The Masque of the Red Death by Edgar Allan Poe Wordcloud

Dayana Moncada

November 6, 2017

Abstract

In this article we construct a wordcloud, using the tidytext R package, for Edgar A. Poe's The Masque of the Red Death novel.

The Masque of the Red Death "The Masque of the Red Death", originally published as "The Mask of the Red Death: A Fantasy" (1842), is a short story by Edgar Allan Poe. The story follows Prince Prospero's attempts to avoid a dangerous plague, known as the Red Death, by hiding in his abbey. He, along with many other wealthy nobles, hosts a masquerade ball within seven rooms of the abbey, each decorated with a different color. In the midst of their revelry, a mysterious figure disguised as a Red Death victim enters and makes his way through each of the rooms. Prospero dies after confronting this stranger, whose "costume" proves to contain nothing tangible inside it; the guests also die in turn

1 The gutenbergr Package

Download and process public domain works in the Project Gutenberg collection http://www.gutenberg.org/. Includes metadata for all Project Gutenberg works, so that they can be searched and retrieved.

```
library(gutenbergr)
gutenberg_works(author == "Poe, Edgar Allan")
```

```
## # A tibble: 16 x 8
##    gutenberg_id
##           <int>
##  1          932
##  2         1062
##  3         1063
##  4         1064
##  5         1065
##  6         2147
```

```
##  7         2148
##  8         2149
##  9         2150
## 10         2151
## 11         8893
## 12        10031
## 13        25525
## 14        32037
## 15        45484
## 16        50852
## # ... with 7 more variables: title <chr>, author <chr>,
## #   gutenberg_author_id <int>, language <chr>, gutenberg_bookshelf <chr>,
## #   rights <chr>, has_text <lgl>

poe<-gutenberg_download(1064)
```

Now we are ready for some data cleaning.

2 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(dplyr)
library(tidytext)
poe_words<-poe%>%
  unnest_tokens(word,text)
```

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

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

poe_words

## # A tibble: 921 x 2
##    gutenberg_id       word
##           <int>      <chr>
##  1         1064     masque
##  2         1064        red
##  3         1064      death
##  4         1064      edgar
##  5         1064      allan
##  6         1064        poe
```

```
##  7         1064        red
##  8         1064      death
##  9         1064 devastated
## 10         1064    country
## # ... with 911 more rows
```

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

```
library(dplyr)
poe_freq<-poe_words%>%
  group_by(word)%>%
  summarize(count=n())
```

Finally, it's time to generate the wordcloud:

```
library(wordcloud)
wordcloud(poe_freq$word,poe_freq$count)
```


movement
ebony
stood
clock
presence
prospero
prince's
foot
horror
revellers
heavy
life
chamber
bold
apartments
velvet
courtiers
hour
mad
gay
deep
purple
night
scarlet
red
terror
black
sharp
hand
time
carpet
thousand
fro
walls
half
mummer
company
light
prince
blue
blood
tinted
assembly
suite
wild
moment
waltzers
tall
dreams
colour
green
wall
panes
echoes
music
apartment
windows
sable
death
chambers
figure
revel
ceased
hue
musicians