Project 06

Jason Grahn

2/15/2019

knitr::opts\_chunk$set(echo = TRUE, message = FALSE)  
  
## A Tidytext example from https://www.tidytextmining.com/ would be   
## more modern and applicable for student learning.   
  
## Instead, let's install a bunch of dead packages because this code   
## is old and out of date.   
  
require(devtools)  
  
# install\_url("http://www.omegahat.org/Rstem/Rstem\_0.4-1.tar.gz")   
## this is a dead URL  
  
# note the original URL is dead too. replace the .org with .net and the download works correctly.   
# install\_url("http://www.omegahat.net/Rstem/Rstem\_0.4-1.tar.gz")  
# install\_url("http://cran.r-project.org/src/contrib/Archive/sentiment/sentiment\_0.2.tar.gz")

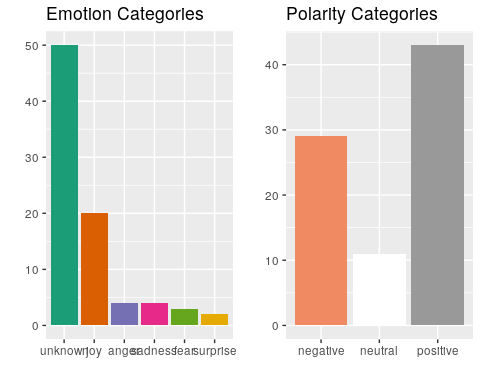
## It's incredibly bad practice to ship code that asks for packages to be installed. This would be better as a docker image or a use\_this:: package.  
  
# install.packages("plyr")   
# install.packages("ggplot2")   
# install.packages("wordcloud")   
# install.packages("RColorBrewer")   
# install.packages("tm")   
# install.packages("SnowballC")  
  
library(tidyverse)  
library(wordcloud)  
library(RColorBrewer)  
library(tm)  
library(SnowballC)  
library(sentiment)

set.seed(42)  
data <- readLines("http://www.r-bloggers.com/wp-content/uploads/2016/01/vent.txt")   
 # from: http://www.wvgazettemail.com/   
df <- data.frame(data)   
textdata <- df[df$data, ]   
textdata = gsub("[[:punct:]]", "", textdata)

textdata = gsub("[[:punct:]]", "", textdata)   
textdata = gsub("[[:digit:]]", "", textdata)   
textdata = gsub("http\\w+", "", textdata)   
textdata = gsub("[ \t]{2,}", "", textdata)   
textdata = gsub("^\\s+|\\s+$", "", textdata)   
  
try.error = function(x) {   
 y = NA   
 try\_error = tryCatch(tolower(x), error=function(e) e)   
 if (!inherits(try\_error, "error"))   
 y = tolower(x)   
 return(y)   
 }   
  
textdata = sapply(textdata, try.error)   
textdata = textdata[!is.na(textdata)]   
names(textdata) = NULL

class\_emo = classify\_emotion(textdata, algorithm="bayes", prior=1.0)  
emotion = class\_emo[,7]  
emotion[is.na(emotion)] = "unknown"  
class\_pol = classify\_polarity(textdata, algorithm="bayes")  
polarity = class\_pol[,4]  
  
sent\_df = data.frame(text=textdata, emotion=emotion, polarity=polarity, stringsAsFactors=FALSE)  
  
sent\_df = within(sent\_df, emotion <- factor(emotion,   
 levels=names(sort(table(emotion),   
 decreasing=TRUE))))

emotion\_plot <- ggplot(sent\_df, aes(x=emotion)) +   
 geom\_bar(aes(y=..count.., fill=emotion)) +   
 scale\_fill\_brewer(palette="Dark2") +   
 labs(title = "Emotion Categories",  
 x="",   
 y="") +  
 theme(legend.position = "none")  
  
polarity\_plot <- ggplot(sent\_df, aes(x=polarity)) +  
 geom\_bar(aes(y=..count.., fill=polarity)) +  
 scale\_fill\_brewer(palette="RdGy") +  
 labs(title="Polarity Categories",   
 x = "",  
 y="") +  
 theme(legend.position = "none")  
  
cowplot::plot\_grid(emotion\_plot,polarity\_plot)



emos = levels(factor(sent\_df$emotion))  
nemo = length(emos)  
emo.docs = rep("", nemo)  
  
for (i in 1:nemo){   
 tmp = textdata[emotion == emos[i]]   
 emo.docs[i] = paste(tmp, collapse=" ")   
}  
  
emo.docs = removeWords(emo.docs, stopwords("english"))  
corpus = Corpus(VectorSource(emo.docs))  
tdm = TermDocumentMatrix(corpus)  
tdm = as.matrix(tdm)  
colnames(tdm) = emos

comparison.cloud(tdm,   
 colors = brewer.pal(nemo, "Dark2"),  
 scale = c(3,.5),   
 random.order = FALSE,  
 title.size = 1.5)

