. It also lays out various algorithmic variants and neural network architectures applied to the problem. Finally, section 3 describes the experiments and results obtained in various scenarios of modelling.

Methodology

The overall approach includes Feature Engineering, Model Training, Stacking and F1-Maximization as shown in Figure **??**. We treat each consumer-item as an individual object and generate weekly time series based on historical transaction for each object. The target value at each time step (week) takes a binary input, 1/0 (purchased/not purchased). We then generate various types of features including date-time related, label encoded and target encoded within and across objects. Below are the feature groups with the type of features within each group:

- **Datetime:** Weekday order split of the shopper and the item, daypart order split of the item, Proportion of repurchase in previous orders, number of times A bought B in n recent orders and multiple interactions of user-item related features.

- **Consumer Profile:** Total number of orders by the user, Number of distinct items ordered by the user Time since

first order by the shopper, Time since last order by the shopper, Averge time gap between orders Reorder rate of the user, Reorder frequency of the user, Time of the day user visits, Specific item ordered by user in the past Order size based features, Orders with no previously reordered item

- **Item Profile:** Time since first order for the merchandise, Time since last order for the merchandise Averge time gap between orders, Reorder rate of the item, Reorder frequency of the item Number of co-occuring items with this item, Item's average position in the cart, Statistics around order streak for this item Item's total number of orders, Number of distinct users, Number of users buying it as one shot item

- **Consumer-Item Profile:** Time since first order for each combination, Time since last order for each combination Averge time gap between orders, Reorder rate of the combination, Reorder frequency of the combination, Streak -user purchased the item in a row, Average position in the cart, Co-occurance Statistics Replacement items, Total number of orders for the combination, If user already ordered the item today

The model we needed to build, thus, should learn to identify similarly behaving time series across latent parameters, and also take into account consumer and item variations in comparing the time series. A point in a time series is represented

$$y_{(C_t, I_t)} = f((IA_t), (CA_t, .., CA_{t\text{-}n}), (CIA_t, .., CIA_{t\text{-}n}), (D_t, .., D_{t\text{-}n}))$$
(1)

where yit is sales for item 'i' at time 't', Ai is attribute of the item 'i' like colour - blue, material - cotton etc., Mit indicate merchandising factors like discount, promotion for items 'i' at time 't', Dit are derived features like trend, seasonality which are inferred from data and affect the sales, p is number of time lag.

Accuracy Measure

Consumer Basket prediction framework should be able to strike good balance between Precision and Recall. Precision refers to the ratio of number of predicted items actually purchased from the predicted items,

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

whereas Recall refers to the ratio of predicted items actually purchased from the items in the actual basket.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

F1 Score is needed when you want to seek a balance between Precision and Recall. We have previously seen that accuracy can be largely contributed by a large number of True Negatives which in most business circumstances, we do not focus on much whereas False Negative and False Positive usually has business costs, thus F1 Score might be a better measure to use if we need to seek a balance between Precision and

Recall and there is an uneven class distribution (large number of Actual Negatives).

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

Model Architectures

As mentioned in previous section, traditional time series models are not suitable choice for f . Hence, we work with machine learning models ranging from tree based models like Random Forest and various flavours of Gradient Boosted Trees, to deep learning models. We train two deep learning models, first of which uses Multi Layer Perceptron (MLP) architecture, and second is based on LSTM (chosen due to its ability to model long term temporal dependencies), to derive the relation