Final Paper

Introduction

Whenever a person hears about a business or think about starting their own, they are always told that a business’s success is all about location, location, location. It’s hard to quantify how much of a business’s success is determined by its location, but we were curious to see if location was linked to higher star ratings on Yelp. Yelp is a service that publishes crowd-sourced reviews about local business online. Additionally, we wanted to see if there was any relationship between certain attributes of restaurants and their ratings on Yelp.

Aside from analyzing the success and behavior of businesses, we also decided to explore the relationship between the text in reviews and the number of stars it received. In our project, we used a deep dataset of Round 7 of the Yelp Academic Dataset challenge, available at: https://www.yelp.com/dataset_challenge/dataset. The dataset includes five json files of business, check-in, user, review, and tip data, but our research only involved looking at the business, user, and review datasets.

Exploring the Data

1. Data Properties

The dataset includes 2.2 million Yelp reviews for 77,455 businesses across the United States. To gain access to the dataset, a name and email had to be provided and the terms of use had to be accepted. The schematic for the business, user, and review datasets were:

{ ‘type’: ‘business’, ‘business_id’: (encrypted business id), ‘name’: (business name), ‘neighborhoods’: [(hood names)], ‘full_address’: (localized address), ‘city’: (city), ‘state’: (state), ‘latitude’: latitude, ‘longitude’: longitude, ‘stars’: (star rating, rounded to half-stars), ‘review_count’: review count, ‘categories’: [(localized category names)] ‘open’: True / False (corresponds to closed, not business hours), ‘hours’: { (day_of_week): { ‘open’: (HH:MM), ‘close’: (HH:MM) }, … }, ‘attributes’: { (attribute_name): (attribute_value), … }, }

{ ‘type’: ‘review’, ‘business_id’: (encrypted business id), ‘user_id’: (encrypted user id), ‘stars’: (star rating, rounded to half-stars), ‘text’: (review text), ‘date’: (date, formatted like ‘2012-03-14’), ‘votes’: {(vote type): (count)}, }

{ ‘type’: ‘user’, ‘user_id’: (encrypted user id), ‘name’: (first name), ‘review_count’: (review count), ‘average_stars’: (floating point average, like 4.31), ‘votes’: {(vote type): (count)}, ‘friends’: [(friend user_ids)], ‘elite’: [(years_elite)], ‘yelping_since’: (date, formatted like ‘2012-03’), ‘compliments’: { (compliment_type): (num_compliments_of_this_type), … }, ‘fans’: (num_fans), }

2. Data Pre-Processing

Data pre-processing were the steps taken to collect and prepare the input data for data mining and visualization.

Stage One: JSON to CSV

Converting the raw data from JSON was quite easy using R libraries. We began by reading in the data and writing the tidied data-frame to CSV objects to be read in our analyses. Once this was completed, other R libraries were used to wrangle and tidy the data for their final transformations.

Stage Two: Data Wrangling

We began by filtering for businesses with more than 300 reviews in the United States. Looking more closely at the business dataset, we noticed a nested data-frame of attributes for each business, so we cleaned and gathered the data so that each row contained one business attribute. Business data set was used to see what makes a business popular and successful; especially by looking at where most-reviewed are clustered and the influence of major business attributes.

Review data was cleaned for sentiment analysis using a library called "tidytext". After reading in the review dataset, we randomly sampled 500,000 observations. With this, we tokenized each word in a review into a single row in the data-frame and removed stop words like "a", "and", "the", etc. since they had no indication of the reviewers' attitudes. Using built-in lexicons, we assigned each word a valence score and computed the mean score for each review. Additionally, we created per-word summaries, computing the number of business and reviews the word appeared in and the average star rating based on each review. We filtered for words that appeared in more than 10 businesses and 500 reviews to exclude rare words with strange valence scores and ratings. Using these cleaned datasets, we were able to create visualizations and make inferences discussed later.

Lastly, business data, user data and review data were all used to analyze the users' behaviors when they are in unfamiliar places. User data was cleaned to filter for active users with a significant amount of reviews. Cleaned review data and business data were then joined together with user data. We created an additional boolean column called 'most_reviewed', after computing the most familiar location for a user (one state per user where most of his/her reviews reside). Using this, we cleaned up and created a new data frame with average star ratings of familiar places and unfamiliar places per user. In the next section, we discuss this further along with results.

Stage Three: Pre-Processing using 'tidytext' for Sentiment Analysis

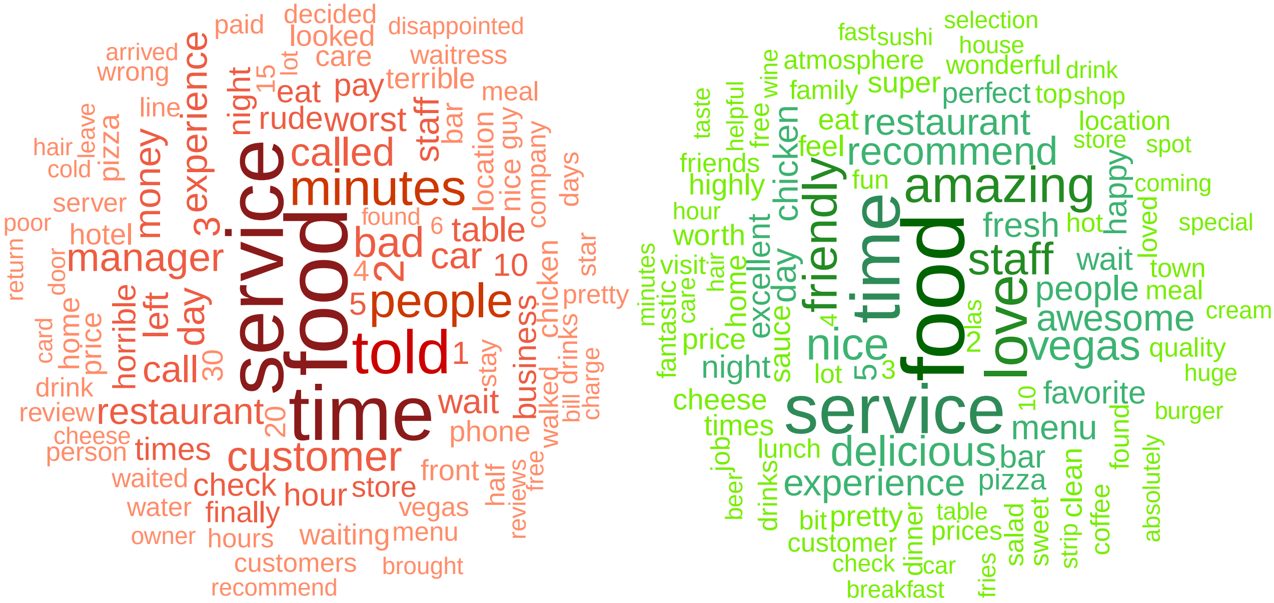
Sentiment analysis, or opinion mining, makes use of natural language processing to analyze pieces of text and determine the attitude of the speaker or writer. To perform sentiment analysis, we researched various text mining libraries. First, *tm* was used to remove stop words, common English words that would give no information, but the process of splitting text reviews into one word-per-row and removing stop words was too inefficient. After more research, we came across "tidytext", a library for text mining. Using this library, cleaning and aggregating the review data was extremely easy and efficient. We then performed tidy analysis to strip reviews and tokenize each word into a single row in the data-frame.

To explore this data, we researched different lexicons and found ANEW (Affective Norms for English Words), a "sentiment lexicon" that scores words for valence. However, upon further research, we came across an article written by Finn Arup Nielsen, where he compares a sentiment lexicon he created to ANEW. His lexicon, AFINN, contains a list of English words that are assigned an integer value from -5 (negative) to +5 (positive) based on the word's sentiment. He concluded that his word list better classifies sentiment rating than ANEW on Twitter sentiment analysis. Since Yelp text reviews are similar to tweets, we used AFINN for this sentiment analysis.

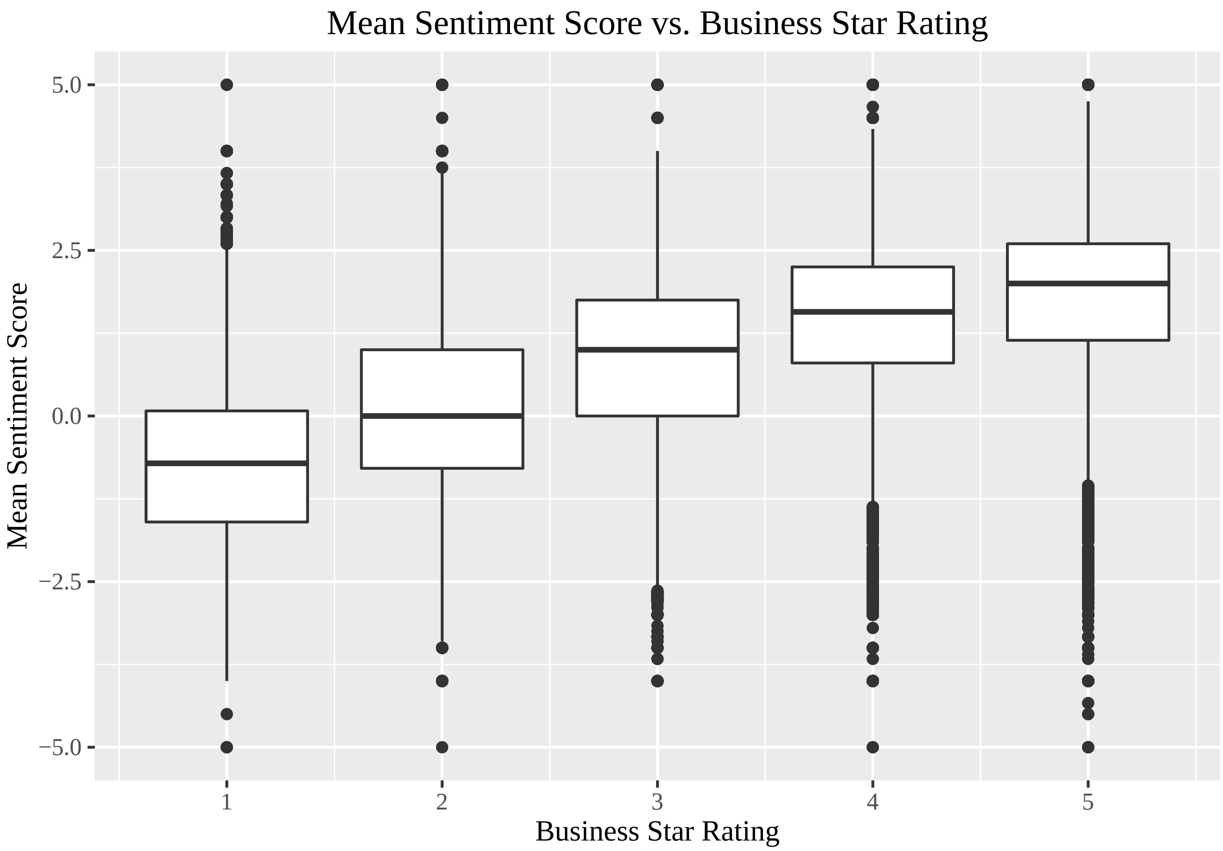
Results and Discussion

B. Sentiment Analysis

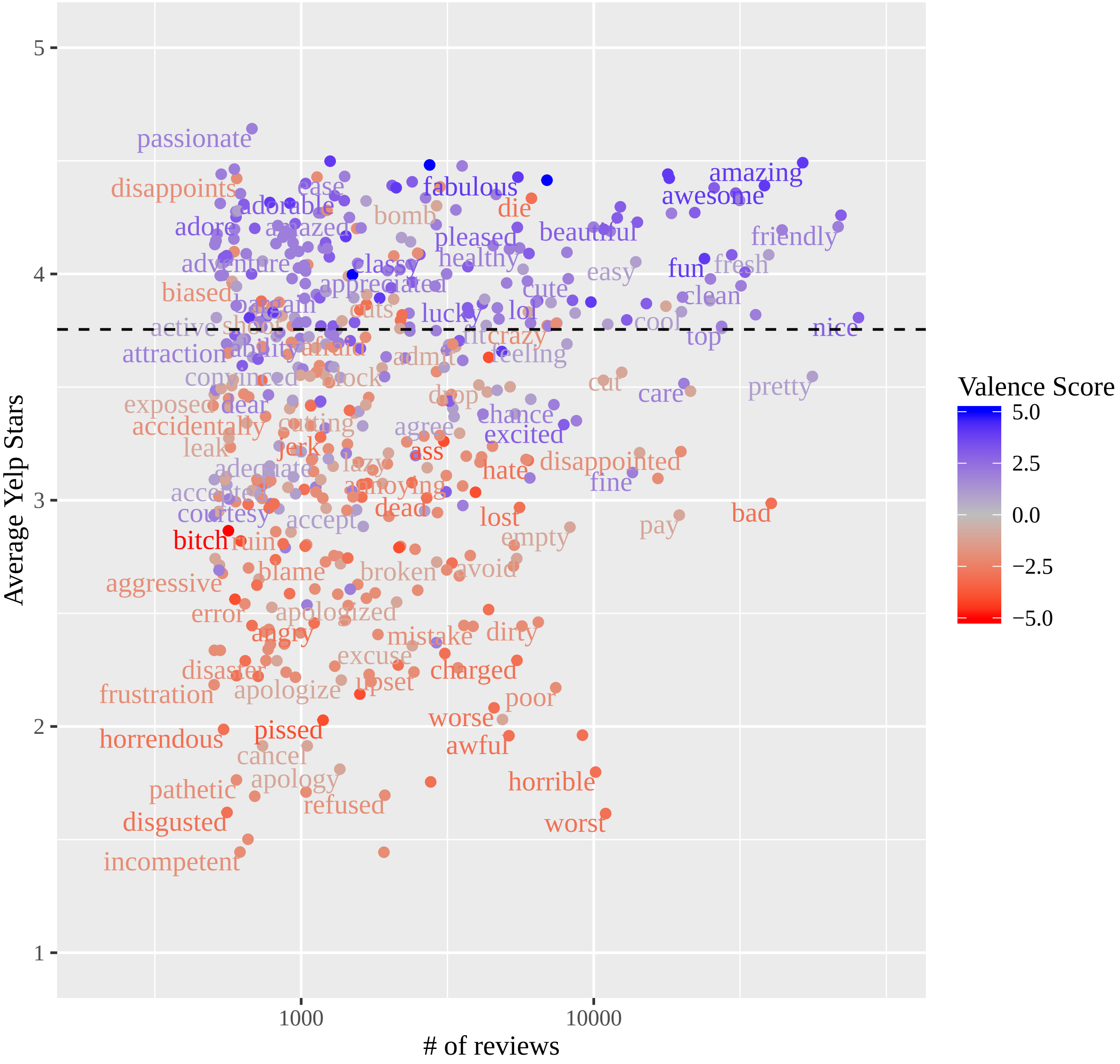
We hypothesized that words with higher valence scores would produce higher Yelp star ratings for businesses. First, using the R library "wordcloud", we formed two word clouds for the 100 most frequent words in text reviews with one-star and five-star ratings to determine if there was a distinct difference between the words used.



While words like “food” and “service” were the most common, there is a noticeable difference between the two word clouds generated. To further explore this difference, we generated boxplots of mean sentiment score versus business star rating to crudely determine if there was a positive trend for valence score and business rating.



There was indeed a positive correlation, despite some words with high valence scores appearing in reviews with low ratings and vice versa. Now to determine which words were indicative of a positive review and a negative review and the accuracy of our valence score classifier, using the word summaries data we generated a scatterplot labeled by word, colored by AFINN score, and plotted by average star rating versus word frequency. Additionally, we overlaid a horizontal line of the mean restaurant rating as a metric to determine the accuracy of our classifier. We would expect all words above to be blue and all words below to be red.

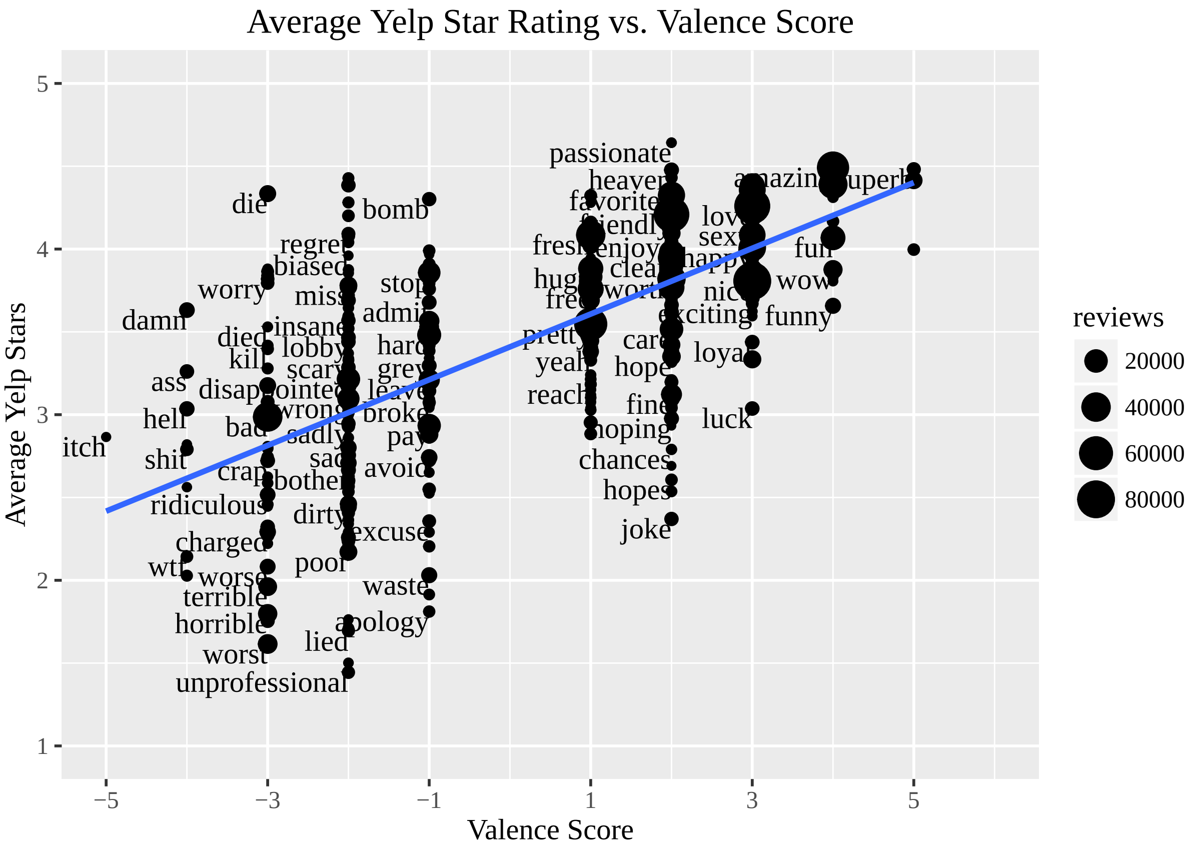


```
## [1] "Positivity Accuracy: 0.650190114068441"
```

```
## [1] "Negativity Accuracy: 0.865853658536585"
```

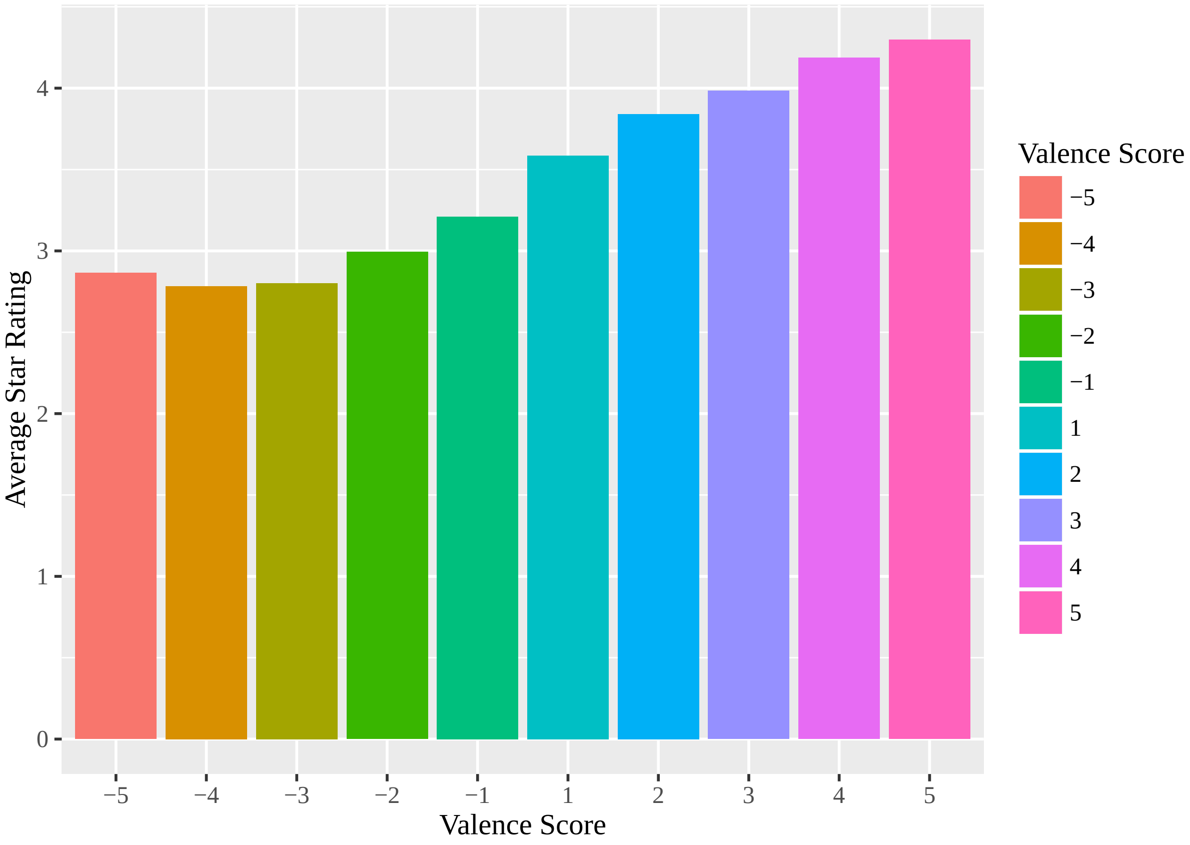
For the most part, our classifier was accurate. 65.02% of positive words sampled lied above the mean restaurant rating, and even better, 86.99% of negative words lied below the mean star rating. Therefore, it seems reasonable to infer that there is indeed a positive correlation between valence score and Yelp business rating.

However, we also wanted to see these words plotted by their valence score versus average star rating to visualize the actual trend between sentiment and rating, rather than just viewing what percentage of words lied above the mean rating.



Based on a simple linear regression line, words with higher scores are clustered towards 4 and 5-star ratings, while most negative words have pretty low average ratings. Some words like “damn” had higher star ratings despite their low AFINN score, but that could be attributed to writers’ different expressions like “damn good” for really good food.

Lastly, we wanted to see what the mean rating was for each AFINN score since that would clearly indicate a correlation.



As expected, higher AFINN scores did in fact result in higher mean business ratings.