The motivation for an updated analysis: Very recently, the nrc lexicon was dropped from the tidytext package and hence the R codes in the original publication failed to run. The NRC emotion terms are also available in the lexicon package.

This update provides a way around to leverage the NRC emotion terms made available in the lexicon package and show alternative R codes to parse text for emotion terms. The text data sets and the analysis figures are kept the same as the first publication.

require(tidyverse)

# assuming that the lexicon package has been installed already,

# Load the NRC emotions lexicon in memory, then

# Reshape the wide data format to a narrow data format

# Finally, keep emotions words(terms) in nrc\_data file

nrc\_data = lexicon::nrc\_emotions %>%

gather("sentiment", "flag", anger:trust, -term) %>%

filter(flag==1)

With the explosion of digital and social media, there are various emoticons and emojis that can be embedded in text messages, emails, or other various social media communications, for use in expressing personal feelings or emotions. Emotions may also be expressed in textual forms using words. R offers the lexicon or the get\_nrc\_sentiment function via Syuzhet packages for analysis of emotion words expressed in text.

I have two companion posts on this subject; this post is the first part. The motivations for this post are to illustrate the applications of some of the R tools and approaches:

* Analysis of emotion words in textual data
* Visualization and presentation of outputs and results

In the second part, unsupervised learning and differential expression of emotion words using R will be attempted.

**The Dataset**

Mr. Warren Buffett’s annual shareholder letters in the past 40-years (1977 – 2016) were downloaded from [this site](http://'http/www.berkshirehathaway.com/letters)

library(pdftools)

library(rvest)

library(XML)

# Getting & Reading in HTML Letters

urls\_77\_97 <- paste('<http://www.berkshirehathaway.com/letters/>', seq(1977, 1997), '.html', sep='')

html\_urls <- c(urls\_77\_97,

'<http://www.berkshirehathaway.com/letters/1998htm.html>',

'<http://www.berkshirehathaway.com/letters/1999htm.html>',

'<http://www.berkshirehathaway.com/2000ar/2000letter.html>',

'<http://www.berkshirehathaway.com/2001ar/2001letter.html>')

letters\_html <- lapply(html\_urls, function(x) read\_html(x) %>% html\_text())

# Getting & Reading in PDF Letters

urls\_03\_16 <- paste('<http://www.berkshirehathaway.com/letters/>', seq(2003, 2016), 'ltr.pdf', sep = '')

pdf\_urls <- data.frame('year' = seq(2002, 2016),

'link' = c('<http://www.berkshirehathaway.com/letters/2002pdf.pdf>', urls\_03\_16))

download\_pdfs <- function(x) {

myfile = paste0(x['year'], '.pdf')

download.file(url = x['link'], destfile = myfile, mode = 'wb')

return(myfile)

}

pdfs <- apply(pdf\_urls, 1, download\_pdfs)

letters\_pdf <- lapply(pdfs, function(x) pdf\_text(x) %>% paste(collapse=" "))

tmp <- lapply(pdfs, function(x) if(file.exists(x)) file.remove(x))

# Combine letters in a data frame

letters <- do.call(rbind, Map(data.frame, year=seq(1977, 2016), text=c(letters\_html, letters\_pdf)))

letters$text <- as.character(letters$text)

**Load additional required packages**

require(tidytext)

require(RColorBrewer)

require(gplots)

theme\_set(theme\_bw(12))

**Descriptive Statistics**

Analysis steps of emotion terms in textual data included word tokenization, pre-processing of tokens to exclude stop words and numbers and then invoking the get\_sentiment function using the Tidy package, followed by aggregation and presentation of results. Word tokenization is the process of separating text into single words or unigrams.

**Emotion words frequency and proportions**

total\_words\_count <- letters %>%

unnest\_tokens(word, text) %>%

anti\_join(stop\_words, by = "word") %>%

filter(!grepl('[0-9]', word)) %>%

group\_by(year) %>%

summarize(total= n()) %>%

ungroup()

emotion\_words\_count <- letters %>%

unnest\_tokens(word, text) %>%

anti\_join(stop\_words, by = "word") %>%

filter(!grepl('[0-9]', word)) %>%

inner\_join(nrc\_data, by=c("word"="term")) %>%

group\_by(year) %>%

summarize(emotions= n()) %>%

ungroup()

emotions\_to\_total\_words <- total\_words\_count %>%

left\_join(emotion\_words\_count, by="year") %>%

mutate(percent\_emotions=round((emotions/total)\*100,1))

ggplot(emotions\_to\_total\_words, aes(x=year, y=percent\_emotions)) +

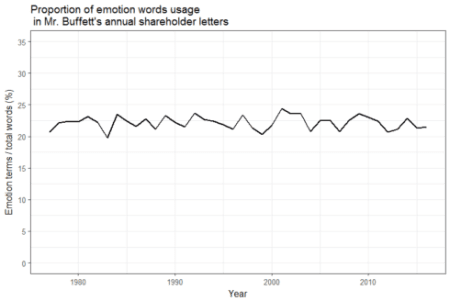
geom\_line(size=1) +

scale\_y\_continuous(limits = c(0, 35), breaks = c(0, 5, 10, 15, 20, 25, 30, 35)) +

xlab("Year") +

ylab("Emotion terms / total words (%)") + theme(legend.position="none") +

ggtitle("Proportion of emotion words usage \n in Mr. Buffett's annual shareholder letters")

Gives this plot:  
[](https://i0.wp.com/datascienceplus.com/wp-content/uploads/2017/05/Fig1.png?ssl=1)

Emotion words in the annual shareholder letters accounted for approximately 20% – 25% of the total words count (excluding stop words and numbers). The median emotion count was ~22% of the total words count.

**Depicting distribution of emotion words usage**

### pull emotion words and aggregate by year and emotion terms

emotions <- letters %>%

unnest\_tokens(word, text) %>%

anti\_join(stop\_words, by = "word") %>%

filter(!grepl('[0-9]', word)) %>%

inner\_join(nrc\_data, by=c("word"="term")) %>%

group\_by(year, sentiment) %>%

summarize( freq = n()) %>%

mutate(percent=round(freq/sum(freq)\*100)) %>%

select(-freq) %>%

ungroup()

### need to convert the data structure to a wide format

emo\_box = emotions %>%

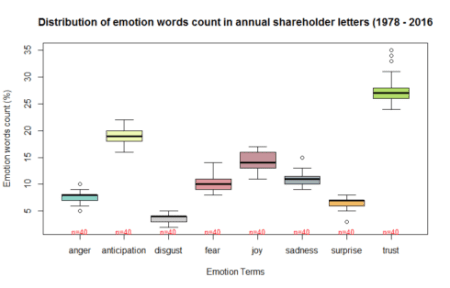
spread(sentiment, percent, fill=0) %>%

ungroup()

### color scheme for the box plots (This step is optional)

cols <- colorRampPalette(brewer.pal(7, "Set3"), alpha=TRUE)(8)

boxplot2(emo\_box[,c(2:9)], col=cols, lty=1, shrink=0.8, textcolor="red", xlab="Emotion Terms", ylab="Emotion words count (%)", main="Distribution of emotion words count in annual shareholder letters (1978 - 2016")

Which gives this plot:  
[](https://i0.wp.com/datascienceplus.com/wp-content/uploads/2017/05/Fig2_boxplot.png?ssl=1)

Terms for all eight emotions types were expressed albeit at variable rates. Looking at the box plot, anger, sadness, surprise, and trust showed outliers. Besides, anger, disgust and surprise were skewed to the left, whereas Joy was skewed to the right. The n= below each box plot indicates the number of observations that contributed to the distribution of the box plot above it.

**Emotion words usage over time**

## yearly line chart

ggplot(emotions, aes(x=year, y=percent, color=sentiment, group=sentiment)) +

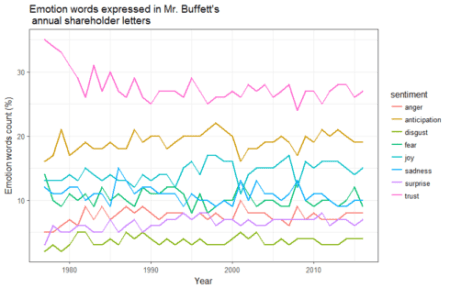
geom\_line(size=1) +

geom\_point(size=0.5) +

xlab("Year") +

ylab("Emotion words count (%)") +

ggtitle("Emotion words expressed in Mr. Buffett's \n annual shareholder letters")

Gives this plot:  
[](https://i1.wp.com/datascienceplus.com/wp-content/uploads/2017/05/Fig3_Linechart.png?ssl=1)

Clearly emotion terms referring to trust and anticipation were expressed consistently higher than the other emotion terms in all of the annual shareholder letters. Emotion terms referring to disgust, anger and surprise were expressed consistently lower than the other emotion terms at almost all time points.

**Average emotion words expression using bar charts with error bars**

### calculate overall averages and standard deviations for each emotion term

overall\_mean\_sd <- emotions %>%

group\_by(sentiment) %>%

summarize(overall\_mean=mean(percent), sd=sd(percent))

### draw a bar graph with error bars

ggplot(overall\_mean\_sd, aes(x = reorder(sentiment, -overall\_mean), y=overall\_mean)) +

geom\_bar(stat="identity", fill="darkgreen", alpha=0.7) +

geom\_errorbar(aes(ymin=overall\_mean-sd, ymax=overall\_mean+sd), width=0.2,position=position\_dodge(.9)) +

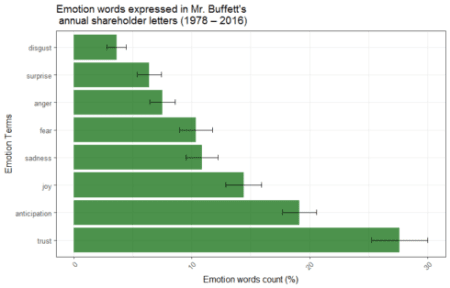
xlab("Emotion Terms") +

ylab("Emotion words count (%)") +

ggtitle("Emotion words expressed in Mr. Buffett's \n annual shareholder letters (1977 – 2016)") +

theme(axis.text.x=element\_text(angle=45, hjust=1)) +

coord\_flip( )

Which gives this plot:  
[](https://i2.wp.com/datascienceplus.com/wp-content/uploads/2017/05/Fig4_Barchart.png?ssl=1)

Emotion words referring to trust, anticipation and joy were over-represented and accounted on average for approximately 60% of all emotion words in all shareholder letters. On the other hand, disgust, surprise and anger were the least expressed emotion terms and accounted on average for approximately 18% of all emotion terms in all shareholder letters.

**Emotion terms usage over time compared to 40-years averages**

For the figure below, the 40-year averages of each emotion terms shown in the above bar chart were subtracted from the yearly percent emotions for any given year. The results were showing higher or lower than average emotion expression levels for the respective years.

## Hi / Low plots compared to the 40-years average

emotions\_diff <- emotions %>%

left\_join(overall\_mean\_sd, by="sentiment") %>%

mutate(difference=percent-overall\_mean)

ggplot(emotions\_diff, aes(x=year, y=difference, colour=difference>0)) +

geom\_segment(aes(x=year, xend=year, y=0, yend=difference),

size=1.1, alpha=0.8) +

geom\_point(size=1.0) +

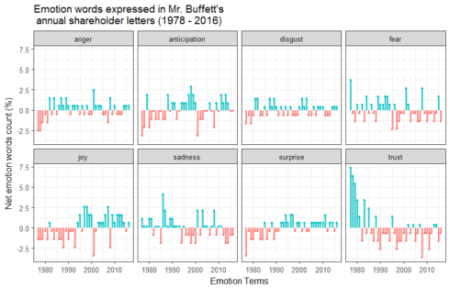
xlab("Emotion Terms") +

ylab("Net emotion words count (%)") +

ggtitle("Emotion words expressed in Mr. Buffett's \n annual shareholder letters (1977 - 2016)") +

theme(legend.position="none") +

facet\_wrap(~sentiment, ncol=4)

Gives this plot:  
[](https://i2.wp.com/datascienceplus.com/wp-content/uploads/2017/05/Fig5_HiLow_plots.png?ssl=1)

Red lines show lower than the 40-year average emotion expression levels, while blue lines indicate higher than the 40-year average emotion expression levels for the respective years.

**Concluding Remarks**

Excluding stop words and numbers, approximately 1 in 4 words in the annual shareholder letters represented emotion terms. Clearly, emotion terms referring to trust, anticipation and joy accounted for approximately 60% of all emotion terms. There were also very limited emotions of fear (approximately 1 in 10 emotion terms). In conclusion, R offers several packages and functions for the evaluation and analyses of emotions words in textual data, as well as visualization and presentation of analysis results. Some of those packages and functions have been illustrated in this post. Hopefully, you find this post and analyses and visualization examples helpful.