Compraison Barplots

Who? Justin Minsk

When? November 20, 2017

Load and Install Libraries

-
  ```
  library(dplyr)
  ```

Load and Install Libraries

-
  ```
  library(dplyr)
  ```

-
  ```
  library(tidytext)
  ```

Load and Install Libraries

-
  ```r
  library(dplyr)
  ```

-
  ```r
  library(tidytext)
  ```

-
  ```r
  library(gutenbergr)
  ```

Load and Install Libraries

-
  ```
  library(dplyr)
  ```

-
  ```
  library(tidytext)
  ```

-
  ```
  library(gutenbergr)
  ```

-
  ```
  library(ggplot2)
  ```

Load and Install Libraries

-
  ```
  library(dplyr)
  ```

-
  ```
  library(tidytext)
  ```

-
  ```
  library(gutenbergr)
  ```

-
  ```
  library(ggplot2)
  ```

-
  ```
  library(stringr)
  ```

Access Progect Gutenburg

```
df<-gutenberg_works(str_detect(title, 'Dracula
df$gutenberg_id

## [1]   345 10150

df$title

## [1] "Dracula"          "Dracula's Guest"
```

Download Dracula

```
dracula <- gutenberg_download(345)
colnames(dracula)

## [1] "gutenberg_id" "text"

substr(dracula$text[500],1,21)

## [1] "my own disappointment"
```

Unpacking the Words

```
dracula_words <- dracula%>%
  unnest_tokens(word, text)
colnames(dracula_words)

## [1] "gutenberg_id" "word"

dracula_words[498:500,]

## # A tibble: 3 x 2
##   gutenberg_id  word
##          <int> <chr>
## 1          345  fail
## 2          345    to
## 3          345  have
```

Bing Lexicon

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

## [1] "word"      "sentiment"

bing[498:500,]

## # A tibble: 3 x 2
##          word sentiment
##         <chr>     <chr>
## 1     bereave  negative
## 2 bereavement  negative
## 3      bereft  negative
```

Interjoin

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

## Joining, by = "word"

dracula_words$gutenberg_id <- NULL
colnames(dracula_words)

## [1] "word"      "sentiment"

dracula_words[498:500,]

## # A tibble: 3 x 2
##      word sentiment
##     <chr>     <chr>
## 1   great  positive
## 2    love  positive
## 3 crowded  negative
```

Top Ten Postive Words I

```
dracula_pos <- dracula_words%>%
  filter(sentiment == 'positive')%>%
  group_by(word)%>%
  summarise(count = n())%>%
  arrange(desc(count))%>%
  filter(count >= 66)%>%
  top_n(10, wt = count)
```

Top Ten Postive Words II

```
dracula_pos
```

```
## # A tibble: 10 x 2
##       word count
##      <chr> <int>
##  1    like   292
##  2    good   258
##  3    well   245
##  4   great   183
##  5    work   146
##  6   right    99
##  7    love    84
##  8  better    77
##  9   ready    71
## 10   sweet    66
```

Top Ten Negtive Words I

```r
dracula_neg <- dracula_words%>%
  filter(sentiment == 'negative')%>%
  group_by(word)%>%
  summarise(count = n())%>%
  arrange(desc(count))%>%
  filter(count >= 53)%>%
  top_n(10, wt = count)
```

Top Ten Negtive Words II

```
dracula_neg

## # A tibble: 10 x 2
##        word count
##       <chr> <int>
##  1     poor   193
##  2     fear   137
##  3     dead   109
##  4 terrible   100
##  5    death    94
##  6  strange    90
##  7     dark    77
##  8     miss    60
##  9     fell    59
## 10  trouble    53
```

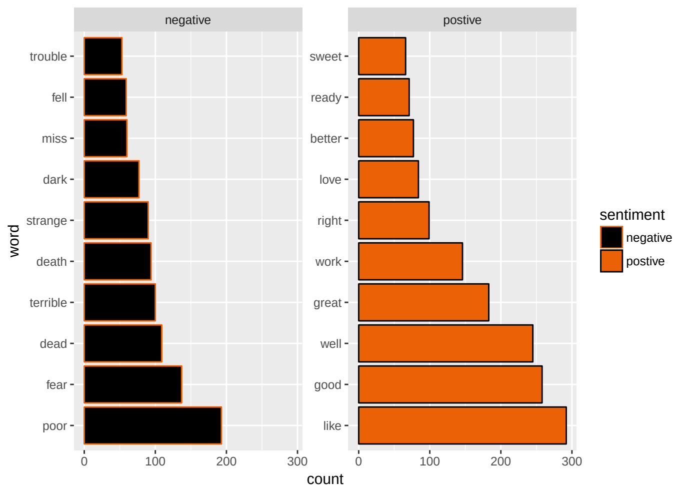
Comparison Bar Plot I

```r
dracula_pos$sentiment <- 'postive'
dracula_neg$sentiment <- 'negative'
dracula_pos$word <- factor(dracula_pos$word,
                           level = dracula_pos
dracula_neg$word <- factor(dracula_neg$word,
                           level = dracula_neg
dracula_comp <- rbind(dracula_neg, dracula_pos
```

Comparison Bar Plot II

```
ggplot()+
  geom_bar(data = dracula_comp,
           aes(x = word, y = count,
               color = sentiment,
               fill = sentiment), stat = 'iden
  coord_flip()+
  facet_wrap(~sentiment, scales = 'free_y')+
  scale_fill_manual(values = c('black', '#ea62
  scale_color_manual(values = c('#ea6205', 'bl
```

Comparison Bar Plot III


negative
postive
trouble
fell
miss
dark
strange
death
terrible
dead
fear
poor
sweet
ready
better
love
right
work
great
well
good
like
word
count
sentiment
negative
postive