GutenbergReads

Brian O'Grady

May 10, 2019

Introduction

The goal of this project is to examine the relationship between books and their online reviews, using Gutenberg.org as the source for online texts and Goodreads.com as the source for their reviews. The initial objective was to see if I could arrive at some sort of statistical/machine learning analysis in this relationship, but we will see throughout this paper why this was not possible. Instead I chose to focus on providing context for the data and understanding the limitations of the methods we learned in this class. Review data is complex enough as it is; however, it seems book reviews are a separate breed from those we read on Amazon, Yelp, etc.. We will see some examples of reviews later on.

I was able to successfully pull in the data from the Goodreads.com API and gain some good experience there, as my background is not in working with web data or development. Getting this code right proved to be the biggest challenge of all, as we had already worked with Gutenberg.org books earlier in the class. I did consider involving some more sophisticated natural language processing techniques such as lemmatization but abandoned these endeavours due to the timeline and decided instead to focus on what has been the overarching purpose of the course: wrestling with APIs and pulling the data into R, wrangling and cleaning the data, and doing some data analysis on the output.

Data

As stated above, I used Gutenberg.org and Goodreads.com as my data sources in this project. Let's take a closer look at the details of both.

Gutenberg.org

Over 58,000 eBooks are available for free download on Gutenberg.org. The Gutenberg Project focuses on collecting "older works for which U.S. copyright has expired" to be made available for public use. On their website they have a Top 100 list which links to the top 100 most downloaded books from Gutenberg, and it is from here that I have made my selection for books. The books I chose are: Pride & Prejudice by Jane Austen; Frankenstein by Mary Shelley; Et dukkehjem (A Doll's House) by Henrik Ibsen; The Importance of Being Earnest by Oscar Wilde; A Tale of Two Cities by Charles Dickens; The Strange Case of Dr. Jekyll and Mr. Hyde by Robert Louis Stevenson; Alice's Adventures in Wonderland by Lewis Carroll; Dracula by Bram Stoker; The Adventures of Sherlock Holmes by Arthur Conan Doyle; and The Awakening and Selected Shorts Stories by Kate Chopin.

Goodreads.com

The world's largest site for readers and book recommendations, Goodreads.com has a large following amongst the Internet-literate readers of the world. According to their website, you can:

- See what books your friends are reading
- Track the books you're reading, have read, and want to read
- Check out your personalized book recommendations
- Find out if a book is a good fit for you from [their] community's reviews

Using their API I chose 5 reviews for each book (we will see why only five in the next section) from Gutenberg, totaling to 50 reviews.

API Limitations

Gutenberg.org

The Gutenberg Project explicitly states that the site is explicitly intended for human users and that "[a]ny perceived use of automated tools to access this website will result in a temporary or permanent block of your IP address", according to their Terms of Use page. They say you can use a mirror site, but I decided to select a few books manually rather than trying to connect to the mirror given the other troubles I was having with the Goodreads.com API.

Goodreads.com

There were several limitations for the Goodreads.com API. I was not allowed to make more than one request per second, and reviews were limited to 300 characters (which was the biggest disappointment). There is a method to select the top reviews from each book given an ISBN; however, sometimes the reviews were not in English so again I manually selected reviews from their website. The exact review ID's are given in my code. A further limitation is that I am not allowed to store any data obtained from their API. They state that we may "[n]ot use the API to harvest or index Goodreads data without our explicit written consent"; consequently, the "tidy" data included with this project will only include the Gutenberg data. Storing data from their website in a CSV and committing it to github would be a direct violation of their Terms of Use.

Exploratory Analysis

I will be saving the analysis for the Goodreads.com data for the results section, since I would prefer not to perform any individual analysis and reveal any of Goodreads's data. Instead in this section I will show some R data analysis techniques for text we learned in class. I will focus solely on one book because representing all ten in one plot is a bit overwhelming. The book I have chosen to show data for is Dracula. For instance, below are the top ten most common non stop-words in Dracula by Bram Stoker.

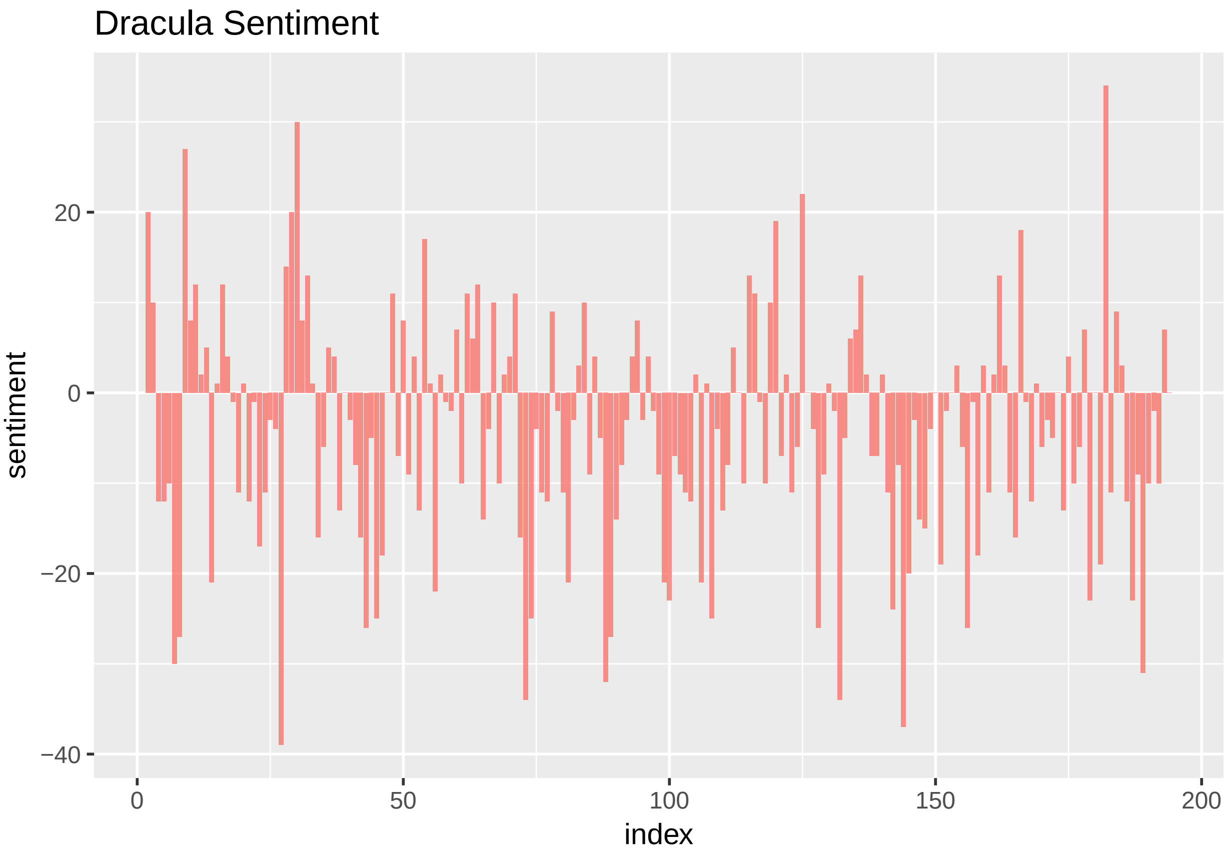
gutenberg_id	word	n
345	time	390
345	van	323
345	night	310
345	helsing	301
345	dear	223
345	lucy	223
345	day	220
345	hand	210
345	mina	210
345	door	200

As we can see many of these words are hardly surprising. There are four names in the top ten, which is hardly surprising. Other words don't give us much context, except for maybe "night" and "time" (since this is a horror novel). Let's see what happens when we get the bigrams.

bigram	n
dr seward	55
dr seward's	30
dr van	60
friend john	54
harker's journal	40
lord godalming	63
madam mina	82
poor dear	34
seward's diary	49
van helsing	282

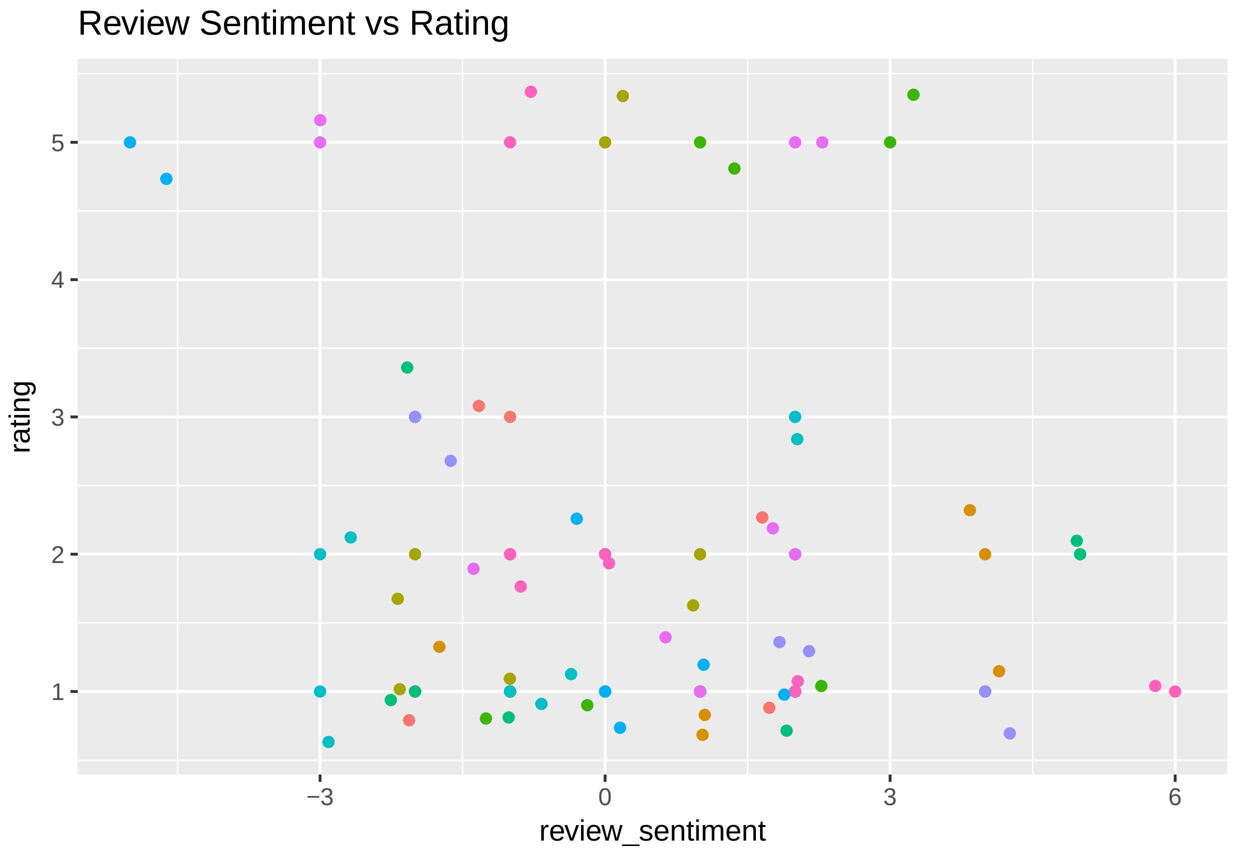
That actually gets a bit worse.

Below I plot the sentiment (by line) for Dracula. Clearly this book has negative sentiment since it is a horror book, although I would be pretty excited to read a horror novel in which the author uses overwhelmingly positive words to describe horrific situations.



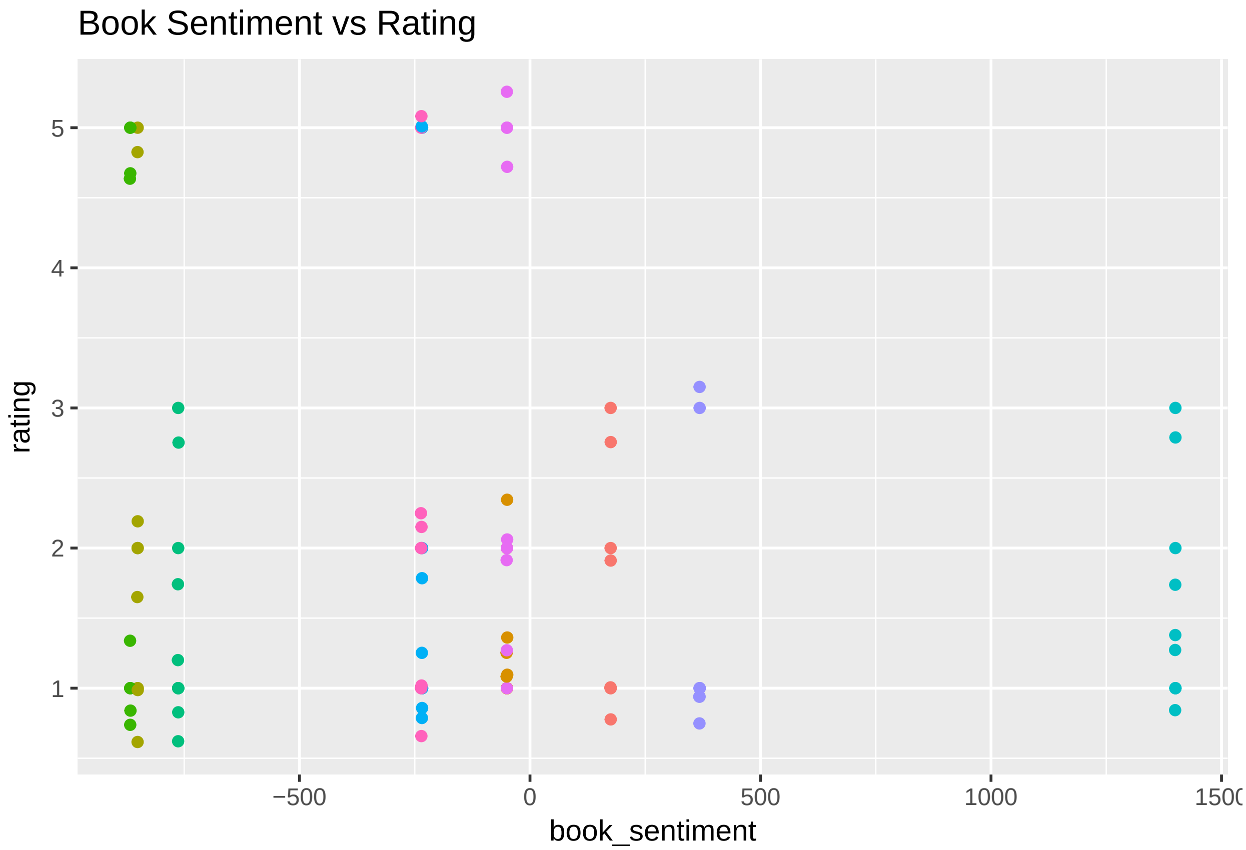
Results

Below I show the review sentiment versus the rating the reviewer gave for that book.

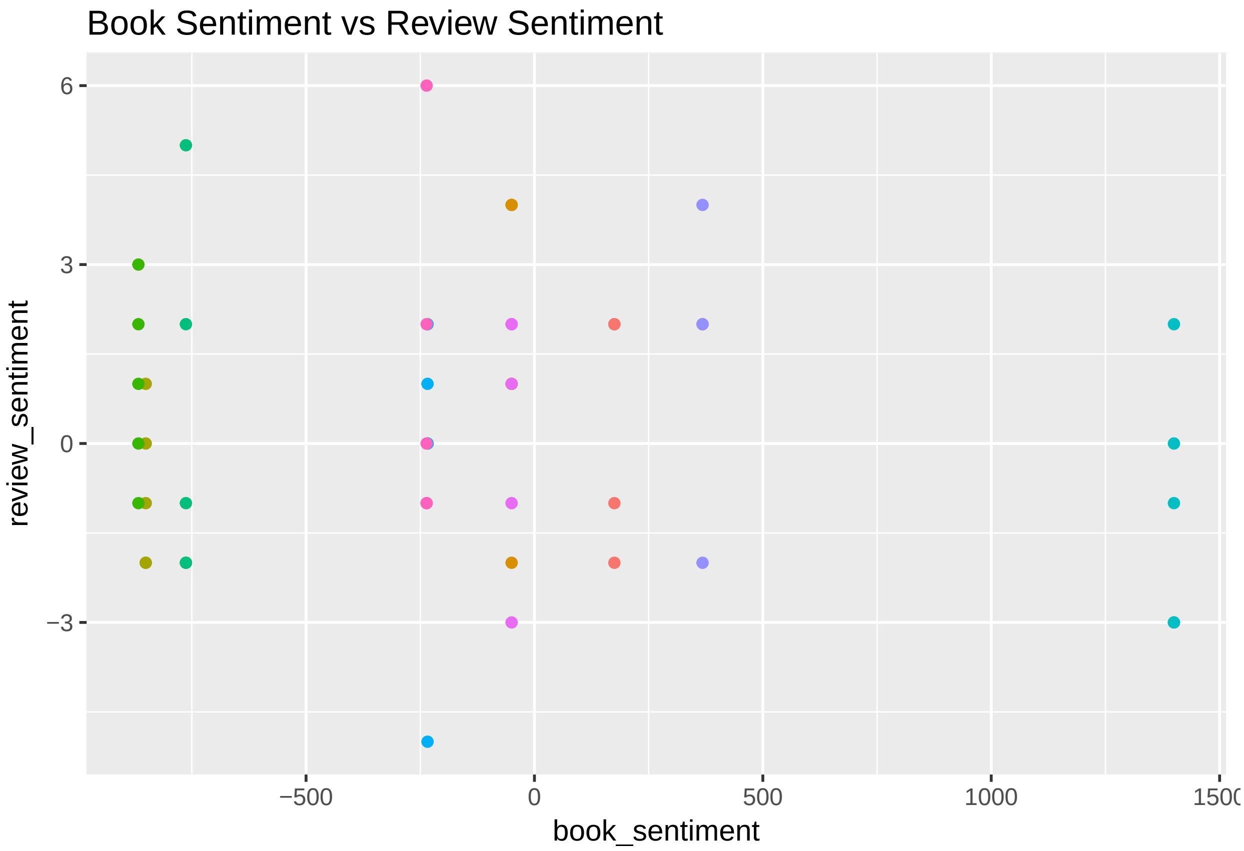


First, let us note that we had no 4 star reviews in our sample. The sample is skewed anyway, since the highest upvoted reviews tend to be extreme and because people who support the extreme views tend to upvote. It is a noticeable phenomenon that most online reviews tend to be extreme.

Otherwise, we don't really see any discernable pattern. It seems 5 star reviews tend to have lower sentiment, but negative reviews have slightly more positive sentiment. Perhaps the 5 star reviews tend to be more even-handed in their reviews, but I cannot explain why the sentiment should be more positive for the lower star reviews. Perhaps they were writing about how happy they were done to be reading.



Above is a plot of the book sentiment versus the review sentiment. Let's note that the more positive the book sentiment, the more negative the review tends to be. All the reviews in the light blue towards the right of the graph are for Pride and Prejudice. It seems people don't really like positive books on this website. Otherwise, an interesting phenomenon is that the books with more neutral sentiment tend to have a wider spread in rating. Had I more data and some knowledge of nonlinear modeling I would take a deeper look into this.



Above we view, in the book sentiment versus rating sentiment plot, the same pattern we noticed in the previous plot. Books with more extreme sentiments tend to have a smaller spread of review sentiment.

Conclusion

Even if we had access to more than 300 characters of each review, truly understanding the sentiment would be tricky. We would probably not be any better off than we are now using the simple tools for text analysis at hand if we tried to do some sort of correlation at scale. This is where NLP would come into play, and had I more experience in that domain I would have tried to create something that would try to correlate a review's sentiment with a book's sentiment. However, this project was a good exercise for me in working with APIs and doing some text analytics, two spaces I definitely have been lagging in compared to the rest of the data science community. Plus, it was fun to read some people's reviews on Goodreads.