Title: Using Latent Semantic Analysis and Latent Dirichlet Allocation to Identify Similar Yelp Reviews

Philip Coyne

November 4, 2015

Introduction

The Yelp Dataset Challenge offers an abundance of information for data scientists. From user and business information to check-in metrics, the data set provided is a mere fraction of what is available. As a guiding hand for interested enthusiast, Yelp suggested *"By adding a diverse set of cities, we want participants to compare and contrast what makes a particular city different"*. This leads to the question of how to accomplish such a goal? In my analysis, I attempt to perform text mining and analysis to Yelp Reviews based on the type of business in a particular city. The question I hope to address is if it is possible to utilize text analytics to focus on reviews that are similar and if so, make it easier for a human to identify. Ultimately ***is it possible to simplify categorize and easily identify reviews based on text analysis***?

Methods & Data

The analysis is broken into 3 parts: Data Extraction and Cleaning, Text Mining and Analytics, and Text Modeling. For the sake of space, code will not be shown in this report. If one wishes to view the source code, please follow this link: https://github.com/gbex384/Capstone-Project.

Data Extraction & Cleaning

The data given was downloaded and unzipped to a local computer and read into a local R workspace from the JSON data format.The data sets came in a total of 5 different JSON files. The ones that were utilized for thsi analysis were those of business information and review information. In the interest of keeping the focus of this analysis to a conveivable limit, a majority of this analysis focuses on the Business categories of *Health And Medical* and *Doctors*. Therefore, any business that has the field *Health and Medical* or *Doctors* is copied from the R Object containing business information and placed a new dataframe, called **doctorsTest**.
The review ID and review text are then extracted from the R object containing review information. Any review pertaining the business IDs were extracted and merged into a new data frame called **doctorsReviews**. **doctorsReviews** includes review ID's, text, business ID's, names, City locations, and stars received.
Afterwards, each city was filtered into individual R Objects (Ex: businesses in Pittsburgh were assigned to an object **Pittsburgh**). Due to space limitations, we will discuss the data and results of **Pittsburgh**.

Text Mining and Analytics: Wordclouds

Pittsburgh was selected because it has approximately 34 reviews and will have a small enough scope to describe methods and results. Exploratory analysis started by importing all reviews into an R Object of class Corpus.

```
inputText<-Pittsburgh$Review_Text
options(stringAsFactors=FALSE)
df.corpus <- Corpus(VectorSource(inputText))
```

Next, a user defined function *clean()* is used to remove frequently used terms that have no connotation or meaning; this is done by passing **df.corpus** and **myStopwords** to the function. Terms are also reduced to lowercase, numbers and punctuations are removed, as is the resulting whitespace. The resulting R Object is **df.corpus.clean**, which is passed to *DocumentTermMatrix()* and *TermDocumentMatrix()*. ***Latent Dirichlet Allocation*** and ***Latent Semantic Analysis*** depend on both Document Term Matrices and Term Document Matrices, respectively. These will be the two models used in this paper, and will be discussed in more detail later.

```
df.corpus.clean <- clean(df.corpus, myStopwords)
dtm <- DocumentTermMatrix(df.corpus.clean);
tdm<-TermDocumentMatrix(df.corpus.clean);
```

R Objects *tdm* and *dtm* are Term Document and Document Term Matrices, respectively. *tdm* contains 34 columns, one representing each document (in our case, each review for Pittsburgh), and is against 1170 rows of terms found in the document. *dtm* is *tdm* transposed. Both matrices have a sparsity (the amount of zeroes within the matrices) of 94% and must be reduced. For this, a sparsity of 0.8 was chosen empirically ; going beyond 0.8 didn't change the sparsity enough for the next portion of our analysis, and going below 0.8 removed too much data.

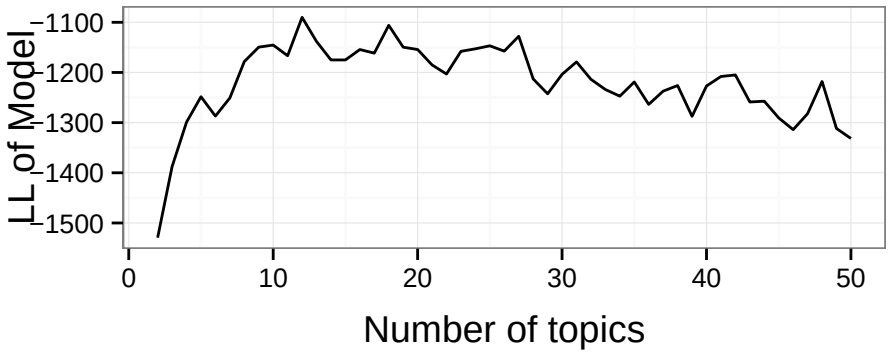
```
dtm.sp <- removeSparseTerms(dtm, sparse=0.80)
tdm.sp <- removeSparseTerms(tdm,sparse=0.80)
dtm.mat<-as.matrix(dtm.sp)
tdm.sp<-as.matrix(tdm.sp)
```

The matrices are characterized through way of word cloud. A word cloud displays terms that are most frequent in the dtm or tdm, giving an idea of what people will be talking about in these reviews.

The left wordcloud contains a large frequency of the term “eye”, “appoint”, and “exam”, leading us to believe that the reviews medical professionals are focused primarily on optometrists. However this is something anyone can infer from looking at the data set that has been put into **Pittsburgh**. More so, the fact that “eye”,“exam”, and “appoint” are so frequent, it leads one to believe that perhaps these words are appearing too frequently and are skewing the data.
A second wordcloud is generated; removing words that appear in more than 99% of documents (as well as words that appear in less than 1% of documents) from our matrices. ***Grimmer and Stuart (2013)*** among other researchers do this to find more meaningful information in the matrices (since its abundantly clear that these reviews are about optometrists and the quality of the staff and visits are being held in question).
In this second word cloud the words “time” “service” and “care” are more frequent, which sheds a bit more light onto the situation: these reviews will likely have more emphasis on the care given, the amount of time during visits, and the service provided. While this should be clear for any person in society, recent viewings of reviews have raised this ones doubts of what people actually value.

Text Mining and Analytics: Latent Dirichlet Allocation

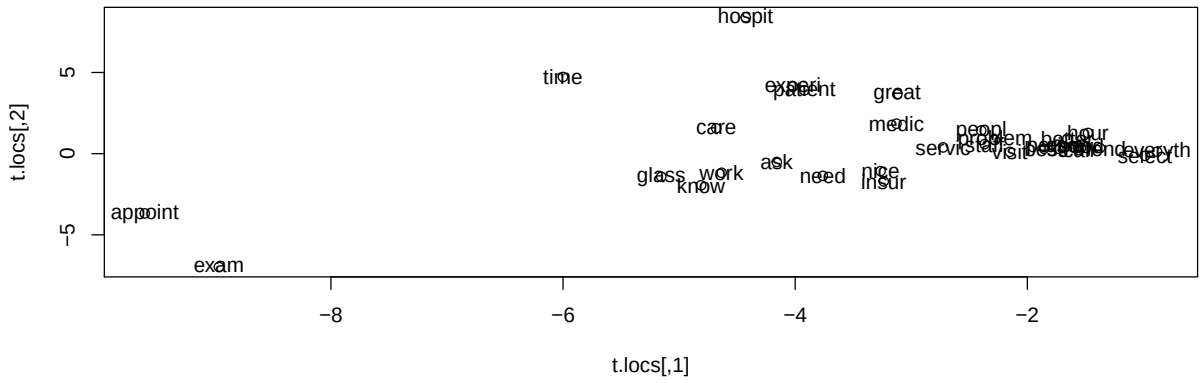
Latent Dirichlet Allocation is a means of attempting to label and sort documents by way of topic modelling. The idea is that every document is primarily associated with some concept, and that can be inferred by the words used in the corpus. In order to accomplish this, a “bag-of-words” approach is used, in which the entire documents structure is effectively scrapped, and every word is looked at without context to the ones before and after it. ***Grun and Hornik (2011)*** notes that if we fit an LDA model to *dtm.sp* and plot a log-likelihood plot against number of topics, we can determine the optimal number of topics to begin labelling documents. For the sake of consistency, set.seed(4) was used.



An ideal loglikelihood chart would appear to be much like a log function. Despite this non-ideal plot we can still determine the number of topics that would be optimal by selecting where the graph would appear to level out. For this analysis, the point before the plot becomes eratic is chosen; the number of topics is 9.

Text Mining and Analytics: Latent Semantic Analysis

Latent Semantic Analysis is a mathematical method for representing documents and determining how closely related the two documents are in vector space. LSA uses the “bag-of-words” approach, and is unique in that it reduces dimensionality of a document matrix.



One can see clusters of words from the documents. From this, we will assume that these clusters can function as categories, that we will use to perform another form of LSA.

Cluster 0 serves to contain any outliers: Time, Hospit, appoint, exam, care Cluster 1 holds three words: Experi, Patient, Great. Cluster 2 holds 16 words: Medic, Peopl, hour, problem, better, staff, service, visit, call, everyth, select, best, friend, give, good, person. Cluster 3 holds seven words: Ask, nice, need, insur, work, glass, know.

Results

Combining the Results

LDA Results

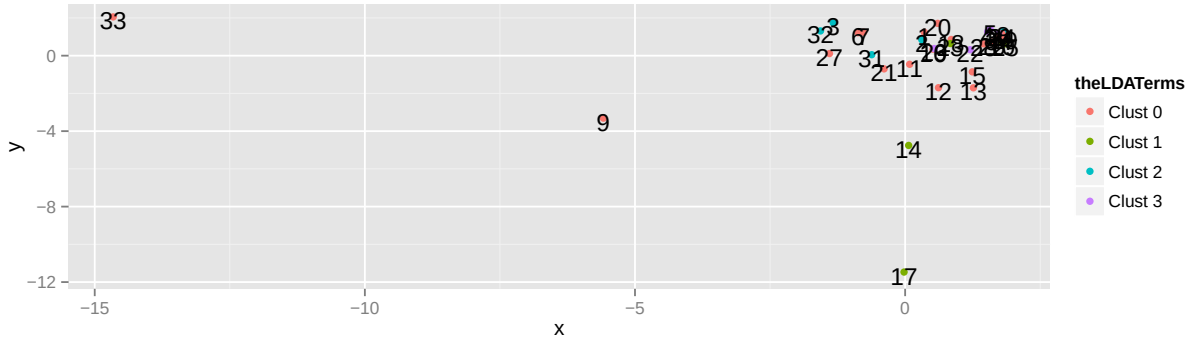
The LDA topics are displayed below. The nine topics are labeled with words found in the documents. This represents which word ranks highest in a topic; Topic 1 is most associated with the word "time", second most with "great", third most with "everything", etc.

```
##      Topic 1   Topic 2  Topic 3 Topic 4  Topic 5 Topic 6   Topic 7
## [1,] "exam"    "experi" "ask"   "best"   "know"  "appoint" "servic"
## [2,] "appoint" "hospit" "give"  "friend" "work"  "glass"   "patient"
##      Topic 8 Topic 9
## [1,] "time"  "care"
## [2,] "great" "peopl"
```

Determining the accuracy of these topic models has been empirical through trial-and-error to adjust **myStopwords**. The results above have shown approximately a 72% accuracy on average for the documents associated with each topic. That is to say, 72% of documents were closely associated with the Topics they were assigned. From reading ***Anaya, Leticia (2011)***, 72% accuracy for LDA is found to perform well, and continued the analysis.

LSA Results

Leveraging the cluster information gathered in the previous section, one can attempt to associate topics with word clusters. Assuming that the highest ranked word in each topic represents the topic more than other words, we can then associate clusters with each topic.



The above plot was utilized by passing the transposed ***tdm.sp*** object into the lsa() function, with default parameters. The resulting matrix was passed to a cmdscale() function, with 'eig=TRUE' and 'k=2'.

Clusters 0 through 3 were determined by passing a vector into a dataframe and running a for loop to categorize words into their appropriate word clusters, and then categorize topics with the corresponding word clusters. The result displays an LSA plot, each point representing a review from the **Pittsburgh** data frame. Each document is colored according the the word cluster they are associated with.

Discussion

The LSA plot displayed shows us how alike, or disalike, documents are. Manipulating RStudio, a better picture can be ascertained, and we can see that the dense cluster in the upper right quadrant is a large grouping of *paired off* documents. From the image here, the most evident are documents 21 and 11, with review_ids bChO4093ZsHDfCMZvOmu8A and 0QYWzxvuP5oedJR9qMQDRQ. These reviews are for the University of Pittsburgh Medical Center Eye Center and Sports Medicine,respectively. Both given lengthy reviews, and the businesses 5 stars.

Reviews 3 and 32, with ids of rxK0wsHd_3XqxhlCuqPOgg and bAly6pKNVQmTxU3___mYtkw. These reviews are of similar length, and give the businesses 4 stars. Outliers 33, 9, 14, and 17 are ***massive*** in length, and vary in their ratings. Unfortunately, pairs like 12 and 13 are very different; one having a short sentence review, while the other has a glowing paragraph by paragraph review.

Various researchers have indicated in their papers that this field will always require a human for verification that the analysis works. However, 100% accuracy is not expected, and accorind to more accomplish researchers, 60% accuracy in LSA or LDA is ideal.

This method of approaching this data is a helpful tool for analysts to identify and focus on documents that are similar and alike to each other, in addition to identifying documents that are disimilar. The next step is to continue to analyze the Yelp data set and continue to find interesting cultural information in the text.

References

Source code can be found at https://github.com/gbex384/Capstone-Project

Gruen B, Hornik K. 2011. topicmodels: An R package for fitting topic models.Journal of Statistical Software.

Grimmer J, Stewart BM (2013) Text as data: The promise and pitfalls of automatic content analysis methods for political texts. Political Analysis.

Anaya, Leticia. 2011. Comparing Latent Dirichlet Allocation and Latent Semantic Analysis as classifiers PhD dissertation, Department of Information and Decision Sciences, University of North Texas.