Building a tool to assess business performance based on customer reviews

Jonathan Leslie

November 22, 2015

Introduction

With the increasing popularity of on-line crowd-sourced review databases such as Yelp, both consumers and business owners have access to a wealth of customer review data that was not possible a decade ago. For the business owner, this data represents a valuable resource, providing insight into customer satisfaction and business performance. While objective assessments such as star ratings and tick-boxes may give some idea as to customer satisfaction, they do little to provide real insight into exactly how a given business is meeting the expectations of its customers, or as the case may be, falling short.

To fill this gap, customers are encouraged to write reviews that appear as blocks of text. These give the consumer a chance to elaborate on his or her experience with more specific descriptions and they offer a wealth of information for the business owner. Unfortunately, reading customer reviews can be a long, painstaking process. For the business owner wanting to draw general conclusions from hundreds of such documents, the task can be arduous and prohibitively time-consuming.

This project is a feasibility study aimed at creating an application that can help streamline this process, making it easier for business owners to gain more insight into the satisfaction of their customers from Yelp reviews. By submitting a query by Yelp business ID, the application will return several pieces of information. The first is a plot of average star ratings (averaged monthly) over time, showing the general trend in the business' customer satisfaction. The second is list of words that are associated with the business' positive and negative reviews, using text mining algorithms and a user-defined threshold star rating (to distinguish "positive" from "negative" reviews). If, for example, a restaurant owner discovers that the word "wait" is often found in the negative reviews, he or she might infer that customers are often critical of how long they must wait for either seating or food/drink service. The goal of this project is to determine if such text-mining algorithms can be tuned to provide useful information to the business owner.

Methods and Data

Loading Data

The data for this study came from the Yelp Dataset Challenge (http://www.yelp.com/dataset_challenge). This dataset includes five JSON files corresponding to several aspect of Yelp review data and should be unzipped in the working directory. For this study, I used only the files pertaining to business IDs and reviews. I loaded the files of interest as below and saved them as rds files for subsequent analyses.

```
library(jsonlite)
library(readr)
library(tm)
library(dplyr)
library(zoo)
library(ggplot2)
```

```r
PATH <- "yelp_dataset_challenge_academic_dataset"
BusData <- stream_in(file(paste(PATH, "/",
                                "yelp_academic_dataset_business.json",sep = "")))
reviewData <- stream_in(file(paste(PATH, "/",
                                   "yelp_academic_dataset_review.json", sep = "")))
saveRDS(reviewData, "reviewData.rds")
saveRDS(BusData, "businessData.rds")
```

Becuase this is a feasibility study, I used only a subset of the data. In this case, I examined only restaurants and only those restaurants that had 100 or more reviews in the Yelp data. Here I save those data in a new data frame RestRevsHiCount.

```r
#Reload the data
BusData <- readRDS("businessData.rds")
ReviewData <- readRDS("reviewData.rds")

#Grab all businesses that have "Restaurants" in their category
Restaurants <- BusData[grepl("Restaurants", BusData$categories),]
#Grabbing all reviews of the restaurant businesses
RestRevs <- ReviewData[ReviewData$business_id %in%
                             Restaurants$business_id,]
#Selecting a shortlist of restaurants that have 50 or more reviews
Threshold <- 100
RestsHiCount <- Restaurants[Restaurants$review_count >= Threshold,]
#Gather all the reviews for those restaurants
RestRevsHiCount <- RestRevs[RestRevs$business_id %in%
                                    RestsHiCount$business_id,]
```

Examining changes in performance over time

A central tenant of business is to "know your audience". While a business may have met the demands of its audience well in the past, consumer taste changes quickly, and it can be hard to stay on top of that as a business owner. To assess how well a business is meeting the demands of its public, I wanted to examine ratings over time. To do this, I wrote a script that gathered review data for a given business (using the business_id field of the business data file, assigned to the variable BID (Business ID). This used only the business_id, review_id, date, stars and text variables of the review data file. Reviews were gathered by month using the as.yearmon() command and then averaged by month.

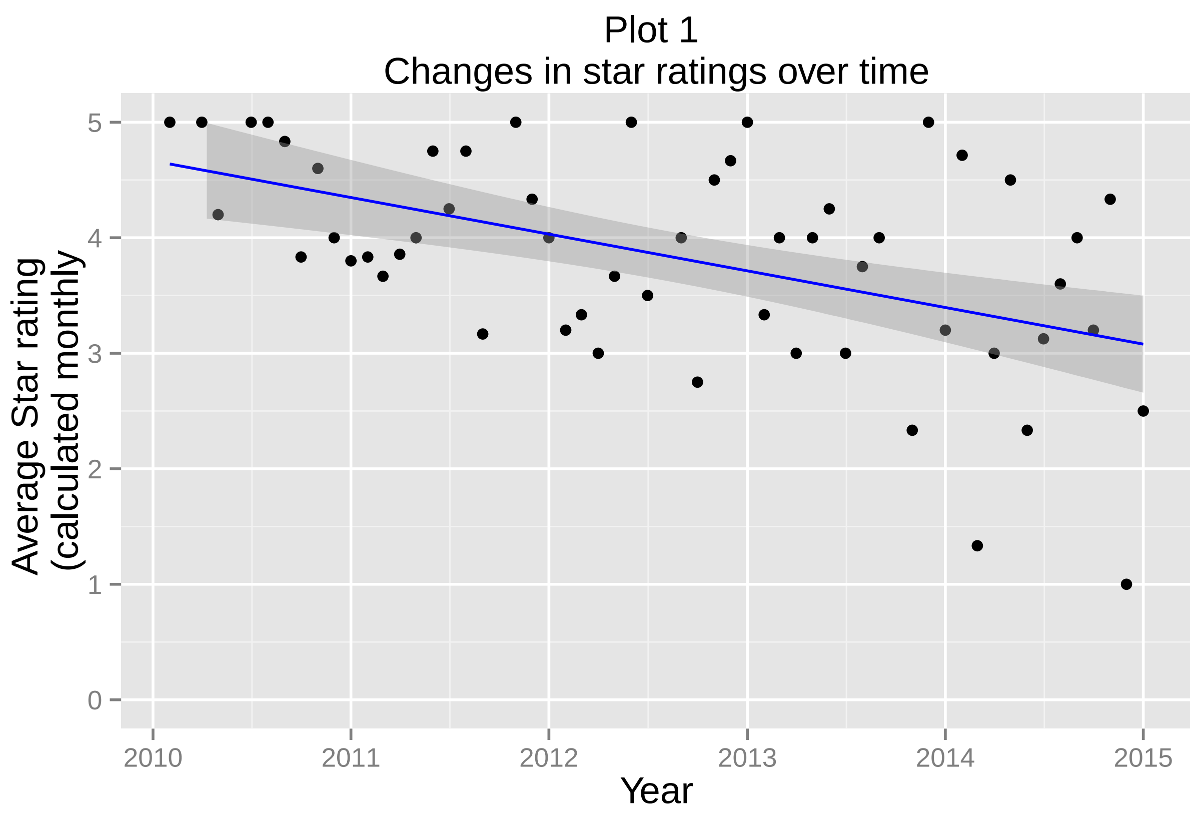
```r
#Will have BID defined here.
BID <- "27ADmOieSUZfbiU15als7w"

#This is where I put in the business ID of interest
#Here we build a dataframe with the monthly averages
#for stars from the BID
BIDRevs <- RestRevsHiCount[RestRevsHiCount$business_id == BID,]
#Make a new data frame with only stars and date data
BIDStarsDates <- select(BIDRevs, business_id,
                      date, stars, review_id, text)
#Add a variable 'YearMonth' that is only the month, ignoring days
BIDStarsDates$YearMonth <- strftime(BIDStarsDates$date, "%Y-%m")
BIDStarsDates$YearMonth <- as.Date(as.yearmon(BIDStarsDates$YearMonth))
#Group by business and YearMonth
```

```r
BIDStarsDatesGrouped <- group_by(BIDStarsDates, business_id, YearMonth)
#Calculate the average stars per month and discard RevID, Date, Text
BIDStarsDatesGrouped <- summarise(BIDStarsDatesGrouped,
                                  AvStarsPerMonth = mean(stars))
#Convert back into a data frame
BIDStarsDatesGrouped <- data.frame(BIDStarsDatesGrouped)
head(BIDStarsDatesGrouped)
```

```
##              business_id  YearMonth AvStarsPerMonth
## 1 27ADmOieSUZfbiU15als7w 2010-02-01        5.000000
## 2 27ADmOieSUZfbiU15als7w 2010-04-01        5.000000
## 3 27ADmOieSUZfbiU15als7w 2010-05-01        4.200000
## 4 27ADmOieSUZfbiU15als7w 2010-07-01        5.000000
## 5 27ADmOieSUZfbiU15als7w 2010-08-01        5.000000
## 6 27ADmOieSUZfbiU15als7w 2010-09-01        4.833333
```

To visualise the trend, the program plots average stars over time and draws a linear regression line. The slope of this line can be thought of as the trend in customer satisfaction.

```r
fit <- lm(AvStarsPerMonth~YearMonth, data = BIDStarsDatesGrouped)
```




```
## [1] "Change in star rating (stars per year):"
```

```
## [1] -0.32
```

As we can see in the plot, for this example restaurant, the reviews have gotten worse with time. The owner of this restaurant might want to know why. The next section deals with ways to help answer that question.

Generating word lists for good and bad reviews

To help shed light on factors that might be contributing to customer satisfaction (either for the good or the bad), I wanted to emply text-mining techniques on the texts that accompany each review. I hypothesize that such an analysis might reveal trends associated with poor customer satisfaction.

Since each business owner may want to decide what star rating differentiates between a "good" and a "bad" review, I allow this threshold to be set. In this case, it is set with the StarsThresh variable. In this example, StarsThresh is set to 3.5, so reviews receiving 1, 2 or 3 stars will be pooled in the "bad" group, and the rest will be considered "good".

```
StarsThresh <- 3.5
```

```
## [1] "Keywords associated with good reviews:"
```

```
##     sushi      roll     place      good      like     order       eat
##      2.08      1.73      1.25      0.89      0.66      0.62      0.57
##     great       can       ayc      food      time       one       sin
##      0.56      0.51      0.49      0.49      0.46      0.43      0.43
##       tri      citi     fresh      just      come    servic      back
##      0.42      0.38      0.38      0.38      0.37      0.37      0.34
##       get      love      best    friend      make    realli      also
##      0.34      0.34      0.33      0.33      0.32      0.32      0.31
##      chef      fish   favorit      will      menu      even     alway
##      0.31      0.30      0.29      0.27      0.26      0.25      0.24
##   definit      much      nice    nigiri    oyster      want       bar
##      0.23      0.23      0.23      0.23      0.22      0.22      0.20
##   everyth      went    pretti     rolls     staff       top      came
##      0.20      0.20      0.19      0.19      0.19      0.19      0.18
##      give      mani     staci      tuna      well     everi     peopl
##      0.18      0.18      0.18      0.18      0.18      0.17      0.17
##      rice    salmon       ask      babi     first    mussel       new
##      0.17      0.17      0.16      0.16      0.16      0.16      0.16
## recommend     right   service      sinc     still      tast    though
##      0.16      0.16      0.16      0.16      0.16      0.16      0.16
##       fri       got      last      look     never     price       say
##      0.15      0.15      0.15      0.15      0.15      0.15      0.15
##     think    around    better    dinner     green     night      seem
##      0.15      0.14      0.14      0.14      0.14      0.14      0.14
##     thing      town    vampir      wait      made   qualiti    review
##      0.14      0.14      0.14      0.14      0.13      0.13      0.13
##      tako   amazing      bake      crab delicious     enjoy      hand
##      0.13      0.12      0.12      0.12      0.12      0.12      0.12
##      know       lot   shooter     vegas  friendly     worth
##      0.12      0.12      0.12      0.12      0.11      0.11
```

```
## [1] "Keywords associated with bad reviews:"
```

```
##      sushi      order       roll       food      place       time
##       2.52       1.58       1.40       1.13       1.13       1.00
##        eat     servic       just        get       good       rice
##       0.95       0.90       0.82       0.81       0.73       0.71
##       come        one        can       like       wait       tabl
```

```
##       0.69       0.63       0.60       0.58       0.56       0.55
##       will       came        ask       back        got     realli
##       0.55       0.52       0.48       0.48       0.47       0.47
##   waitress       even     shrimp       fish     server       took
##       0.47       0.45       0.45       0.42       0.42       0.40
##       want        ayc       chef      first       walk        bad
##       0.40       0.37       0.37       0.37       0.37       0.34
##     better      check      never     review       said        tri
##       0.34       0.32       0.32       0.32       0.32       0.32
##      great       know    service       told     friend       give
##       0.31       0.31       0.31       0.31       0.29       0.29
##      minut       love        sit       tast      final       left
##       0.29       0.27       0.27       0.27       0.26       0.26
##      still        two       ever     experi       much    restaur
##       0.26       0.26       0.24       0.24       0.24       0.24
##      rolls       best     custom       menu    probabl        say
##       0.24       0.23       0.23       0.23       0.23       0.23
##       serv       slow       star      fresh       hour       look
##       0.23       0.23       0.23       0.21       0.21       0.21
##        lot        now      peopl       sinc      three       also
##       0.21       0.21       0.21       0.21       0.21       0.19
##     around        bar       made       make       mani       mayb
##       0.19       0.19       0.19       0.19       0.19       0.19
##  recommend     return      sever       well      worst      anyth
##       0.19       0.19       0.19       0.19       0.19       0.18
##     dinner       long     pretti    qualiti       seem        sin
##       0.18       0.18       0.18       0.18       0.18       0.18
##      start       take       tell      anoth    everyth       last
##       0.18       0.18       0.18       0.16       0.16       0.16
## restaurant        sat        way        ate       citi      clean
##       0.16       0.16       0.16       0.15       0.15       0.15
##      coupl        end       full    horribl    however       need
##       0.15       0.15       0.15       0.15       0.15       0.15
##       nice     reason       sure      table      think       town
##       0.15       0.15       0.15       0.15       0.15       0.15
##       went       away      decid       dine     finish       hand
##       0.15       0.13       0.13       0.13       0.13       0.13
##       hope     person     someth      usual       bare       feel
##       0.13       0.13       0.13       0.13       0.11       0.11
##       item      plate        see
##       0.11       0.11       0.11
```

Results

Discussion