Predicting Waiting Time Duration in Digital Queues

Capstone Project Public Summary

by Rolibeth Arielle Ramos Miranda

Submitted to
Central European University
Department of Economics and Business

In partial fulfillment of the requirements for the degree of MSc in Business Analytics

Supervisor: Gergely Daróczi
Project Sponsor: Linistry

Budapest, Hungary
June 2020

CEU eTD Collection

Project Details

Background

Linistry is a disruptive queue management solution offered to companies with customer service offices and branch networks like banks, telekom companies, retail stores or even large events. With Linistry, people can join a queue virtually, from any location, book an appointment, or request a ticket on-sight via a digital, paperless kiosk. Paramount to a better customer experience is the ability to forecast waiting time. AI and Machine Learning solutions to predict waiting time are being explored to replace the current calculation methods being used.

Key Problems

Current process of estimating waiting time for digital queuing is calculated using an algorithm based on variables related to transaction and service type. Other factors that may affect waiting time are not being considered in the current model.

Objective

The goal is to build a model that can give more accurate predictions of waiting time, factoring in the data from Linistry such as historical transactions, information and attributes of clients, and other available data. After building the model, the estimated waiting time range will be optimized. The overall goal is to have high percentage of "correct predictions" (transactions with actual waiting time falling within the estimated time frame).

Methodology

Software and Tools

A local database was set up using DBeaver (running with SQLite). All data processing, including data cleaning, variable creation, merging, and initial data exploration was done locally using DBeaver. Modeling, additional data preparation, and model assessments were done in Python 3.7 using Jupyter Notebook (via Anaconda distribution). H2O Python module was used for modeling, to enable distributed processing and faster run time.

Data

All data used in the project are from Linistry. Tables are in Excel and rpt file formats and are all anonymized. The data ranges from June 2018 to February 2020. The main source table used for modeling has around 3 million records of service transactions from different companies. Prior to modeling, filters were applied and additional variables are calculated. The merged Analytic Base Table (ABT) was split into 50-25-25 for training, validation, and test.

The target variable for the model is the **waiting time duration.** This is calculated as the minutes between the time the customer is called and the time the customer signed up. The predictors used in the models are standardized, and various transformations were applied.

CEU eTD Collection

Modeling Approach

LASSO regression was used to select a subset of predictors from the initial set of variables. This method applies a penalty to the coefficients of the regression and shrinks some coefficients to zero.

Multiple models were built with different sets of hyperparameters. Both the complete set of predictors and the LASSO-selected predictors were used in the models. The algorithms used are Deep Learning (Neural Networks), Gradient Boosting Machines, Random Forest, Generalized Linear Models, and AutoML stacked ensembles.

The loss function used is the Root Mean Square Error (RMSE). This measures how close the model's predicted values are with the actual values of the target variable. The best model is selected based on the lowest RMSE.

Time Range Optimization

After building the models and selecting the best one, the next step is to determine the optimal rule in recommending the time range for the waiting time estimation. The estimated waiting period will be calculated as follows:

$$(Sign\ up\ time\ +\ predicted\ waiting\ duration - lower\ margin,\\ Sign\ up\ time\ +\ predicted\ waiting\ duration + upper\ margin\ )$$

Simulations are performed to choose the optimal lower and upper margins (in minutes). The percentage of actual waiting time falling within the waiting period range is calculated for all simulated values.

The final waiting period range estimates are rounded to the nearest 5 minutes.

Results

17 models using different algorithms were generated. Seed numbers were fixed and 5-fold cross-validation was applied. For models that can have an early stopping rule, stopping tolerance was set to 0.001 based on RMSE with a maximum runtime of 300 or 500 seconds. This means that the models will stop training when there is no difference in RMSE of at least 0.001 or when the run time exceeds 300 or 500 seconds, whichever occurs earlier.

The selected model is a Random Forest model with grid-search algorithm, using all predictors as input. This model has RMSE of 13.1 for train, 13.3 for validation, and 13.1 for test set.

Simulations were performed using different combinations of lower and upper margins (in minutes). Applying the selected margins on the estimated waiting time will result to about 93% correctly captured predictions.

CEU eTD Collection