Journal of Statistical Software

February 2018, Volume 83, Book Review 1. *doi: 10.18637/jss.v083.b01*

Reviewer: Colin Fay
ThinkR

Text Mining with R: A Tidy Approach

Julia Silge, David Robinson
O'Reilly Media, Sebastopol, 2017.
ISBN 1491981652. 194 pp. USD 39.99 (P).
http://tidytextmining.com/

Text mining (and more broadly natural language processing) has been an active field in science for decades. Yet, its use in business has become more and more important as the web became what it is now: an almost infinite source of (textual) data. And as the amount of available data is larger and larger every day, there is no doubt we need tools and methods to explore, organize, and analyze these data.

R users can choose various toolkits to perform these tasks of exploration and analysis. One paradigm for Data Science with R is the tidyverse, a suite of packages relying on a common philosophy, and which have been developed to work hand in hand. *Text Mining with R* focuses on how to do text mining within this framework, notably by using the **tidytext** package.

Written by Julia Silge and David Robinson (also author of **tidytext**), this book "serves as an introduction to text mining using the **tidytext** package and other tidy tools". In other words, this book is focused on explaining how the package works, and how to combine it with other packages like **dplyr** or **ggplot2**.

Chapter 1 focuses on the *tidy text format* and its use for word frequency calculation. The *tidy text format* refers to a *one token per row* format, a token being a word, a n-gram, a sentence, a paragraph, or any unit that can be useful for text analysis. Based on the tidy data principles, this format contrasts with other formats like strings, corpus or document term matrix, in that sense that it is based on keeping everything in one data frame, with one observation by row and one variable by column. This format is obtained by calling the `unnest_tokens` function from the **tidytext** package. In this first chapter, the authors exemplify how to perform word frequency analysis, starting from data collection with **gutenbergr**, moving to manipulation (turning data into a tidy text format, counting occurrences, removing stop words, filtering...) and ending with visualisation.

Chapter 2 of *Text Mining with R* is devoted to "Sentiment analysis with tidy data". In this chapter, the authors introduce the three datasets available in the package: the AFINN, Bing and NRC lexicons. Based on unigrams, these datasets have been transformed to a tabular format, in order to be usable with the packages and functions presented in the book. In this chapter, the lexicons are used to conduct a sentiment analysis on Jane Austen's novels: either by sentiment frequency (fear, joy, . . .) or by polarity (positive or negative words), with the help of a joining function.

Chapter 3 is dedicated to computing tf-idf, which is an acronym for term frequence – inverse document frequency, a statistic which measures the importance of a word to a document in a corpus. Still with the Jane Austen's novels as an example, the authors introduce the reader to the `bind_tf_idf` function, used to calculate this statistic. They also briefly present the zipf law (a law stating that a word frequency is inversely proportional to its rank), and show how to manually compute this law. The chapter ends with a second use case of the tf-idf measure, with physics texts taken from the Gutenberg project.

In Chapter 4, Julia Silge and David Robinson "explore some of the methods tidytext offers for calculating and visualizing relationships between words". As the book has focused on monograms in the first three chapters, this fourth chapter is dedicated to units of text which contain more than one word. A token format which is obtained by passing `token = "ngrams"` to the `unnest_tokens` function.

In the section on bigrams (n-grams with two words), the authors show how to manipulate these tokens, in order to filter stop words, or to add more context to sentiment analysis by collecting the bigrams with a "not" as the first element. Once these bigrams are separated into two columns, the authors describe how to turn this format into a graph using the **ggraph** package. Finally, the chapter focuses on how to compute correlation between words with the help of **widyr**.

Chapter 5 is dedicated to "nontidy formats". In other words, this chapter gives common recipes to interact with other text mining packages like **quanteda**, **tm**, or **topicmodels**. This interaction is made possible by several functions presented here: `tidy`, `cast`, `cast_sparce`, `cast_dtm` and `cast_dfm`.

This series of functions are used in Chapter 6, where the authors introduce the concept of topic-modeling. Here, the reader will find a methodology to perform this analysis with **tidytext** and **topicmodels**, along with a conceptual presentation of this machine learning technic.

The authors end up the book with three case studies: one comparing between two Twitter feeds, one mining metadata from the NASA datasets, and one with an analysis of 20,000 messages from online newsgroups.

Starting with simple examples and concepts, and moving gradually to more complex topics, *Text Mining with R* gives the reader all the keys to gain insights from their textual data, with a large emphasis on practice. As you read this book, you will be gently introduced to the concepts of text mining: word-frequencies, sentiment analysis, tf-idf, topic modeling, etc., with just enough theory to understand what these concepts are.

Even if you are totally new to text mining, this book is a the perfect starting point if you want to analyze textual data. And if you are already familiar with the key concepts, you will find in this book common recipes for text mining, and will be able to apply them to your specific case right away.

Reviewer:

Colin Fay
ThinkR
E-mail: colin@thinkr.fr
URL: https://thinkr.fr/

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
February 2018, Volume 83, Book Review 1 *Published:* 2018-02-22
doi:10.18637/jss.v083.b01