blog-2017-11-13 wordcloud & wordcloud2

HLB

Slug blog-2017-11-10-wordcloud-wordcloud2

Tidy text

Gutenbergr

Bing

Dracula

Text mining

reshape

summary: Creating a wordcloud & wordcloud 2 with tidy text, sentiment words and text mining in gutenbergr.

We are still working with text mining. Today we are going to create a wordcloud using the positive & negative sentiments mined from the text.

Just as always, we begin my loading (and library) the packages that we’ll use today: dplyr, gutenbergr, tidytext, tm (don’t have to library), wordcloud & (new) wordcloud2 & (also new) reshape2.

Now we need some text; we’ll download the novel Dracula from gutenbergr we’ve been working with this for a while & we know the download # is 345.

dracula<-gutenberg\_download(345)

Then we will ‘unpack’ the text column by unnesting the tokens with tidy text to get the words in Dracula. We’ll save this to a data frame Dracula\_words.

dracula\_words<-dracula%>%

unnest\_tokens(word,text)

We aren’t going to worry about the stop words (a, an, the, etc.) because we are going to work with ‘bing’ to rate the words as positive or negative.

bing<-get\_sentiments('bing')

dracula\_words

We’ll look at the data frame for dracula\_words & bing.

dracula\_words

bing

Dracula Words has 3 columns: the line number, Gutenberg id number & the word. Bing is the list of words & the sentiment attached to each word as positive or negative.

We’ll join the Dracula words & bing together.

dracula\_words<-inner\_join(dracula\_words,bing)

head(dracula\_words)

Now we have 4 columns: the line number, Gutenberg id number, the word & the sentiment.

We don’t need the Gutenberg id number column so we’ll get rid of that with NULL.

dracula\_words$gutenberg\_id<-NULL

head(dracula\_words)

We now have the information we want to make a word cloud.

**Wordcloud**

To get a wordcloud, we need to group the words & count by the sentiment. We’ll use freq(frequency) for n (rather than count.

dracula\_words<-dracula\_words%>%

group\_by(word)%>%

summarize(freq=n(), sentiment=first(sentiment))

Now a simple wordcloud:

wordcloud(dracula\_words$word, dracula\_words$freq,min.freq = 5)

Now let’s try wordcloud2, the new package we installed for today. For wordcloud 2 we need ca column for word & a column frequency (freq)

Wordcloud2

wordcloud2(dracula\_words)

Very simple code & the result is in color.

We can even specify the background color:

wordcloud2(dracula\_words,background ='black')

The other new package we installed is reshape2. This allows us to get a comparison wordcoud.

We need to change our data frame into a matrix. The ‘word’ column becomes our ‘row’ with each word representing a row & then we add/create 2 columns (‘negative & positive’).

We don’t want to see an ‘NA’ in our matrix so we need to make the NA 0 (zero)

fill = 0 is for Na to be 0

There will be a 0 (zero) in the positive column for negative words & a 0 (zero) in the negative column for positive words.

The code is: acast(dracula\_words,word~sentiment,value.var='freq',fill =0)

We’ll save this into a variable dracula\_matrix & look at the head.

dracula\_matrix<-acast(dracula\_words,word~sentiment,value.var='freq',fill =0)

head(dracula\_matrix)

We see the word and then a negative and positive column. The totals for each word are summed in these columns.

With wordcloud2 the comparison cloud is easy:

comparison.cloud(dracula\_matrix)

The positive & negative are in different colors. We can choose our own colors using colors=c(('black', 'orange'). We can use the htmlcolor codes that we find on the web too.

comparison.cloud(dracula\_matrix,colors=c('black', 'orange'))

Now our positive sentiments are orange & our negative sentiments are black.

Using Wordcloud 2 to ‘shape’ our cloud

Go out to google & find some clipart. Since this is a Halloween theme, we’ll look for a bat. When we google ‘free clip art bat” Download the .jpg, .gif, or .png file the clipart & put it in your working directory with your R script document.

For the blog, we’ll need to save it to our images.

Then we just add fig=’filename.jpg’ & it will shape the cloud.

We can use a different clip art, add a size & the background color too.

fin