Tweet Feature Extraction

Ada Lazuli

2022-07-17

Contents

```
library(tidyr)
library(dplyr)
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
library(ggplot2)
library(stringr)
library(tidytext)
library(forcats)
library(textdata)
```

Data Loading

Load the prepped data compiled by Winfrey Johnson.

```
df <- read.csv("../data/tweets_prepped_no_http.csv")
glimpse(df)
```

```
## Rows: 8,560
## Columns: 6
## $ tweet      <chr> "cdc currently reports deaths general discrepancies death c~
## $ label      <chr> "real", "real", "fake", "real", "real", "real", "real", "fa~
## $ tweet_orig <chr> " cdc currently reports deaths general discrepancies death ~
## $ hashtag    <chr> NA, NA, " coronavirus nashville ", " indiafightscorona covi~
## $ link       <lgl> FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, TRU~
## $ ssl_link   <lgl> FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE, TRU~
```

Data Enrichment

To give the decision tree more features to work with, the data will be enriched to include: 1. A flag on whether HTTP is present 2. A numeric designation on overall sentiment using `afinn` 3. A other sentiment labels from the `nrc` corpus.

Add Length

Use the `mutate` verb to add the length of the tweet to the dataframe

```
df <- df %>% mutate(length = nchar(tweet))
```

Flag HTTP

Use `mutate` to find the http keyword in the tweet text then convert that logical to an integer.

```
df <- df %>%
  mutate(http = grepl("http", tweet)) %>%
  mutate(http = as.integer(http))
```

Add Integer Sentiment

First, create a function that calculate the sentiment positivity value for one tweet using `afinn`. Then apply that function over all of the tweets in the dataframe.

```
calc_afinn_sentiment <- function(df){
  df$sentiment_afinn <- 0
  for (i in 1 : nrow(df)){
    # extract the row
    temp <- df[i,]
    # un nest the row to join with afinn then calculate the average sentiment
    temp <- temp %>%
      unnest_tokens(word, tweet) %>%
      inner_join(get_sentiments("afinn"), by = "word") %>%
      group_by(label) %>%
      summarize(sentiment = mean(value))
    # sometimes the sentiment is not returned if the intersection of the sets is empty, so check for th
    if (length(temp$sentiment) > 0){
      df[i,]$sentiment_afinn <- temp$sentiment[1]
    }
  }
  return (df)
}
df <- calc_afinn_sentiment(df)
```

Add Sentiement Labels

Similar to the previous section, create a function to add sentiment labels to each row of the dataframe using the `nrc` dataset.

```
find_nrc_sentiment <- function(df){
  # Add empty columns to the dataset for each label prior to calculation
  nrc_sentiments <- c("anger", "anticipation", "disgust",  "fear", "joy", "negative",  "positive", "sadr
  for (x in 1:length(nrc_sentiments)){
    df[nrc_sentiments[x]] <- 0
  }

    for (i in 1 : nrow(df)){
    # extract the row
    temp <- df[i,]
    # un nest the row to join with nrc then count the occurrences of each sentiment
    temp <- temp %>%
      unnest_tokens(word, tweet) %>%
      inner_join(get_sentiments("nrc"), by = "word") %>%
      count(sentiment)
    # sometimes the sentiment is not returned if the intersection of the sets is empty, so check for th
    if (nrow(temp) > 0){
      for (x in 1:nrow(temp)){
        df[i,][temp[x,]$sentiment] <- temp[x,]$n
      }
    }
  }
  return(df)
}
df <- find_nrc_sentiment(df)
```

Save Enriched Dataset

Finally, save the enriched dataset as a CSV.

```
write.csv(df, "../data/enriched_tweet_data.csv", row.names = FALSE)

glimpse(df)
```

```
## Rows: 8,560
## Columns: 19
## $ tweet           <chr> "cdc currently reports deaths general discrepancies de~
## $ label           <chr> "real", "real", "fake", "real", "real", "real", "real"~
## $ tweet_orig      <chr> " cdc currently reports deaths general discrepancies d~
## $ hashtag         <chr> NA, NA, " coronavirus nashville ", " indiafightscorona~
## $ link            <lgl> FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE~
## $ ssl_link        <lgl> FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FALSE~
## $ length          <int> 138, 79, 72, 71, 147, 161, 109, 57, 66, 115, 110, 110,~
## $ http            <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ sentiment_afinn <dbl> -2.00, 0.00, -1.00, 0.00, 0.00, 2.00, 2.00, -2.00, 1.0~
## $ anger           <dbl> 2, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, ~
## $ anticipation    <dbl> 2, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 2, ~
## $ disgust         <dbl> 2, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ fear            <dbl> 2, 0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, ~
## $ joy             <dbl> 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ negative        <dbl> 3, 1, 2, 0, 1, 1, 1, 3, 1, 1, 0, 1, 0, 1, 1, 0, 3, 0, ~
## $ positive        <dbl> 1, 0, 0, 1, 0, 2, 1, 0, 1, 0, 2, 1, 2, 0, 5, 1, 0, 0, ~
## $ sadness         <dbl> 2, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, ~
## $ surprise        <dbl> 2, 0, 0, 0, 0, 0, 0, 2, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, ~
## $ trust           <dbl> 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, ~
```