**Step1: update positive words and negative words and add the following words into the word lists of Hu and Liu Opinion Lexicon.**

**Good Words (need to add to positive-words.txt)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| just | follow | support | positive | inspiring | strong |
| contribute | rebuild | endorse | ahead | smart | tough |
| favors | great | victory | restored | first | massive |
| proud | bright | humble | safe | success | affordable |
| nice | new | follow | future | got | equal |
| beautiful | praise | encouragement | happy | endorsement | progressive |
| thank | hope | right | dear | real | important |
| love | congratulations | vote | qualified | luck | presidential |
| help | save | supporter | historic | big | productive |
| win | good | open | passionate | special | comprehensive |
| grateful | fair | hardworking | wealthy | accomplished |  |

**Bad words (need to add to negative-words.txt)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| pledge | protest | prison | none | dirty | criminal |
| but | wrong | lawsuits | fool | destroy | condemn |
| crash | hate | trial | shame | over | down |
| golem | unfit | threat | swamp | hateful | gooner |
| crash | bad | waste | hiding | fault | liar |
| problem | lied | mess | undo | cheater | disaster |
| lying | scandal | corrupt | fake | cheated | hidden |
| shame | dumb | sad | not | lawlessness | deviousness |
| dishonest | loser | phony | hoax | danger | stupid |
| jail | rape | fraud | ties | guilty | crook |
| laugh | victim | nothing | evil | fear | disappointed |
| joke | embarrassing | negative | hard | serious | lightweight |
| con | shocked | lyin' | biased | reckless | protester |
| unlawful | rigged | weak | crazy | little | systemic |
| incarcerate | failing | illegal | goofy | crippled | dangerous |

**Step2: create three new word lists for better words, more negative words and policy words.**

**Wonderful words (need to add to wonderful-words.txt)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| bigger | better | perfect | safer | stronger | super |
| wonderful | amazing | fantastic |  |  |  |

**Terrible words (need to add to terrible-words.txt)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| nasty | predator | idiot | worst | shit | disgusting | fuck |
| disastrous | pesky | stiffed | dumbass | dictator | bitch | satan |

**Policy words (need to add to policy-words.txt)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| guns | terror | taxes | immigration | jobs | health | obamacare |
| education | foreign | vets | housing | isis | putin | russia |
| china | chinese | war | nuclear |  |  |  |

**Step3: create a scoring logic.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Word** | **Positive** | **Negative** | **Wonderful** | **Terrible** | **Policy** |
| **Score** | 1 | -1 | 2 | -2 | 1 |

Here is the score function:

score = sum(pos.matches) - sum(neg.matches) - sum(ter.matches)\*2 + sum(won.matches)\*2 + sum(pol.matches)

**Ste4: develop the scoring function:**

# function score.sentiment

score.sentiment = function(sentences, pos.words, neg.words, ter.words, won.words, pol.words, .progress='none')

{

# Parameters

# sentences: vector of text to score

# pos.words: vector of words of postive sentiment

# neg.words: vector of words of negative sentiment

# ter.words: vector of words of terrible sentiment

# won.words: vector of words of wonderful sentiment

# pol.words: vector of words of policy sentiment

# .progress: passed to laply() to control of progress bar

# create simple array of scores with laply

scores = laply(sentences,

function(sentence, pos.words, neg.words, ter.words, won.words, pol.words)

{

# remove punctuation

sentence = gsub("[[:punct:]]", "", sentence)

# remove control characters

sentence = gsub("[[:cntrl:]]", "", sentence)

# remove digits?

sentence = gsub('\\d+', '', sentence)

# define error handling function when trying tolower

tryTolower = function(x)

{

# create missing value

y = NA

# tryCatch error

try\_error = tryCatch(tolower(x), error=function(e) e)

# if not an error

if (!inherits(try\_error, "error"))

y = tolower(x)

# result

return(y)

}

# use tryTolower with sapply

sentence = sapply(sentence, tryTolower)

# split sentence into words with str\_split (stringr package)

word.list = str\_split(sentence, "\\s+")

words = unlist(word.list)

# compare words to the dictionaries of positive, negative, terrible, wonderful and policy terms

pos.matches = match(words, pos.words)

neg.matches = match(words, neg.words)

ter.matches = match(words, ter.words)

won.matches = match(words, won.words)

pol.matches = match(words, pol.words)

# get the position of the matched term or NA

# we just want a TRUE/FALSE

pos.matches = !is.na(pos.matches)

neg.matches = !is.na(neg.matches)

ter.matches = !is.na(ter.matches)

won.matches = !is.na(won.matches)

pol.matches = !is.na(pol.matches)

# final score

score = sum(pos.matches) - sum(neg.matches) - sum(ter.matches)\*2 + sum(won.matches)\*2 + sum(pol.matches)\*1

return(score)

}, pos.words, neg.words, ter.words, won.words, pol.words, .progress=.progress )

# data frame with scores for each sentence

scores.df = data.frame(text=sentences, score=scores)

return(scores.df)

}

**Ste5: call the function**

# Call the function and return a data frame

feelthatrump <- score.sentiment(feed\_trump, pos, neg, ter, won, pol, .progress='text')

feelthaclinton <- score.sentiment(feed\_clinton, pos, neg, ter, won, pol, .progress='text')

**Step6: sum the scores for each candidates**

# sum the total scores

resTrumpe <- c(sum(feelthatrump$score))

resClinton <- c(sum(feelthaclinton$score))

print(resTrumpe)

print(resClinton)

**Here are the results:**

**Trumpe’ total score is 80.**

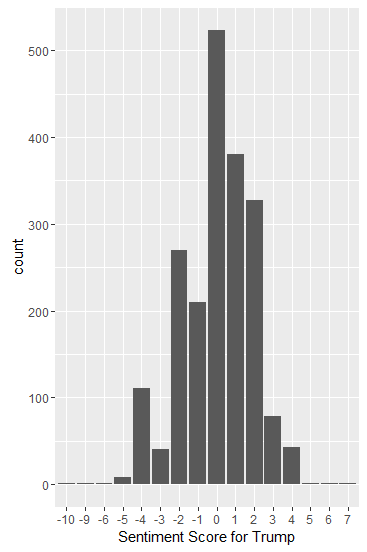
**Clinton’s total score is 542.**

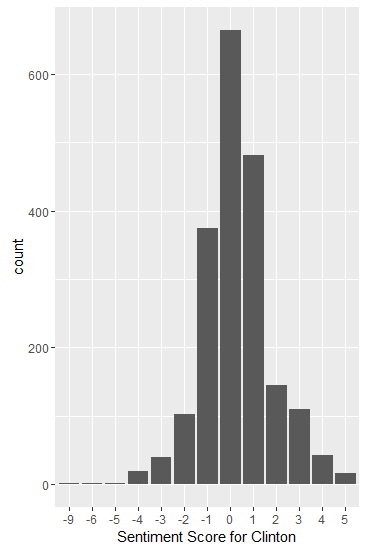
**Step7: plots**

# Nice little quick plot

qplot(factor(score), data=feelthatrump, geom="bar", xlab = "Sentiment Score for Trump")

qplot(factor(score), data=feelthaclinton, geom="bar", xlab = "Sentiment Score for Clinton")





**Step8: calculate these two candidates’ scores which twitter’s score greater than 2 or less than -2.**

# how many tweets of each

nd = c(length(feed\_trump), length(feed\_clinton))

# join texts

candidates = c(feed\_trump, feed\_clinton)

# apply function score.sentiment

scores = score.sentiment(candidates, pos, neg, ter, won, pol, .progress='text')

# add variables to data frame

scores$candidates = factor(rep(c("@realDonaldTrump", "@HillaryClinton"), nd))

scores$very.pos = as.numeric(scores$score >= 2)

scores$very.neg = as.numeric(scores$score <= -2)

# how many very positives and very negatives

numpos = sum(scores$very.pos)

numneg = sum(scores$very.neg)

# global score

global\_score = round( 100 \* numpos / (numpos + numneg) )

# colors

cols = c("red", "blue")

names(cols) = c("@realDonaldTrump", "@HillaryClinton")

# boxplot

ggplot(scores, aes(x=candidates, y=score, group=candidates)) +

geom\_boxplot(aes(fill=candidates)) +

scale\_fill\_manual(values=cols) +

geom\_jitter(colour="gray40",

position=position\_jitter(width=0.2), alpha=0.3) +

labs(title = "Boxplot - Presidential Candidates' Sentiment Scores")

