The Shunned House Wordcloud

Andrew Innes

November 1, 2017

Abstract

In this article we construct a wordcloud, using the tidytext R package, for H.P. Lovecraft's The Shunned House.

The Shunned House is a horror fiction novelette by American author H.P. Lovecraft, published in 1937[1]. Below we craft a wordcloud for the most common words appearing in the novelette.

1 The Gutenberg Package

The Gutenberg Package is a package for R, gutenbergr, that gives one access to the books in Project Gutenberg. One has to first install this package and bring it in with library. You may then call the following function and store the result. Since we will be using The Shunned House we will download it using its unique integer identifier. In order to do this we must execute the following code:

```
library(gutenbergr)
library(stringr)
gutenberg_works(str_detect(title,'The Shunned House'))
```

```
## # A tibble: 1 x 8
##   gutenberg_id             title                              author
##          <int>             <chr>                               <chr>
## 1        31469 The Shunned House Lovecraft, H. P. (Howard Phillips)
## # ... with 5 more variables: gutenberg_author_id <int>, language <chr>,
## #   gutenberg_bookshelf <chr>, rights <chr>, has_text <lgl>
```

```
House<-gutenberg_download(31469)

House
```

```
## # A tibble: 1,065 x 2
##    gutenberg_id
```

[1]The novel was published in the October 1937 issue of Weird Tales.

```
##           <int>
##  1        31469
##  2        31469
##  3        31469
##  4        31469
##  5        31469
##  6        31469
##  7        31469
##  8        31469
##  9        31469
## 10        31469
## # ... with 1,055 more rows, and 1 more variables: text <chr>
```

This dataframe has two columns, one for the The ID Number of the book, and one containing the text from the book. Now we are ready for very litte data cleaning.

2 Very Little Data Cleaning

We would like to remove the front matter of the book. By inspection, we have determined that the front matter ends on line 6. Therefore we can redefine House to begin on line 7:

```
library(dplyr)
House<-House[7:1055,]
```

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

words_df
```

```
## # A tibble: 10,968 x 2
##    gutenberg_id       word
##           <int>      <chr>
##  1        31469         _a
##  2        31469 posthumous
##  3        31469      story
```

```
##  4         31469         of
##  5         31469    immense
##  6         31469      power
##  7         31469    written
##  8         31469         by
##  9         31469          a
## 10         31469     master
## # ... with 10,958 more rows
```

We can remove the 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: 4,547 x 2
##    gutenberg_id       word
##           <int>      <chr>
##  1         31469         _a
##  2         31469 posthumous
##  3         31469      story
##  4         31469    immense
##  5         31469      power
##  6         31469    written
##  7         31469     master
##  8         31469      weird
##  9         31469    fiction
## 10         31469       tale
## # ... with 4,537 more rows
```

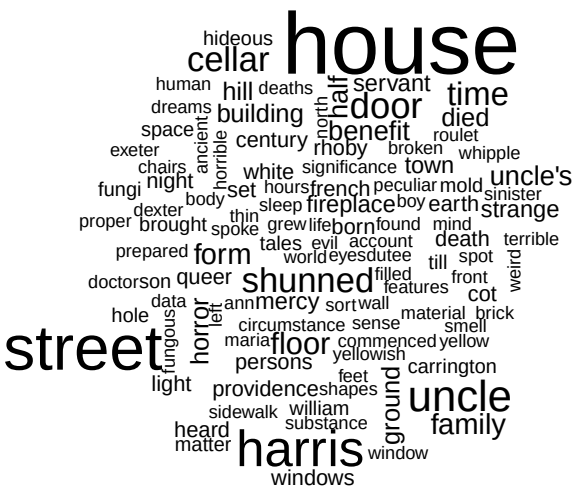
Now, we need to calculate the frequencies of the words in the novelette. Again, we can use standard dplyr techniques for this:

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

word_freq

## # A tibble: 2,652 x 2
##           word count
##          <chr> <int>
##  1          _a     1
##  2    _cellar_     1
```

```
##  3      _daily     1
##  4     _elbow_     1
##  5 _emanation_     1
##  6    _gaspee_     1
##  7       _had_     1
##  8         _in     1
##  9    _jacques     1
## 10 _providence     1
## # ... with 2,642 more rows
```

Finally, it's time to generate the wordcloud:

```
library(wordcloud)
wordcloud(word_freq$word,word_freq$count,min.freq = 5)
```



References

Silge, J. and Robinson, D. (2017). *Text Mining with R: A Tidy Approach.* O'Reilly Media.

Wickham, H. and Grolemund, G. (2017). *R for Data Science: Import,Tidy, Transform, and Model Data.* O'Reilly Media.