Algorithm Aversion

Kopkow, Angermaier, Rohrer, Sonnleitner

February 2022

Contents

Motivation: Erwin

Data retrieval &

Data processing: Mathias

```r
library(tidyverse)
library(tidytext)
library(textdata)
library(dplyr)
library(vader)
library(academictwitteR)
library(data.table)
library(readr)
```

used libraries

```
## `summarise()` has grouped output by 'alg.data$Year'. You can override using the `.groups` argument.

## `summarise()` has grouped output by 'Year'. You can override using the `.groups` argument.
```

```r
listofdfs <- list()

#data group for cummulated dataframe

for (i in c(7:13)){
  alg.data %>%
    filter(alg.data[,i]>=1)%>%
    mutate(Topic = colnames(alg.data)[i], VADERclass=as.factor(VADERclass))%>%
    group_by(Year,Topic,VADERclass, .drop=FALSE)%>%
    summarise(Sent = n(), .groups = "drop") ->listofdfs[[i]]
}
summ.cumm <- bind_rows(listofdfs)


#adds percentage to cummulated DF
summ.cumm <-
  summ.cumm %>%
  group_by(Topic, Year) %>%
  mutate(All = sum(Sent),percent=(100*Sent/All))

summ.cumm
```

```
## # A tibble: 324 x 6
## # Groups:   Topic, Year [84]
##     Year Topic    VADERclass  Sent   All percent
##    <dbl> <chr>    <fct>      <int> <int>   <dbl>
##  1  2010 Business Aversive       2   297   0.673
##  2  2010 Business Negative      14   297   4.71
##  3  2010 Business Neutral      226   297  76.1
##  4  2010 Business Positive      55   297  18.5
##  5  2011 Business Aversive       7   337   2.08
##  6  2011 Business Negative      17   337   5.04
##  7  2011 Business Neutral      217   337  64.4
##  8  2011 Business Positive      96   337  28.5
##  9  2012 Business Aversive       4   357   1.12
## 10  2012 Business Negative      36   357  10.1
## # ... with 314 more rows
```

```r
#data group for topic oriented dataframe

listofdfs <- list()

for (i in c(7:13)){
  alg.data %>%
    filter(alg.data[,i]>=1)%>%
    mutate(Topic = colnames(alg.data)[i])%>%
    group_by( VADERclass, Topic) %>%
    summarise(Sent = n()) ->listofdfs[[i]]
}
```

```
## `summarise()` has grouped output by 'VADERclass'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'VADERclass'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'VADERclass'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'VADERclass'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'VADERclass'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'VADERclass'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'VADERclass'. You can override using the `.groups` argument.
```

```r
summ.topic <- bind_rows(listofdfs)

#adds percentage to topicwise DF
summ.topic<-
  summ.topic %>%
  group_by(Topic) %>%
  mutate(All = sum(Sent),percent=(100*Sent/All))


head(summ.topic)
```

```
## # A tibble: 6 x 5
## # Groups:   Topic [2]
##   VADERclass Topic         Sent   All percent
##   <chr>      <chr>        <int> <int>   <dbl>
## 1 Aversive   Business        73  3928    1.86
## 2 Negative   Business       373  3928    9.50
## 3 Neutral    Business      2165  3928   55.1
## 4 Positive   Business      1317  3928   33.5
## 5 Aversive   Social.Media  1601 43621    3.67
## 6 Negative   Social.Media  5918 43621   13.6
```

```r
#--------------Test wordgroups---------------------
listofdfs <- list()


for (i in c(7:13)){
  alg.data %>%
    filter(alg.data[,i]>=1)%>%
    mutate(Topic = colnames(alg.data)[i])%>%
    sample_n(size = 20)%>%
  group_by( VADERclass, Topic)  ->listofdfs[[i]]
```

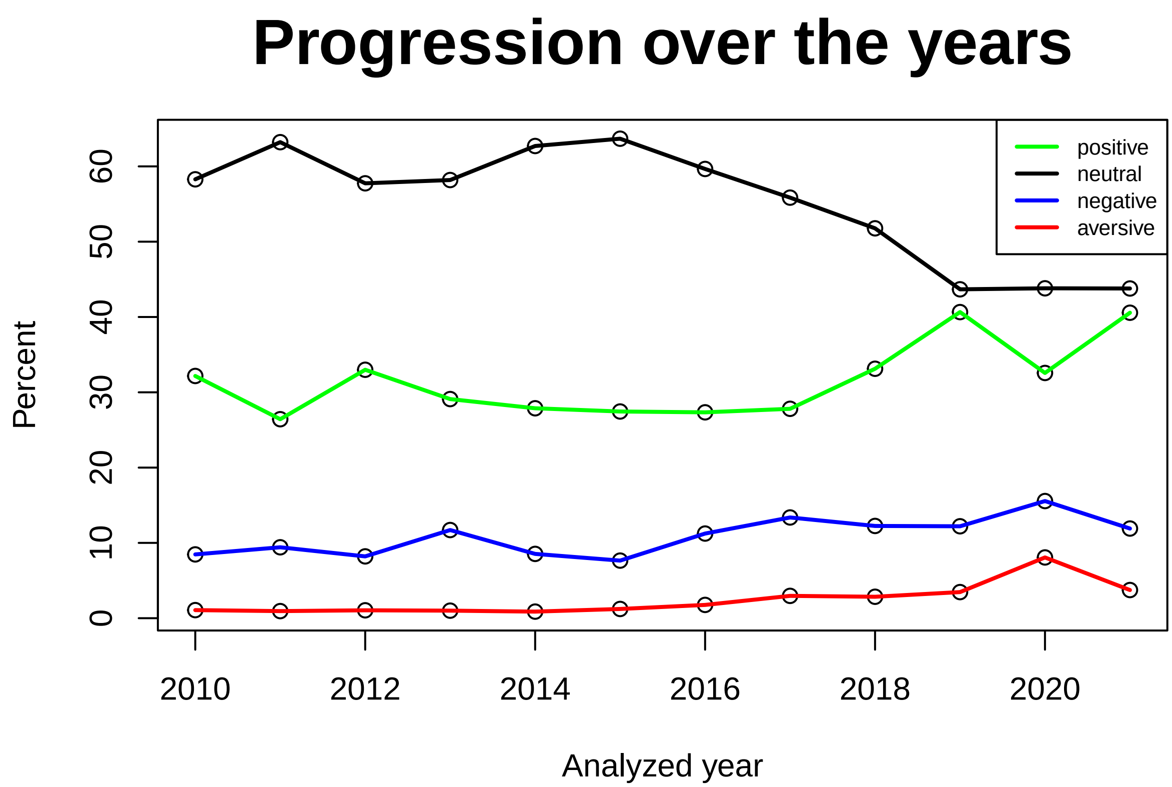
```
}
Wordgrouptest<- bind_rows(listofdfs)
```

Analysis

a

Progression over the years

The diagram shows the percentage progression of the results of the sentiment analysis of tweets over the years 2010 to 2021. Values between 0.3 and -0.3 were classified as neutral. As positive between 0.3 and 1, as negative between -0.3 and -0.7 and finally as aversive values between -0.7 and -1. A total of 143271 tweets were analyzed. The fewest tweets were aversive and most were neutral since all tweets not evaluated in the sentiment analysis are classified as neutral.



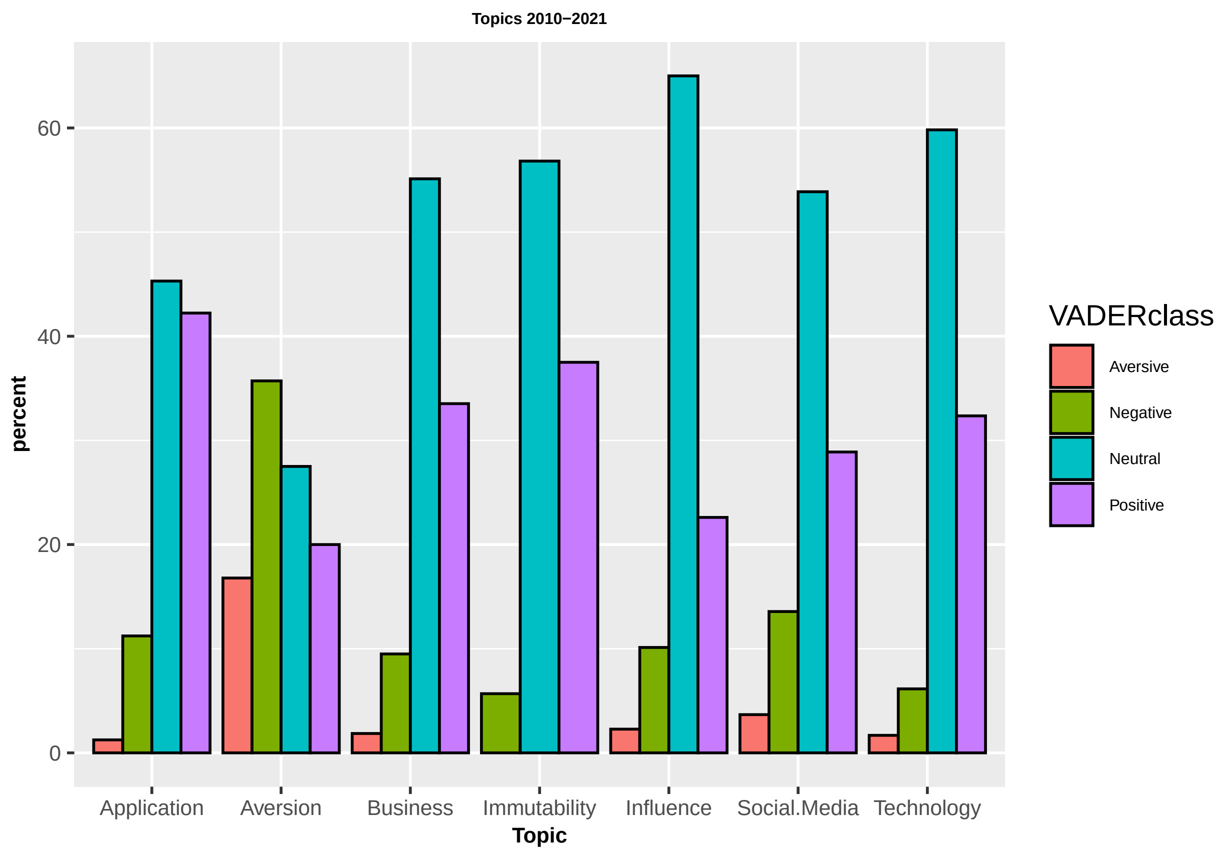
Over time, the number of neutral tweets decreases. In 2010, there were 6934 tweets and 58.3%. In 2015, the most neutral tweets were posted with an absolute of 7641 tweets and 63.8% posted. In 2019 the fewest mi 5241 and 43.7% were posted. 2021 total 5254 which is 43.8%. The last 3 years the number of neutral tweets remained at the same level.

An increasing trend can be seen in the number of positive tweets. In 2010, there were 3827 tweets and 32.2%. In 2019, the most positive tweets were posted with absolute 4878 tweets and 40.7% posted. In 2011 the least mi 3171 and 26.7% were posted. 2021 total 4868 which is 40.6%. The last 3 years the number of positive tweets remained high, with a bump in 2020 with 3908 and 32.6%.

Negative tweets show an increasing trend. In 2010, there were 1008 tweets and 8.5%. In 2020 the most positive tweets were posted with absolute 1867 tweets and 15.6% posted. In 2015 the least mi 918 and 7.7% were posted. In 2021 a total of 1429 which gives 11.9%.

The aversive tweets behave similarly to the negative tweets 2011 113 0.95% there were the fewest were 2014 105 and 0.88% with the most there were 2020 968 8.1% and 2021 there were 3.7% and 449 tweets.

Distribution of classes by topic




```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 10000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = results, type = "bca")
##
## Intervals :
## Level       BCa
## 95%   ( 0.0604,  0.1197 )
## Calculations and Intervals on Original Scale
```

Hier werden die verschieden Topics Applicattion Aversion Buisness Immutability Influeence Social Media und Technology verglichen

Conclusion:

Critique: Alina

Github Repository

https://github.com/sonnleit/AlgorithmAversion

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
#summary(cars)
```

Including Plots

You can also embed plots, for example:

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

Header 1

Header 2

Header 3

Header 4

Header 5

Header 6