Gutenbergr and Bar Plots

Kimberly Staudt

December 13, 2017

Outline

Install Libraries

First we must install the following libraries before we can begin.

- `library(dplyr)`
- `library(gutenbergr)`
- `library(tidytext)`
- `library(stringr)`
- `library(ggplot2)`

Using Gutenbergr

Download the Dunwhich Horror with the given ID and set to the dataframe, Lovecraft

```
Lovecraft<-gutenberg_download(50133)
```

Using Stringr

Once downloaded, use stringr to detect and remove all instances of the word Chapter.

```
Lovecraft<-Lovecraft%>%
filter(!str_detect(Horror$text,'CHAPTER'))

## Error in filter_impl(.data, quo):  Evaluation
error:  object 'Horror' not found.
```

Unnest Data

Next, Unnest the the text, and store into a dataframe.

```
words_df<-Lovecraft%>%
unnest_tokens(word,text)
colnames(words_df)

## [1] "gutenberg_id" "word"
```

Bing Lexicon

Use the bing lexicon to get the positive and negative sentiments in the text.

```
bing<-get_sentiments('bing')
colnames(bing)

## [1] "word"      "sentiment"
```

Inner Join

Use inner join to display positive and negative words in the text. Remove the gutenberg id tag.

```
 words_df<-inner_join(words_df,bing)

words_df$gutenberg_id<-NULL
```

Positive Words

Use dplyr to filter and count the top 10 most frequently occuring positive words. Then store this as a factor.

```
pos<-words_df%>%
  filter(sentiment=='positive')%>%
  group_by(word)%>%
  summarize(count=n(),sentiment=first(sentiment))%>%
  arrange(count)%>%
  top_n(10,wt=count)

pos$word<-factor(pos$word,levels=pos$word)
```

Negative Words I

Now, do the same thing for the 10 most frequently occuring negative words. Store this as a factor.Lastly, use rbind to combine the negative and positive words.

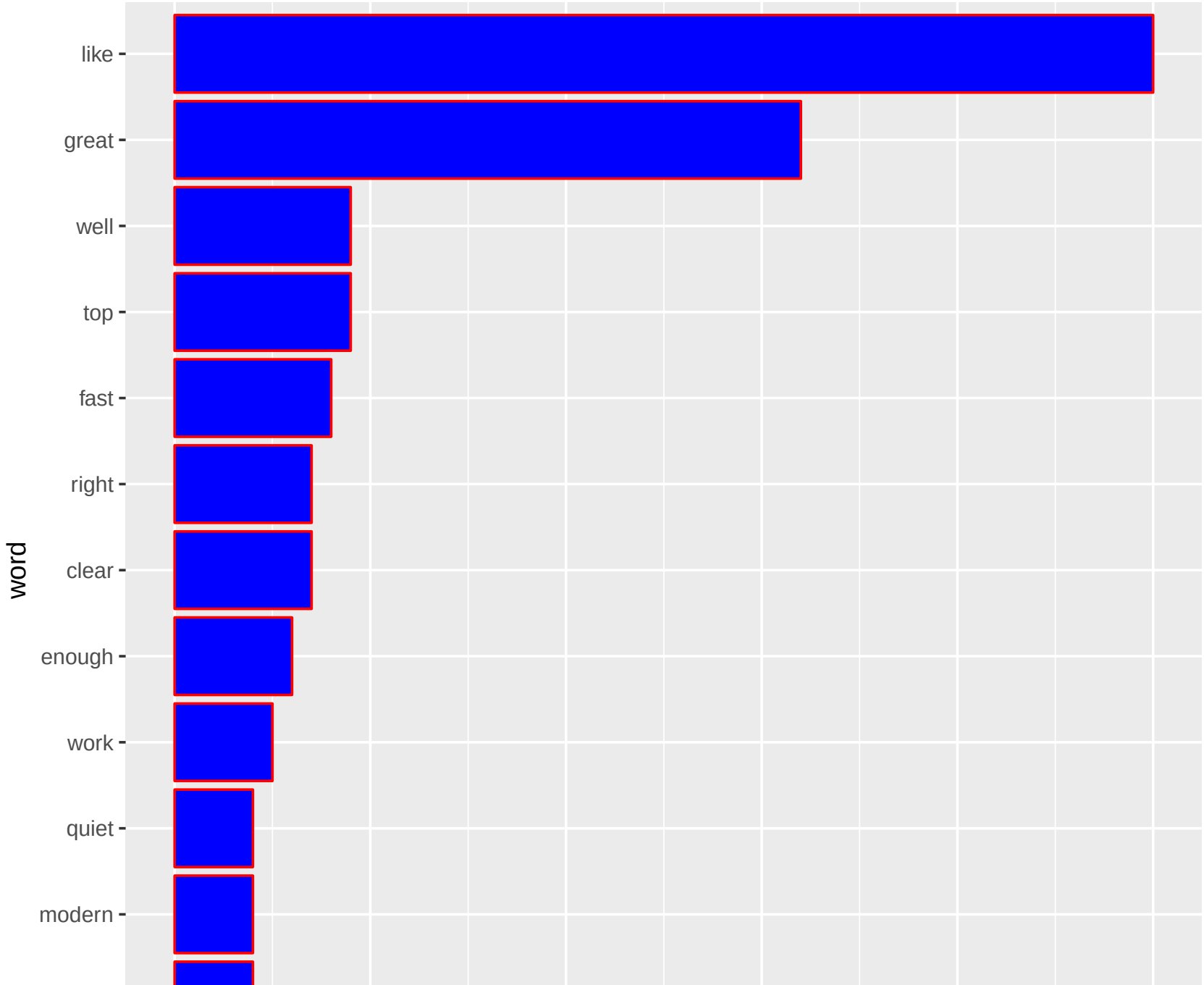
```
neg<-words_df%>%
  filter(sentiment=='negative')%>%
  group_by(word)%>%
  summarize(count=n(),sentiment=first(sentiment))%>%
  arrange(count)%>%
  top_n(10,wt=count)

neg$word<-factor(neg$word,levels=neg$word)

combo<-rbind(pos,neg)
```

Plotting I

Use ggplot to create a bar chart of the positive words.

```
ggplot()+
  geom_bar(data=pos,aes(x=word,y=count),color='red',fill=
coord_flip()
```

Plotting II


like
great
well
top
fast
right
clear
enough
work
quiet
modern
word

Comparing Words I

```
ggplot()+
  geom_bar(data=combo,
           aes(x=word,y=count, fill=sentiment,
               color=sentiment),stat='identity')+
  coord_flip()+
  facet_wrap(~sentiment,scales='free_y')+
  scale_fill_manual(values=c('red','yellow'))+
scale_color_manual(values=c('yellow','red'))
```

Comparing Words II