Customer Retention for the Google Merchandise Store

Jay Patel | Sanchit Deora | Anshika Saxena | James Diffenderfer

Group 10

Objective

- Predicting ***Customer Retention*** for a company or store
- Given data collected on a Google Merchandise Store (GStore) customer, predict if that customer will return to shop at the GStore again

Software and Tools

pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$

TensorFlow

jupyter

scikit learn

K Keras

CO

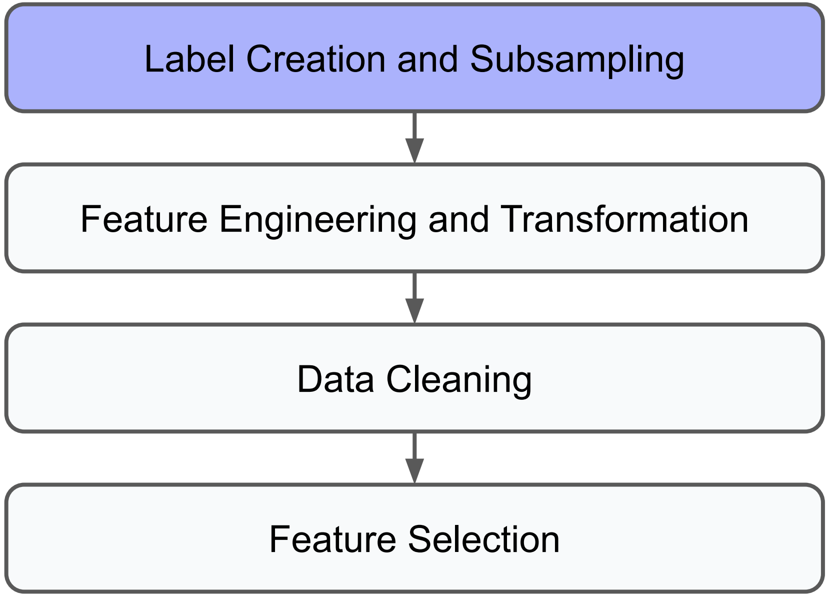
Seaborn

Google Merchandise Store Data Set

- Data originally provided for Kaggle Customer Revenue Prediction competition
- Data provided in training and testing sets
  - **Training set (25 GB)**: User transactions from August 1, 2016 to April 30, 2018
  - **Testing set (8 GB)**: User transactions from May 1, 2018 to October 15, 2018
- List of **13 original features** (orange indicates JSON data)
  - fullVisitorId, channelGrouping, date, **device**, **geoNetwork**, **totals**, sessionId, socialEngagementType, **hits**, trafficSource, visitId, visitNumber, visitStartTime

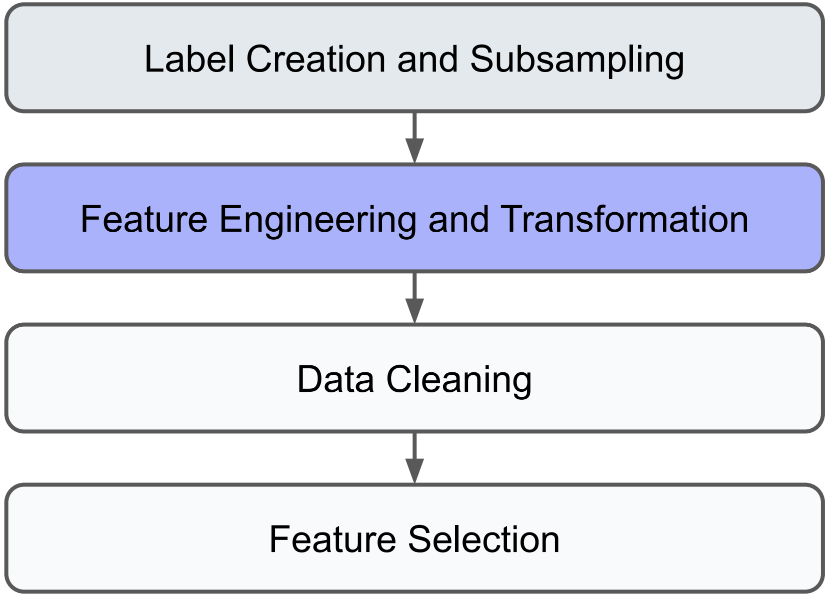
Data Preprocessing Pipeline

- Created **customerReturns** labels
- **Subsampled rows** from full data set
  - Used **KMeans clustering** and **stratified sampling**
- Percentage of returning customers:
  - Full Dataset: 33.3 %
  - Subsampled Dataset: 33.4 %


Label Creation and Subsampling
Feature Engineering and Transformation
Data Cleaning
Feature Selection

Data Preprocessing Pipeline

- Used data transformation to make data more comprehensive
  - Cyclic features (**Date**, **Time**)
  - Location features (**Latitude**, **Longitude**)
- Normalised data for better performance



Label Creation and Subsampling

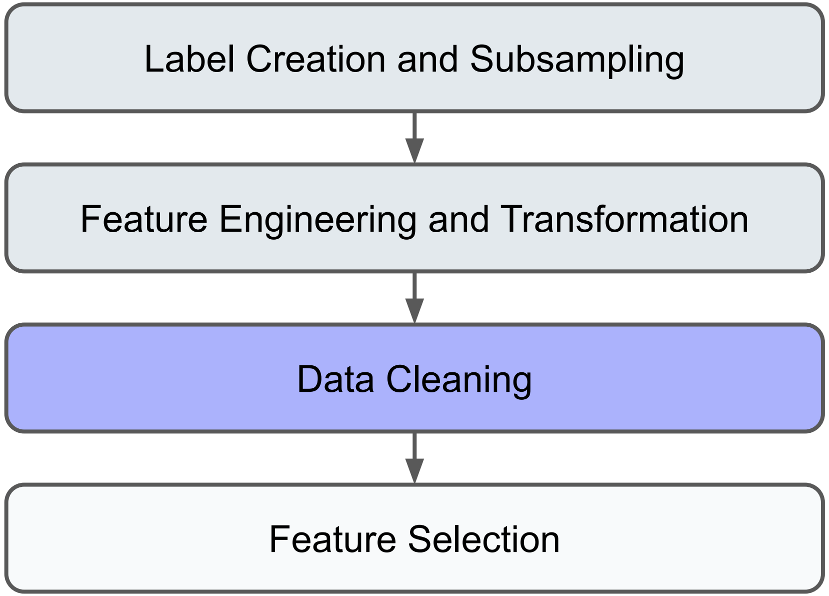
Feature Engineering and Transformation

Data Cleaning

Feature Selection

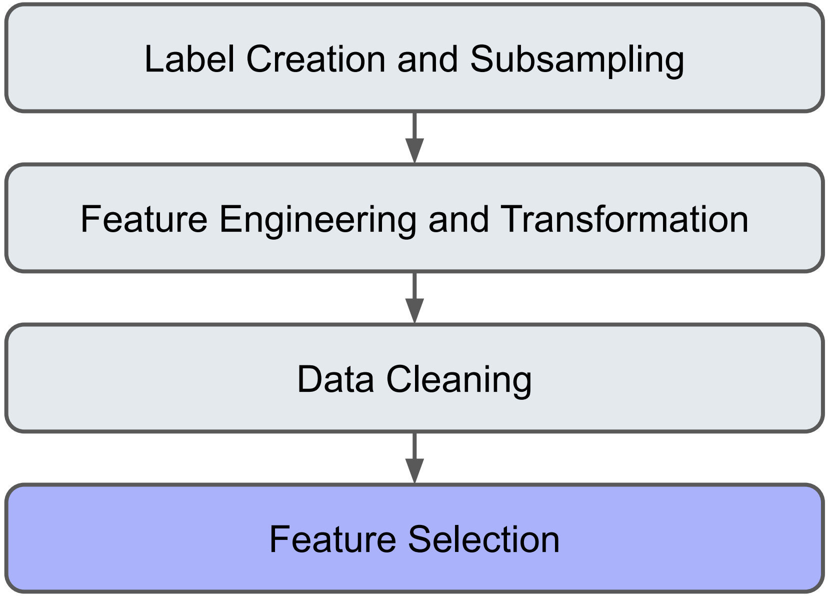
Data Preprocessing Pipeline

- Filled in **missing values**
  - Filled **-1** for numerical values
  - Filled **'UNK'** for string values
- **Deleted columns** if more than 90% of the data was missing


Label Creation and Subsampling
Feature Engineering and Transformation
Data Cleaning
Feature Selection

Data Preprocessing Pipeline

- Used **Extra Trees Classifier to get importance** for each feature
- Removed possibly non-contributing features



Label Creation and Subsampling

Feature Engineering and Transformation

Data Cleaning

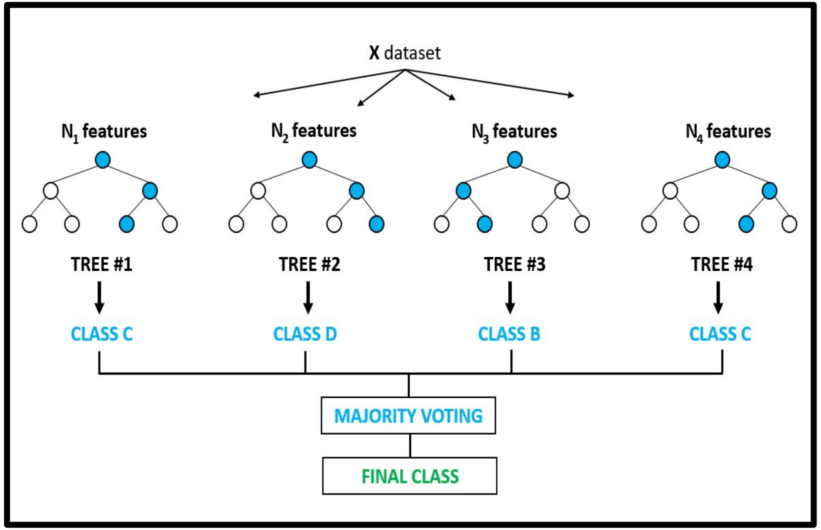
Feature Selection

Data Set After Preprocessing

- Preprocessing pipeline resulted in **42 features**
- **Training/Testing split** is approximately **81/19**
  - **Training Set:** 1,537,503 samples
  - **Testing Set**: 361,429 samples

Modeling

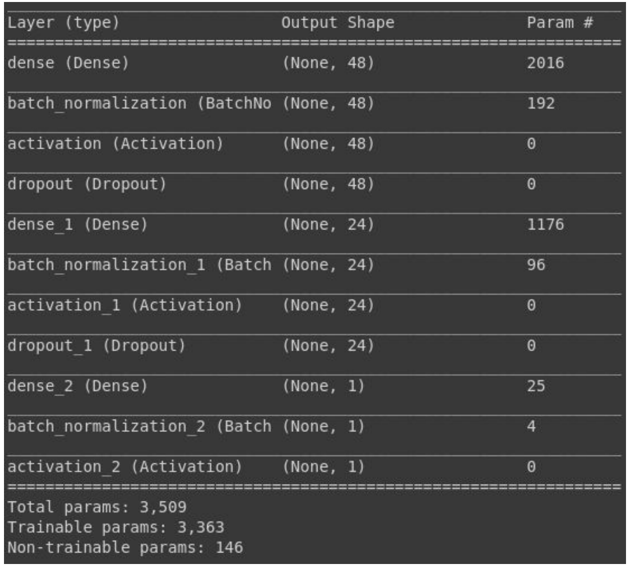
- **Baseline Models**
  - Linear Regression
  - Gaussian Naive Bayes Classifier
  - Multinomial Naive Bayes Classifier
- **Trees**
  - Random Forest Classifier
  - XGBoosted Trees



Random Forest

Modeling

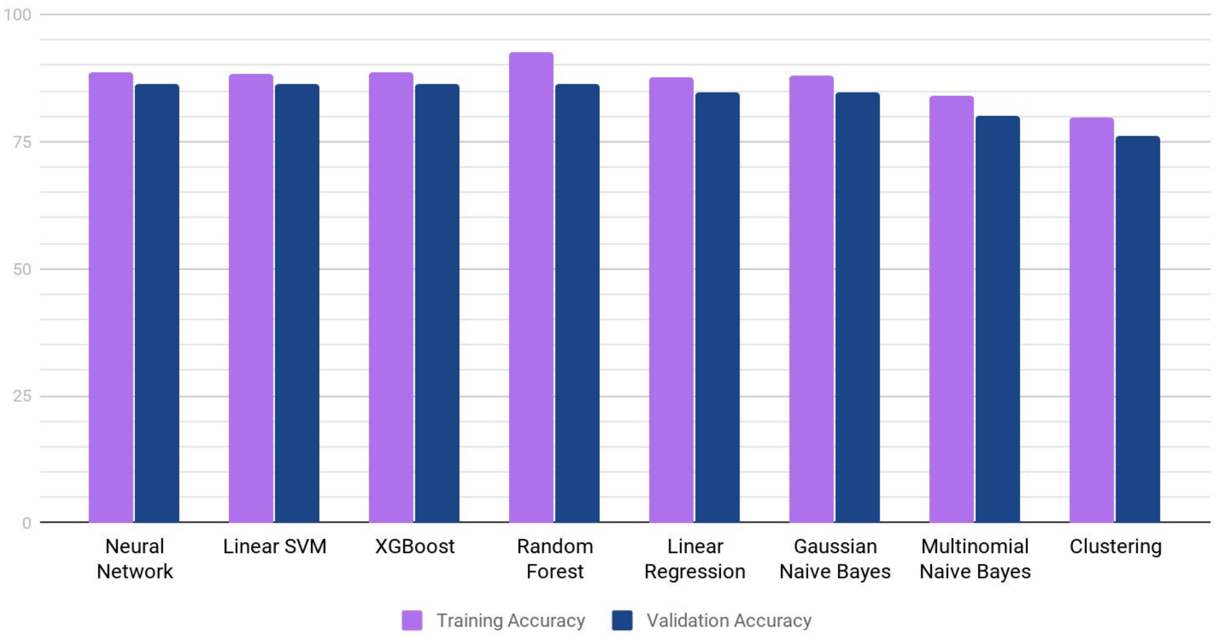
- **Support Vector Machines**
  - Linear SVM
- **Neural Networks**
  - DNN: 2 hidden layers, ReLU activation, dropout layers
  - DNN: 1 hidden layer, dropout layers, batch normalization, ReLU activation
- **Clustering**
  - KMeans

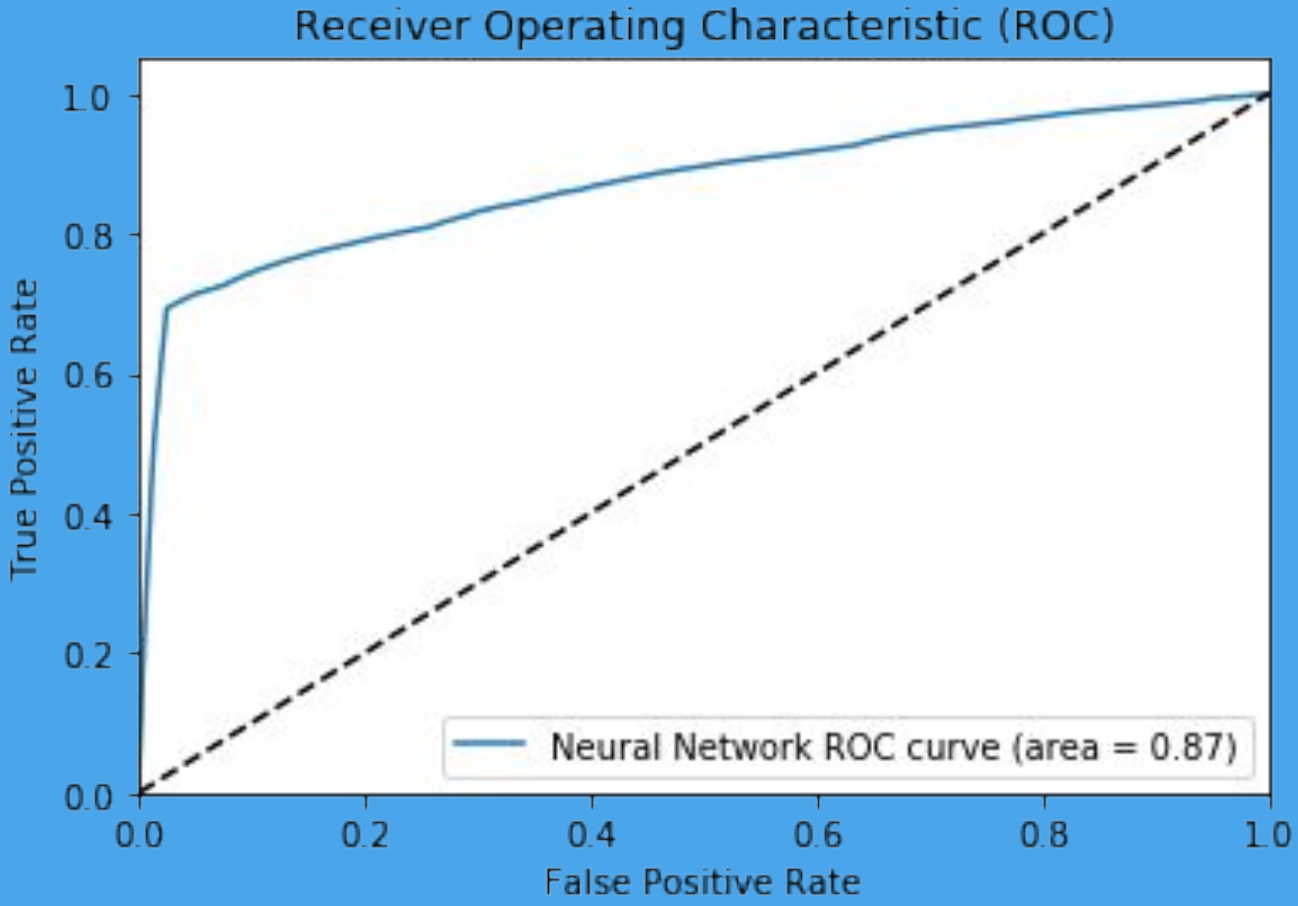


```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
dense (Dense)                (None, 48)                2016      
_________________________________________________________________
batch_normalization (BatchNo (None, 48)                192       
_________________________________________________________________
activation (Activation)      (None, 48)                0         
_________________________________________________________________
dropout (Dropout)            (None, 48)                0         
_________________________________________________________________
dense_1 (Dense)              (None, 24)                1176      
_________________________________________________________________
batch_normalization_1 (Batch (None, 24)                96        
_________________________________________________________________
activation_1 (Activation)    (None, 24)                0         
_________________________________________________________________
dropout_1 (Dropout)          (None, 24)                0         
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 25        
_________________________________________________________________
batch_normalization_2 (Batch (None, 1)                 4         
_________________________________________________________________
activation_2 (Activation)    (None, 1)                 0         
=================================================================
Total params: 3,509
Trainable params: 3,363
Non-trainable params: 146
```

DNN Model Summary

Model Accuracy




Receiver Operating Characteristic (ROC)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Neural Network ROC curve (area = 0.87)

Challenges

- **Data Preprocessing**
  - Large Data Set - Combined Training/Testing totals 33 GB
  - JSON columns in original data set
  - Processing missing values

- **Model Training**
  - Memory issues training certain models

Future Work

- **Ensemble Methods**
  - Combine models using weighted voting to create ensemble method
- **Develop Scalability**
  - Implement data preprocessing pipeline using Spark
- **Additional Data Preprocessing**
  - Attempt to extract and engineer more useful features from some JSON data

Thank you!