Laith Barakat, Puneet Bhatia, Leila Ouellaj

M Numbers: M08790430, M13433518

M13408603

Sentiment Analysis on Restaurant Reviews in R

IS 8070 Group Project

Table of Contents

[Introduction 2](#_Toc38918390)

[Overview of Category of Chosen Software 2](#_Toc38918391)

[Similar Available Tools 3](#_Toc38918392)

[Description of R Packages Used 5](#_Toc38918393)

[Demonstration of Analysis and Software Implementation 6](#_Toc38918394)

[Business case for Sentiment Analysis 6](#_Toc38918395)

[Analysis Results 7](#_Toc38918396)

[Future Trends in Sentiment Analysis 10](#_Toc38918397)

[Conclusion 11](#_Toc38918398)

# Introduction

This project is an in-depth, introductory project that employs a Machine Learning technique known as sentiment analysis. Sentiment analysis is the inference of text data into actionable results based on the language and connotation of included text. This method is often used to analyze social media entries. The analysis conducted here weighs different options for software tools that can conduct sentiment analysis before settling on the statistical program R as the intended tool for use. With R, the project team conducts a sentiment analysis of tweets selected for highlighting numerous fast-food chains. This sentiment analysis is an example of the business applications of text analytics and its potential power in the marketplace, which is discussed throughout this paper.

# Overview of Category of Chosen Software

Sentiment analysis refers to the use of natural language processing, text analysis, computational linguistics, and biometrics to systematically identify, extract, quantify, and study affective states and subjective information. Natural language processing is now a major part of machine learning and data science. While many software packages are available, this project will be carried out using R programming language and RStudio.

R is a language used for statistical computations, data analysis and graphical representation of data. It was created in the 1990s and was designed to be solely a statistical platform for data cleaning and analysis. [[1]](#footnote-1)Since then, R has gained immense popularity in the world of data analytics and data science. According to 2107 Burtch Works Survey, out of all surveyed data scientist, 40% prefer R, 34% prefer SAS and 26% Python.[[2]](#footnote-2) According to KDNuggets’ 18th annual poll of data science software usage, R is the second most popular language in data science.

R is popular in the field of data science as many researchers and scholars use it for experimenting with data science which means that there a lot of hands on resources to learn R from. This makes the learning curve for R much faster than other languages.

The popularity of R also relies on the fact that is has extensive libraries for data manipulation and wrangling, a step that’s very important and time consuming in data science. Some of the most popular packages are dplyr, data.table and readr. Moreover, R provides multiple packages for data visualizing that enables analyzing data from angles that aren’t necessarily clear in unstructured or tabular data. Many R packages have become the standard popular packages for visualizing data such as ggplot2.

R is famous for its specificity in the sense that it was designed especially for statistical analysis and data reconfiguration. Almost all the libraries are designed for the purpose of making data analysis better and easier.

With all the reasons mentioned above making R a suitable language, the main reason this project is carried out using this language is that R is an open source software that made it adapt to developments happening at a rapid scale in the world of data science. R is the perfect choice to begin immersing in data science due to the packages constantly being added and adapting to the new trends in data science. For example, Caret package includes a set of functions that streamline the process of creating predictive models. It contains tools for data splitting, pre-processing, feature selection, model tuning via resampling, and variable importance estimation. Mlr is a framework that provides the infrastructure for methods such as classification, regression, and survival analysis, as well as unsupervised methods such as clustering.[[3]](#footnote-3) For this project, we will be using two sentiment analysis packages: *sentimentr* and *syuzhet.*

# Similar Available Tools

A sentiment analysis tool is software that analyzes text conversations and evaluates the tone, intent, and emotion behind each message. By digging deeper into these elements, the tool uncovers more context from text conversations and helps the customer service team accurately analyze feedback.

There are two categories of tools available for sentiment analysis: open source and SaaS.

Open-source APIs are completely free and publicly accessible to all developers who want to use them. Open-source APIs offer flexibility and customization, giving developers a lot of room to play with.

Because open-source APIs require a lot of coding, the user needs to be fluent in at least one programming language and familiar with machine learning concepts.

SaaS APIs for sentiment analysis are better option for someone not well-versed in machine learning, don’t want to spend too much time on building infrastructure, or invest in extra resources. With SaaS APIs, in-house dev teams or an on-premise infrastructure to run machine learning models is not required.

**Open Source APIs for Sentiment Analysis**

**Python**

Python is a favorite with developers interested in machine learning. It has a large number of libraries that are handy for implementing a sentiment analysis model from scratch.

NLTK, or the Natural Language Toolkit, is one of the leading libraries for building Natural Language Processing (NLP) models, thus making it a top solution for sentiment analysis. It provides useful tools and algorithms such as tokenizing, part-of-speech tagging, stemming, and named entity recognition.

SpaCy is an industrial-strength NLP library in Python which can be used for building a model for sentiment analysis. It provides interesting functionalities such as named entity recognition, part-of-speech tagging, dependency parsing, and word vectors, along with key features such as deep learning integration and convolutional neural network models for several languages.

Scikit-learn is a machine learning toolkit for Python that is excellent for data analysis. It features classification, regression, and clustering algorithms.

TensorFlow is the dominant framework for machine learning in the industry. It has a comprehensive ecosystem of tools, libraries, and community resources that lets developers implement state-of-the-art machine learning models.

PyTorch is another popular machine learning framework that is mostly used for computer vision and natural language processing applications. Developers love PyTorch because of its simplicity; it’s very pythonic and integrates really easily with the rest of the Python ecosystem. PyTorch also offers a great API, which is easier to use and better designed than TensorFlow’s API.

Keras is a neural network library written in Python that is used to build and train deep learning models. It is used for prototyping, advanced research, and production.

**Java**

Java is another programming language widely used for machine learning and provides some great options for implementing sentiment analysis.

CoreNLP is Stanford’s proprietary NLP toolkit written in Java with APIs for all major programming languages. It is powerful enough to extract the base of words, recognize parts of speech, normalize numeric quantities, mark up the structure of sentences, indicate noun phrases and sentiment, extract quotes, and much more.

OpenNLP is an Apache toolkit designed to process natural language text with machine learning. It supports language detection, tokenization, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing, and conference resolution.

Weka is comprised of a set of machine learning algorithms for data mining tasks. It includes tools for data preparation, classification, regression, clustering, association rules mining, and visualization.

**SaaS APIs for Sentiment Analysis**

Key advantages of SaaS APIs for sentiment analysis:

No coding: Since SaaS solutions are a ready-to-use solution, a whole bunch of code is not required to start using sentiment analysis.

No machine learning knowledge needed: One of the main benefits of using a SaaS tool is that the user does not need to worry about learning the ins and outs of NLP or machine learning, they are built to use sentiment analysis right away.

No setup: Getting started from scratch to implement a sentiment analysis solution is certainly challenging. Typically, open-source libraries require hours of coding and testing before they can be deployed, but with SaaS APIs, one can forget about spending time building the necessary infrastructure. Plus, there is no need to worry about maintenance. One can leave that to the vendor responsible for managing the tool, eliminating unnecessary work for the team.

Some of the most popular SaaS solutions sentiment analysis include:

* MonkeyLearn
* Google Cloud NLP
* IBM Watson
* Lexalytics
* MeaningCloud
* Amazon Comprehend
* Aylien

# Description of R Packages Used

Despite the multitude of other tools described above, our team utilized R and RStudio due to a number of packages and advantages that would be key to our project. First and foremost, R is the primary language that the MS-BANA program at UC uses for many of its classes, and our team is most familiar with R as a whole for general analysis and programming. Committing to R means that the whole team can understand the process of the project and interpret results accordingly. In addition, the two packages that are used for this analysis are great reasons to begin a sentiment analysis in R: *sentimentr* and *syuzhet.*

The *sentimentr* package in R is intended to calculate text polarity sentiment and is maintained by Tyler Rinker. Published in early 2019, the package is an introductory package to those generally unfamiliar with sentiment analysis and its associated processes[[4]](#footnote-4). Within the package, users can calculate the text polarity sentiment at a sentence level and optionally aggregate by rows or grouping variables. This package is a concise and easy way to calculate sentiment within text data. This package has the function *sentiment*, which is a key function in our analysis. The *sentiment* function passes a text variable through a polarity data table of positive and negative words and weights. This polarity data table can be one of several dictionaries that are previously developed for this analysis, including any dictionary within the expansive *lexicon* package that is a sentiment dictionary. A vital advantage of this package and this function is that the entirety of underlying sentiment analysis legwork is already completed; that is, our team can easily import this package and conduct sentiment analysis without a huge upfront investment of time into the background analysis. For a team which is conducting its first sentiment analysis, this is a distinct plus. Yet, a drawback of using both *sentimentr* and *syuzhet* is that these packages provide a ceiling to the level of sophistication within the analysis. For an introductory project, this is not a consideration, but if our team wanted a more fine-tuned, deep sentiment analysis, a different tool could be preferable.

The other package used in the analysis, *syuzhet,* is quite similar in practice to the *sentimentr package.* It was developed in 2017[[5]](#footnote-5), a few years prior to *sentiment*r, and is maintained by Matthew Jockers. The purpose of the package is to extract sentiment from text using a variety of sentiment dictionaries conveniently packaged for R users. This package is used identically within our analysis; the main purpose for using both packages was to vary the sentiment dictionaries and the approach to the analysis. *Syuzhet*’s primary function for extracting sentiment is the *get\_nrc\_sentiment*, a package that calls the NRC sentiment dictionary by default to determine the presence of eight different emotions and the relative valence.

A key advantage of both of these packages, and of R as a whole, within this data analysis is, of course, the ease of use for our team; however, it is also quite important that these tools are out-of-the-box, and, most importantly, open source. Open source software has distinct advantages in their own right, including a lower project cost, a higher level of flexibility, and additional security[[6]](#footnote-6). For the purposes of our project, this gives our team an understanding of sentiment analysis that we can easily adapt to our jobs in the workforce without incurring large costs to our employer to begin.

# Demonstration of Analysis and Software Implementation

The first part of the project is data gathering. We’ve decided to use tweets as our main input to the sentiment analysis model. In order to do so, the first part of the code extracts tweets from a free Twitter API provided for academia. The API provides the user with an access token that gives the user access to searching for tweets and information about them and stores them in a data frame.

The second part is tokenization which is especially important in the context of tweets as they contain many uninformative words. We’ve created a function that removes “stop” words such as “and”, “the”, “of”, “or” as well as twitter specific words such as “rt” etc. The function also splits the tweets into individual words and removes all #hashtags and @signs.

After tokenizing and cleaning the data, we feed the tweets to specific functions of each package. For the first packages sentimentr, the function takes the tweets and returns a score for each tweet. This score is then classified according to a function our team coded into positive, negative and neutral. For the second packages syuzhet, the input is similar. However, the function returns a range of emotions such as anger, joy, sadness additional to the positive, negative and neutral scores.

Finally, a data table is created from the positive and negative classification and a graph is created to visualize the results.

# Business case for Sentiment Analysis

Our project, as outlined in technical detail in other sections of this paper, analyzes the sentiments of 1,000 tweets for five popular fast-food franchises: Burger King, Chick-fil-A, KFC, McDonald’s, and Subway. These tweets are analyzed independently from each other and rely on user-generated content (UGC). In today’s world, UGC is tremendously available and easy to access. There are volumes of UGC from social media that allow enterprises to reach the Voice of the Customer (VoC) more easily than ever before. Prior to the advent of Big Data Analytics, the VoC was reached through surveys, focus groups, and interviews. While these methods often brought management of enterprises face-to-face with their customers, they also brought uncertainty and the possibility of sampling error and high levels of bias. Not to mention, surveys and focus groups on a large scale quickly become expensive for a business to continually run or to repeatedly run, including compensation and management of interviewees.

To combat the difficulties of traditional methods in reaching the VoC, text analytics and sentiment analysis has become a norm in the business world. Generated from UGC, conducted with a much smaller investment from a company, and inclined to a higher degree of accuracy, sentiment analysis can infer product knowledge, visualize competition between brands, and help customers make better, quicker, and more profitable decisions when buying. Not only that, but sentiment analysis can be much faster to deliver insights and results than traditional methods. Business leaders have loved sentiment analysis, across many industries: it has been reported that the Obama administration employed sentiment analysis leading up to the 2012 election to measure how it has engaged with key voters[[7]](#footnote-7). Of course, sentiment analysis is not a perfect solution for reaching the VoC. It is often difficult to train a program to recognize sarcasm, to provide context to a comment, and to infer an overall tone. Yet, the article cited above addresses that advanced systems have been largely innovating to defeat these threats to the legitimacy of text analytics so that the advantages discussed above quite offset the difficulties within sentiment mining.

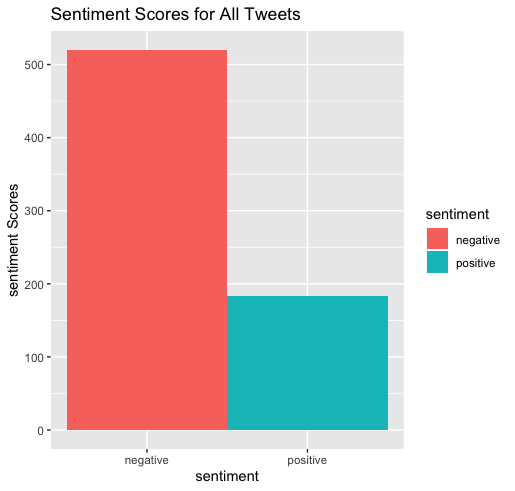
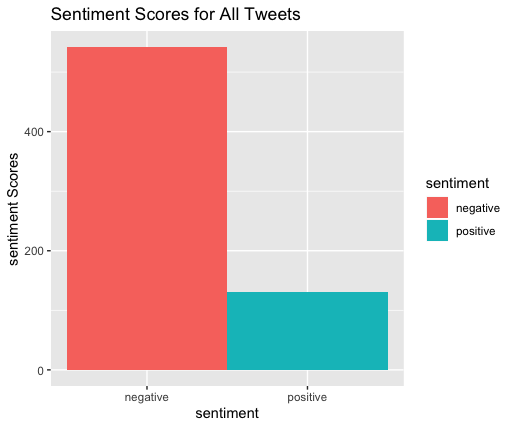
Our project specifically hopes to provide an overall consumer sentiment to the aforementioned five restaurant chains. A business leader at the helm of any of these chains would love to see information about customer “buzz” for each of his/her competitors in comparison to the business. While this project outlines a very introductory analysis to overall sentiment, this launch-point of information could trigger questions that the business leader did not think to ask prior to the results of the analysis.

# Analysis Results

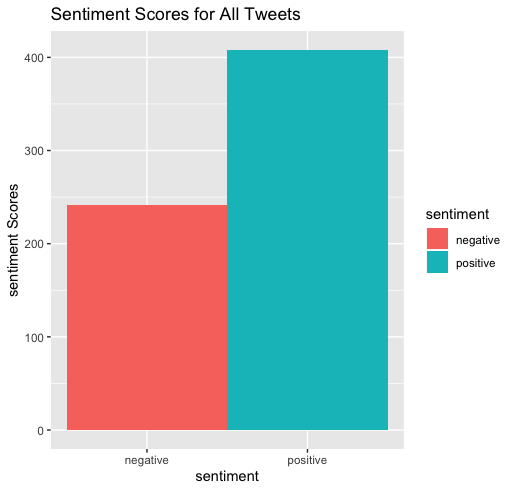
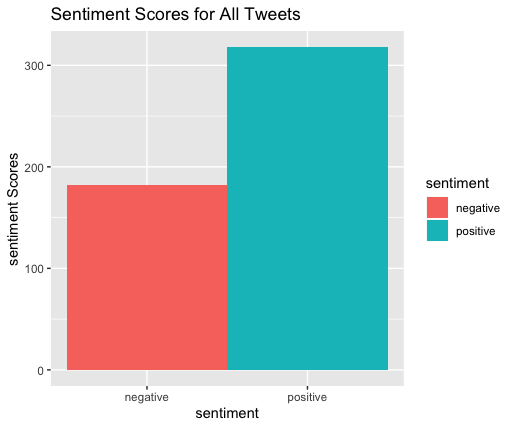
Our analysis has rendered results that a business leader would find intriguing. In terms of results, each sample set of 1,000 tweets was passed through *sentimentr* and *syuzhet* functions to yield a sentiment score for each tweet. Any tweet not classified as either positive or negative was neutral and not included. The results for each chain are visualized below. The first barplot is the *syuzhet’*s results, and the second is the *sentimentr* results.

**Positive and Negative Sentiments (per 1,000 tweets) for:**

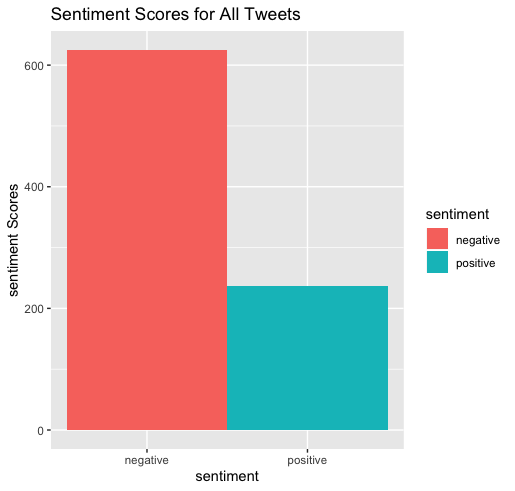
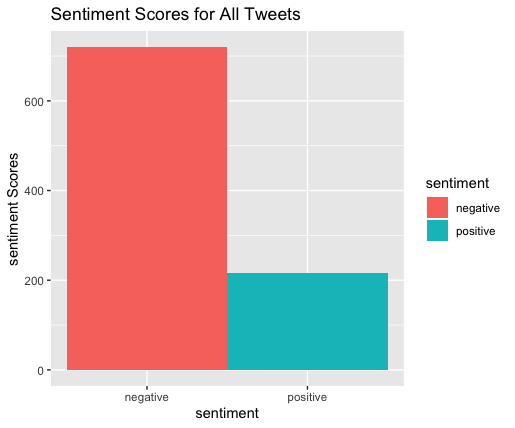
**Burger King:**



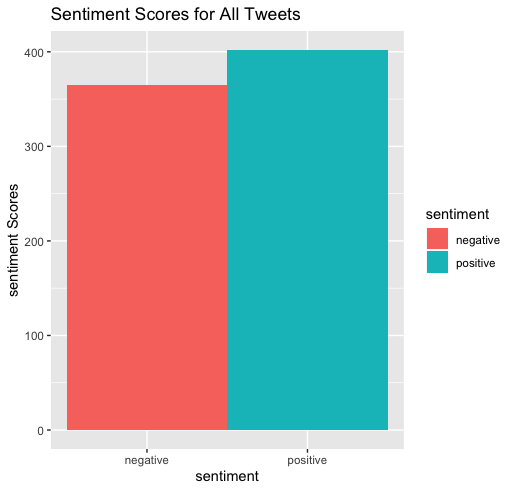
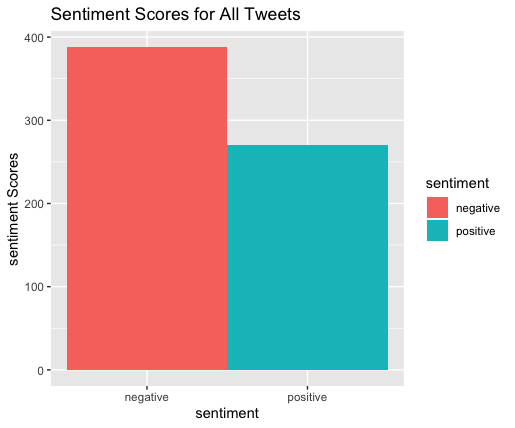
**Chick-fil-A:**



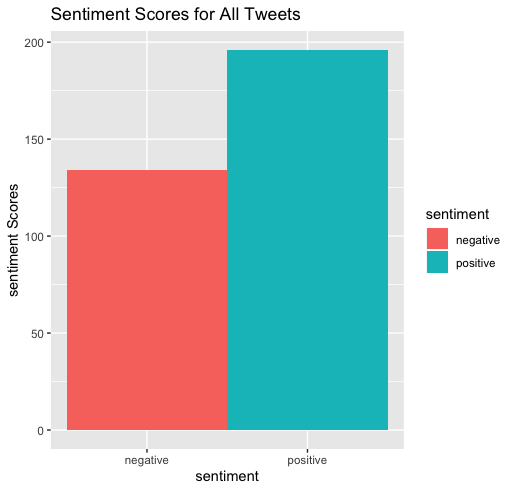
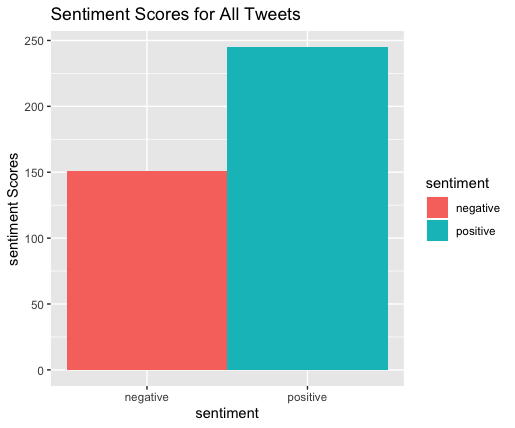
**KFC:**



**McDonald’s:**



**Subway:**



As discussed above, business leaders are always searching for the best ways to reach the voice of the customer. This analysis demonstrates that, once they get the voice of the customer, they might not like the results. Chick-Fil-A and Subway had a majority of tweets in the positive sentiment score, while McDonald’s was closely split by dictionary and functions used, and Burger King and KFC were both resoundingly negative in result. This can be interpreted to yield a clear set of winners and losers in the fast-food space among these competitors. Ultimately, the analysis gives detail to the structure of the market based on consumer thinking.

This analysis conducted is a relatively imprecise method to sentiment analysis. There are few drawbacks to this analysis- however, sample size and sentiment precision would be opportunities of improvement. Sample size for this analysis was limited to 1,000 tweets due to computing power limitations of our team; an enterprise could easily circumnavigate this issue with a minimal investment into a larger computer with expanded power. Sentiment precision is a trickier opportunity: this opportunity opens up the subjects of attribute extraction and hierarchy identification, topics that were not covered in the scope of this work. For a typical professional text analytics method, these processes are key for identifying specific details and attributes of each analytical subject, so that detailed analysis can be conducted on aspects of a product or company, as opposed to a general overall sentiment. Since our team focused on general, overall sentiment for this project, attribute and hierarchy identification was not included.

# Future Trends in Sentiment Analysis

The use cases for sentiment analysis are many. As previously discussed, market research and structuring, customer feedback, and social media monitoring[[8]](#footnote-8) are all consistently pursued outcomes for sentiment analysis. Yet, as the cutting edge of sentiment analysis and machine learning in general is always sought-after, the question becomes: what is next?

There will undoubtedly be innovations in both the technical aspects of sentiment analysis and in the applications for use. Within the technical realm, some drawbacks of sentiment analysis were previously discussed, including the difficulty in identifying sarcasm and more nuanced tone within text. One future trend in this field could be the improving condition of programs to detect these more complex aspects of human language. A neural network could be employed to identify sarcasm in text and, when used in conjunction with improved and more numerous dictionaries within the sentiment analysis field, could create even more in-depth levels of sentiment precision.

On the application side, it would be encouraging for politics to embrace sentiment analysis in a benevolent way. Of course, a previous application that was discussed was within re-election campaigns; but it could be a great turning point in our history if politicians kept their “finger on the pulse” with advanced text mining/sentiment analysis techniques in regards to proposed legislation and authority measures. One of the historical difficulties of democracy is taking everyone’s voice into account. Advancements in sentiment analysis could eradicate this difficulty and make a more efficient democracy (that caters to the true voice of the people) a reality.

# Conclusion

This project ultimately encapsulates the importance of sentiment analysis in a modern-day organization. As discussed, this machine learning technique can highlight the consumer’s views of a certain market, crowdsource feedback about a product or service, and much more. With the plethora of software tools available for sentiment analysis, it is best to conduct such a product with an easy to use and open source tool when doing so for the first time. The project team’s analysis in R highlights how sentiment can be sourced from tweets about fast-food restaurants and show whether customers largely like or dislike each player in the market. In addition, this paper discusses why a business leader may opt to pursue sentiment analysis and where the future of the field may lie. In conclusion, sentiment analysis is a powerful tool and could unlock huge potential for any organization when utilized properly.

1. <https://www.newgenapps.com/blog/6-reasons-why-choose-r-programming-for-data-science-projects/> [↑](#footnote-ref-1)
2. <https://www.newgenapps.com/blog/6-reasons-why-choose-r-programming-for-data-science-projects/> [↑](#footnote-ref-2)
3. https://monkeylearn.com/blog/sentiment-analysis-apis/ [↑](#footnote-ref-3)
4. <https://cran.r-project.org/web/packages/sentimentr/sentimentr.pdf> [↑](#footnote-ref-4)
5. <https://cran.r-project.org/web/packages/syuzhet/syuzhet.pdf> [↑](#footnote-ref-5)
6. <https://www.zivtech.com/blog/benefits-open-source-software> [↑](#footnote-ref-6)
7. <https://www.forbes.com/sites/jiawertz/2018/11/30/why-sentiment-analysis-could-be-your-best-kept-marketing-secret/#5754f0d92bbe> [↑](#footnote-ref-7)
8. <https://monkeylearn.com/sentiment-analysis/> [↑](#footnote-ref-8)