Pride and Prejudice Wordcloud

Dayana Moncada

October 30, 2017

Abstract

In this article we construct a wordcloud, using the tidytext R package, for Jane Austen's Pride and Prejudice novel. This is a replica from my professor Dr. Charles Redmond's work with the Sense and Sensibility's wordcloud using the same packages. I will try to tweak and personalize some codes. Finally, this is a learning experience and comments/questions are deeply appreciated.

Pride and Prejudice Pride and Prejudice is one of Austen's famous book where she tells the love story between Elizabeth Bennet and Mr. Darcy. For a lot of time, this novel has been considered a classic "must=read". Jane Austen shows us "the folly of judging by first impressions and superbly evokes the friendships, gossip and sobberies of provincial middle-class life."[1]

1 The Jane Austen Package

This package contains the complete text of Jane Austens 6 completed, published novels, formatted to be convenient for text analysis.

```
library(janeaustenr)
pnp<-austen_books()
```

This dataframe has two columns, one for each line in Austen's novels, and one indicating which book the line is from. Let's first filter, using dplyr, so that we have only the lines from Pride and Prejudice:

```
library(dplyr)
pnp<-pnp%>%
  filter(book == 'Pride & Prejudice')

head(prideprejudice)
```

```
## [1] "PRIDE AND PREJUDICE" ""                    "By Jane Austen"
## [4] ""                    ""                    ""
```

Now we are ready for some data cleaning.

[1] Amazon.com

2 Some Data Cleaning

We would like to remove all of the 'Chapter' lines. We can use dplyr again, along with the package stringr.

```
library(stringr)
pnp<-pnp%>%
  filter(!str_detect(pnp$text,'^CHAPTER'))
```

Next, we would like to remove the front matter. By inspection, we have determined that the front matter ends on line 9. Therefore we can redefine sns to begin on line 10:

```
pnp<-pnp[10:12030,]
```

3 The Wordcloud

To make the wordcloud, we first have to break up the lines into words. We can use a function from the tidytext package for this:

```
library(tidytext)

## Warning:  package 'tidytext' was built under R version 3.4.2

words_df<-pnp%>%
  unnest_tokens(word,text)

words_df

## # A tibble: 112,909 x 2
##                  book         word
##                <fctr>        <chr>
##  1 Pride & Prejudice           it
##  2 Pride & Prejudice           is
##  3 Pride & Prejudice            a
##  4 Pride & Prejudice        truth
##  5 Pride & Prejudice  universally
##  6 Pride & Prejudice acknowledged
##  7 Pride & Prejudice         that
##  8 Pride & Prejudice            a
##  9 Pride & Prejudice       single
## 10 Pride & Prejudice          man
## # ... with 112,899 more rows
```

We can remove common, unimportant words with the stop_words data frame and some dplyr:

```
words_df<-words_df%>%
  filter(!(word %in% stop_words$word))

words_df
```

```
## # A tibble: 34,525 x 2
##                 book         word
##               <fctr>        <chr>
##  1 Pride & Prejudice        truth
##  2 Pride & Prejudice  universally
##  3 Pride & Prejudice acknowledged
##  4 Pride & Prejudice       single
##  5 Pride & Prejudice   possession
##  6 Pride & Prejudice      fortune
##  7 Pride & Prejudice         wife
##  8 Pride & Prejudice     feelings
##  9 Pride & Prejudice        views
## 10 Pride & Prejudice     entering
## # ... with 34,515 more rows
```

Now, we need to calculate the frequencies of the words in the novel. Again, we can use standard dplyr techniques for this:

```
word_freq<-words_df%>%
  group_by(word)%>%
  summarize(count=n())

word_freq
```

```
## # A tibble: 5,831 x 2
##            word count
##           <chr> <int>
##  1   _accident_     1
##  2 _advantages_     1
##  3     _affect_     1
##  4        _all_     4
##  5         _am_     1
##  6     _another     1
##  7        _any_     1
##  8  _anybody's_     1
##  9 _appearance_     3
## 10        _are_     2
## # ... with 5,821 more rows
```

Finally, it's time to generate the wordcloud:

```r
library(wordcloud)

## Loading required package:  RColorBrewer

library(tm)

## Loading required package:  NLP

wordcloud(word_freq$word,word_freq$count,min.freq=25)
```