Capstone Report for Data Science Specialization

Coursera & Johns Hopkins University

R. E., 22 November 2015

Abstract

This analysis is concerned with a predicting ratings of business based on Yelp tips from the Yelp Challenge Dataset. In this project, project use the Yelp Challenge dataset to model the relationship between tip text data and a business' rating, and successfully apply random forest model to predict a business' overall rating. In addition, a general linear regression model predicting the star rating was explored, with less successful results.

Introduction

This capstone report for the Coursera Data Science Specialization from Johns Hopkins University will answer the question: what kinds of tip features correlate most with business ratings. It is important for a business how it can leverage user tips to serve its customers better and, in general, can we predict the number of stars from 1 to 5 given to a business from the tip's text.

Methods and Data

The dataset was provided by Yelp. The Yelp dataset is divided into five parts: 'reviews', 'users', 'businesses', 'tips', and 'check-in'. For our predictive task, I decided to use 'tips' and 'businesses' because tips are short impactful statements about the business. The dataset was provided on an academic license agreement. All analyses performed are purely for academic purposes.

Analysis Software

Because the main software used in this specialization was R, I chose to perform the data processing, exploration and building the prediction model

using R packets and RStudio.

Data Processing

The data was originally in JSON form, in a total of 5 datasets connected by identifiers. The data contained tips from businesses in 10 cities around the world. The features of the tips opposite to reviews are similar for di???erent kinds of businesses.

Because research is dependent on using business IDs for the predictor, and not all businesses have tips, I decided to omit any business that do not have any tips in our dataset

Review Text Processing

After the subset of the data and specific variables of interest were obtained some processing of the text was necessary

Cleaning of the Tip Text

Using the 'tm' package, the sampled data is used to create a corpus. Subsequently, the the following transformations are performed:

convert to lowercase, characters /, @ |, common punctuation, numbers, strip whitespace, stemming (Porter's stemming).

Data Exploration

In the dataset, there are 495,107 tips total. There are 42,111 out of 61,184 businesses that contain at least one tip. Of those businesses that do contain tips, there are 11.76 tips on average for each business.

The average rating of all businesses is 3.67 which implies that there are more positively rated businesses. This most likely means that most tips will indeed be something positive about the business or restaurant. To get a better understanding of this, I visualized the most common words for all tip text below.

N-grams

N-grams models are created to explore word frequencies. Using the 'RWeka' package, unigrams, bigrams are created.

```
load("C:/Downloads/LearningR/CapstoneProjectR/tip_training.RDat
a")
```

```
library(tm)
```

```
## Loading required package: NLP
```

```
tip_text<-as.data.frame(tip_training$text)
docs <- Corpus(DataframeSource(tip_text))
docs <- tm_map(docs, content_transformer(tolower))
toSpace <- content_transformer(function(x, pattern) gsub(patter
n, " ", x))
docs <- tm_map(docs, toSpace, "/|@|\\|")
docs <- tm_map(docs, stripWhitespace)
docs <- tm_map(docs, removePunctuation)
docs <- tm_map(docs, removeNumbers)
docs <- tm_map(docs, removeWords, stopwords("english"))
library(SnowballC)
docs <- tm_map(docs, stemDocument)
library(RWeka)
Tokenizer <- function(x) NGramTokenizer(x, Weka_control(min = 1
, max = 1))
unidtm <- DocumentTermMatrix(docs,
                             control = list(tokenize = Tokenizer))
library(RWeka)
Tokenizer <- function(x) NGramTokenizer(x, Weka_control(min = 1
, max = 1))
unidtm <- DocumentTermMatrix(docs,
                             control = list(tokenize = Tokenizer))
tm_unifreq <- sort(colSums(as.matrix(unidtm)), decreasing=TRUE)
tm_uniwordfreq <- data.frame(word=names(tm_unifreq), freq=tm_un
ifreq)

library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
set.seed(39)
wordcloud(names(tm_unifreq), tm_unifreq, max.words=50, scale=c(
5, .1), colors=brewer.pal(6, "Dark2"))
```

The Model

The darasets ‘tips’ and ‘business’ were united on datasets with the fields:

‘business_id’, ‘user_id’, ‘text’, ‘likes’, ‘date’, ‘stars’.

cData was selected by randomly splitting the entire dataset into a 75% training and 25% testing dataset. The most popular words in tips based on training set of tip data are considered as the relevant features.
The validity of model’s predictions against the test set assessed using Mean Squared Error (MSE).

The first model I will look at is just a linear regressiom. This model will aim to answer the question: Is the review positive or negative based on the text? Unigrams occurring in more than 10% tips (100 unigrams) and bigrams occurring in more than 1% tips (50 bigrams) were token for the research.

With the top 100 (50) popular words (bigrams) , 101 (51) index feature vectors were created for each business. The feature vector includes the number of occurrences of each popular word in the tips for that business, followed by an offset term. General linear regression was performed on the training data to observe how each of these popular phrases affects the business’ overall rating, to correlate popular words in tips to the rating of businesses. The implementation of glm fared worse then second random

forest model.

Adding features such as bigrams did not increase the accuracy on the training dataset and significantly increased computation time so they were not included in the final model. Algorithm Random Forests with cross validation gave the most accurate results in the training dataset. The final model uses frequencies of top unigrams.

Results

Model Applied to a Specific Business

A model to predict the star rating of the business based on the trip's text was created as well, using the testing and training data. Random Forest and GLM were explored however Random Forest model with cross validation proved to be most accurate.

As an example of the output, the model was applied to the several types of business by means of 'grep' R-functions. Overall, it appears that the model does excellent in predicting the stars of the business. The importance of the variables in Random Forest model tuning by cross validation was estimated. Top-10 unigrams from tips are:

great, servic, love, best, awesom, bad, tri, amaz, food, friend, dont, chicken, delici, staff, locat, peopl, yum, excel

Evaluation of the Model

Applied to the test dataset, the Random Forest with CV model achieved mse 0.0005263199 on the testing dataset in predicting the star rating of the business based on the trip's text. The GLM achieved mse 0.3487754 on the testing dataset.

Discussion

Text analysis and prediction are very important parts of modern data science and has been used in many areas such marketing, writing, custom relations, business managment. In this research the star rating of business were successfully predicted and most important unigrams from trips were obtained.

In summary, the Yelp trip dataset was used to predict overall business star ratings.