Final Project - Analyzing Use of Verbs in Tweets

Deanna Schneider

October 20, 2017

Step One - Read in the data that was output from Sentiment Analysis.ipynb, clean it, and sample it

Two files were generated: metoo_cleaned.csv and takeaknee_cleaned.csv - each of which contained between 7000 and 1800 tweets, collected on 10/26/17 and 12/04/17. Collected data was analyzed in Python and exported with the addition of sentiment and verb counts.

Each data set has more than the number of tweets I ultimately want (5000). In the following steps, I will make sure that we have only full-text tweets, only 1 tweet per author, all unique tweets, remove non-native English speakers, and verify that there is some content left in the tweet after it had been cleaned with the regular expression in Python. After cleaning the tweets, I will get a random set of 5000 tweets from each hashtag.

```
pacman::p_load('tidyverse')

#define a function that does all the cleaning in R - Note, code to do this
all seperately is FinalProject-babysteps.RMD
clean_data <- function(filename, seed = 71, samplesize = 5000){
  #The file Utilities.R includes timings for read.csv, scan, and tidyverse's
read_csv. Read_csv is the winner.
  library(tidyverse)
  #read in the file
  out_data = read_csv(filename)
  #remove truncated tweets
  out_data <- out_data[which(out_data["truncated"]=='FALSE'), ]

  #get unique authors
  out_data <- out_data[!duplicated(out_data$AuthorID),]
  #get unique tweets
  out_data <- out_data[!duplicated(out_data$Text),]
  #get only native English Speakers
  out_data <- out_data[out_data$Language == 'en',]
  #make sure there's something in the clean tweet
  out_data <- out_data[length(out_data$cleanTweet) > 0,]
  #set a seed
  set.seed(seed)
  #return a random sample of 5000 tweets

  out_data <- out_data[sample(nrow(out_data), samplesize), ]
```

```
  return(out_data)
}

metoo <- clean_data('metoo_cleaned.csv', samplesize=2000)
metoo_split <- clean_data('metoo_cleaned_split.csv')
knee <- clean_data('takeaknee_cleaned.csv')
knee_split <- clean_data('takeaknee_cleaned_split.csv')
dim(metoo)
dim(metoo_split)
dim(knee)
dim(knee_split)

#fetch a random sample from each dataset, for Table 1
#set the seed (so this can be reproduced)
set.seed(71)

metoo_tweet <- metoo[sample(nrow(metoo), 1), ]
knee_tweet <- knee[sample(nrow(knee), 1), ]

print(paste(metoo_tweet$AuthorID, ' - ', metoo_tweet$cleanTweet))
```

```
## [1] "221031253  -  two colleges bound by history are roiled by the metoo
moment"
```

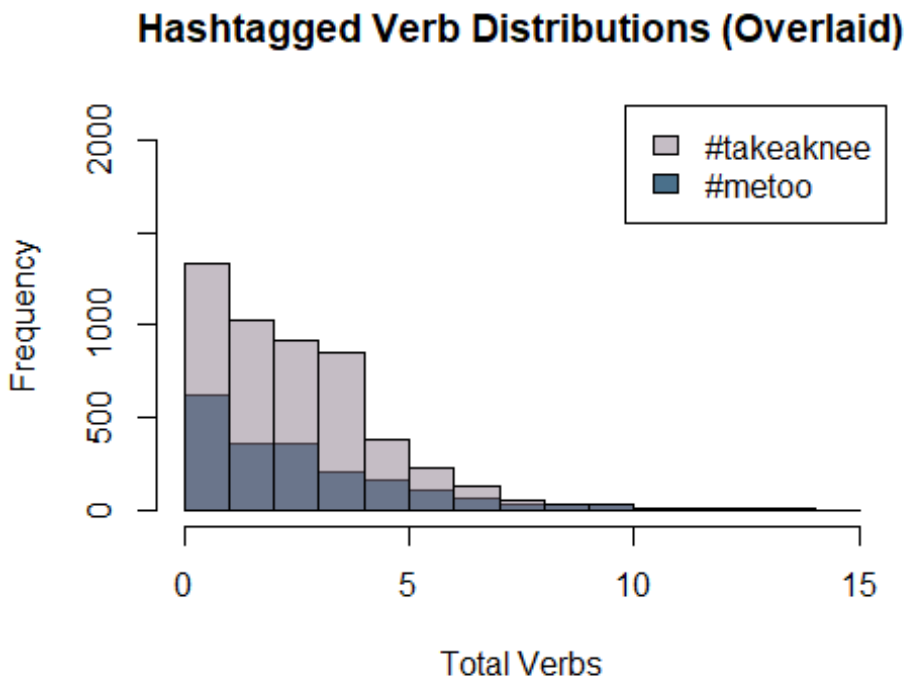
```
print(paste(knee_tweet$AuthorID, ' - ', knee_tweet$cleanTweet))
```

```
## [1] "899809169221365760  -  takeaknee just stand up"
```

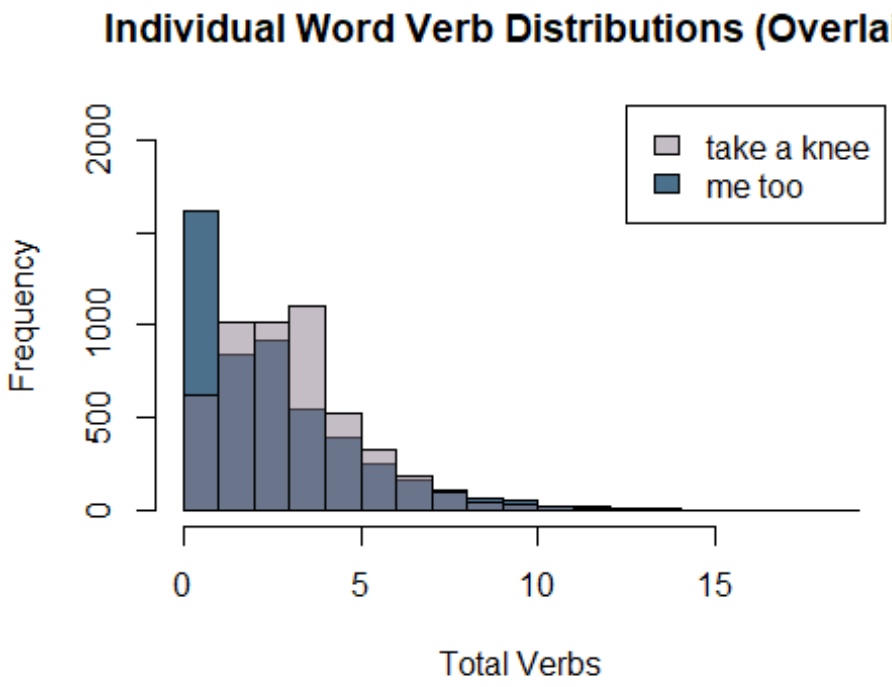
Step Three - Review Verb Distribution

In order to use a t.test, we need to have a sizable sample (requirement met), and ideally have a normal distribution. In the following code cells, I produce histograms of the data to review the distributions. I also review the distribution of the logged data.

```
#look at the distribution of verbs in each dataset
hist(metoo$total_verbs, main="Hashtagged Verb Distributions (Overlaid)",
xlab="Total Verbs", col='skyblue4', ylim=c(0, 2100))
hist(knee$total_verbs, col='#8B7B8B7F', ylim=c(0,2100), add=T)
legend("topright", c('#takeaknee', '#metoo'), fill=c('#8B7B8B7F',
'skyblue4'))
```




```
#look at the distribution of verbs in each dataset
hist(metoo_split$total_verbs, main="Individual Word Verb Distributions
(Overlaid)", xlab="Total Verbs", col='skyblue4', ylim=c(0, 2100))
hist(knee_split$total_verbs, col='#8B7B8B7F', ylim=c(0,2100), add=T)
legend("topright", c('take a knee', 'me too'), fill=c('#8B7B8B7F',
'skyblue4'))
```

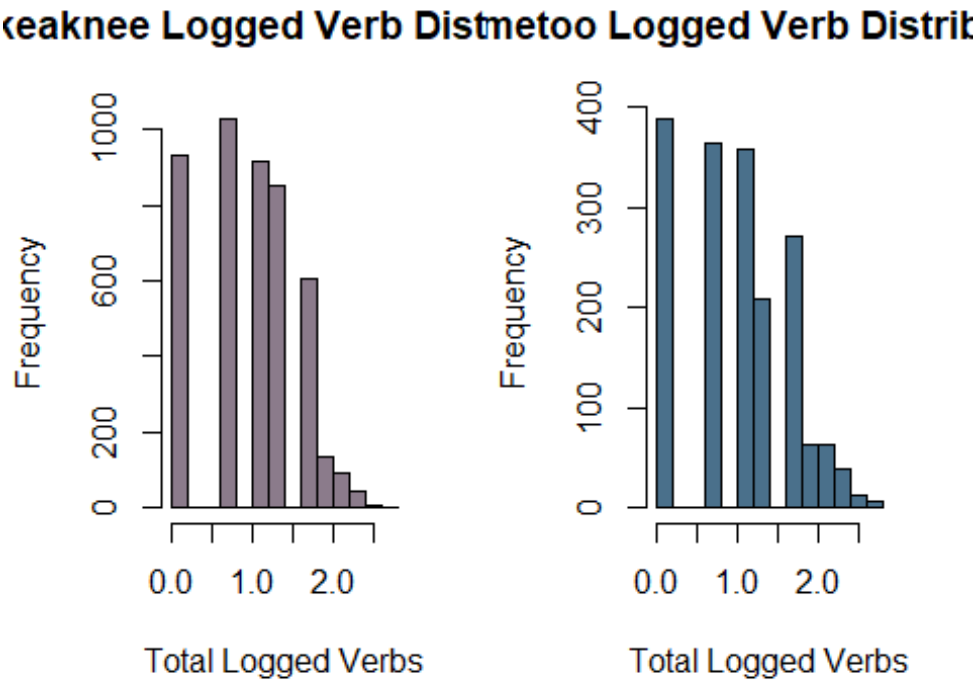



```
#Anything over 10 verbs seems like a lot for 140 characters. Let's do a
sanity check on those.

metoo[which(metoo$total_verbs > 10), ]
knee[which(knee$total_verbs > 10), ]
```

Step Three-A

The sanity check indicates that there really are legitimate tweets with a large number of verbs. I also visually inspected the zero verb tweets in Excel. I considered removing the zero verb tweets, but they are also a legitimate part of the dataset. So, I chose to leave both ends of the spectrum in the data.

```
#look at the distribution of logged verbs in each dataset. This is going to
be remarkably simiar between the split and not split, so just do it for the
not split datasets.
par(mfrow=c(1,2))

hist(log(knee$total_verbs), main="#takeaknee Logged Verb Distribution",
xlab="Total Logged Verbs", col='thistle4')
hist(log(metoo$total_verbs), main="#metoo Logged Verb Distribution",
xlab="Total Logged Verbs",col='skyblue4')
```



Step Four - Decision on Distribution Appropriateness

The data for verb count is clearly heavily skewed. While the sample size is large enough to use a t-test, the distribution would suggest using a Wilcox text instead. Proceed with a both a t-test and a Wilcoxon test. Hopefully, the results will match.

Step Five - Prepare for and Complete a T-Test

```
#These summaries are just helpful for my own understanding of the data.
print("Knee - not split")

## [1] "Knee - not split"

summary(knee$total_verbs)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   3.000   2.917   4.000  14.000

print("Knee - split")

## [1] "Knee - split"

summary(knee_split$total_verbs)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   2.000   3.000   3.581   4.000  14.000

print("Metoo - not split")
```

```
## [1] "Metoo - not split"
```

```
summary(metoo$total_verbs)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   3.000   3.039   4.000  15.000
```

```
print("Metoo - split")
```

```
## [1] "Metoo - split"
```

```
summary(metoo_split$total_verbs)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   3.000   2.978   4.000  19.000
```

```
#do a t test
t.test(knee$total_verbs, metoo$total_verbs, alternative="two.sided")
```

```
##
##  Welch Two Sample t-test
##
## data:  knee$total_verbs and metoo$total_verbs
## t = -1.9476, df = 3091.2, p-value = 0.05156
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2454274892  0.0008274892
## sample estimates:
## mean of x mean of y
##    2.9172    3.0395
```

```
#do a t test
t.test(knee_split$total_verbs, metoo_split$total_verbs,
alternative="two.sided")
```

```
##
##  Welch Two Sample t-test
##
## data:  knee_split$total_verbs and metoo_split$total_verbs
## t = 13.571, df = 9543.2, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.5159015 0.6900985
## sample estimates:
## mean of x mean of y
##    3.5808    2.9778
```

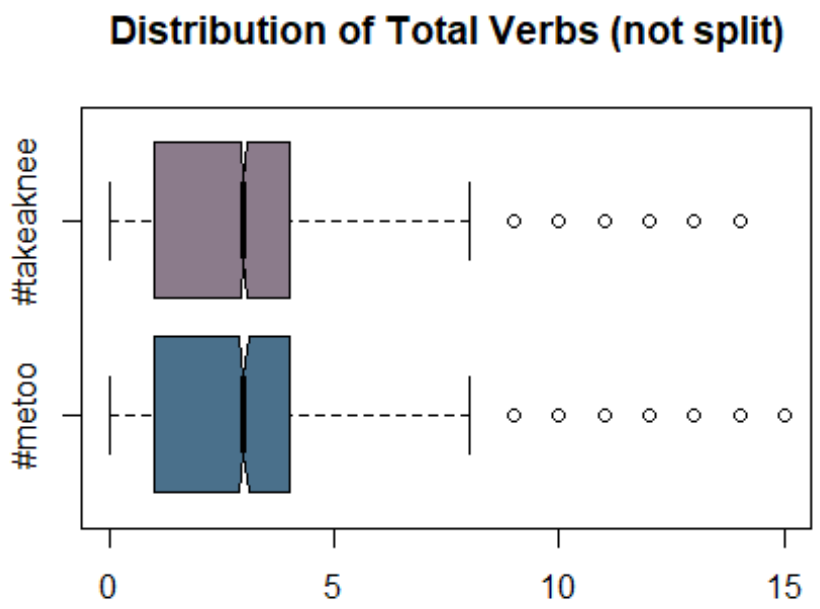
Step Five Results: T-Test Results

The null hypothesis is that the mean use of verbs in the metoo tweets equals the mean use of verbs in the takeaknee tweets. We have contradictory results between the two tests. At a significance level of .01, we do not have evidence of varying significantly when the hashtags

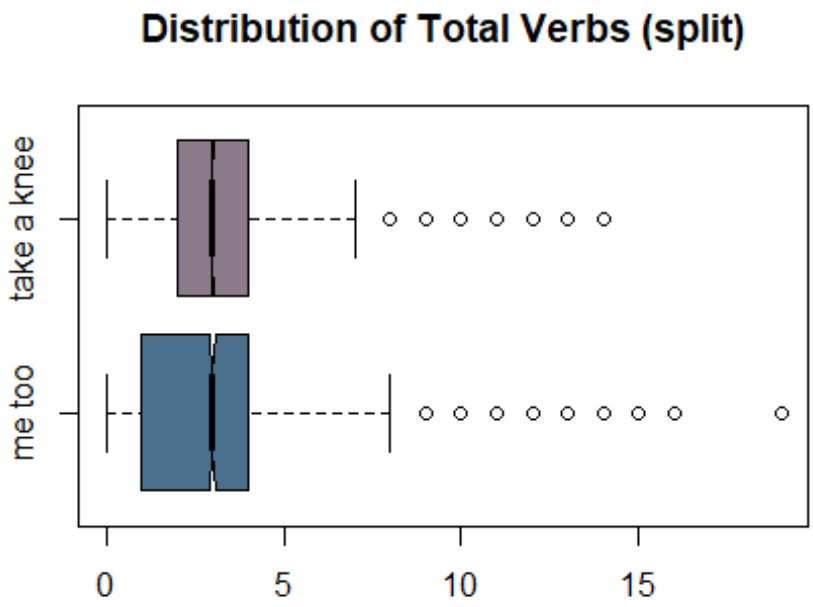
are treated as hashtags, but we do have significant evidence of varying when the hashtags are treated as individual words.

Step Six: Complete a Wilcoxon Test

```
#do a wilcox test
#https://www.stat.auckland.ac.nz/~wild/ChanceEnc/Ch10.wilcoxon.pdf
wilcox.test(knee$total_verbs, metoo$total_verbs, alternative="two.sided")

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  knee$total_verbs and metoo$total_verbs
## W = 5079600, p-value = 0.2913
## alternative hypothesis: true location shift is not equal to 0

#do a wilcox test
#https://www.stat.auckland.ac.nz/~wild/ChanceEnc/Ch10.wilcoxon.pdf
wilcox.test(knee_split$total_verbs, metoo_split$total_verbs,
alternative="two.sided")

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  knee_split$total_verbs and metoo_split$total_verbs
## W = 15206000, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0

#if the slopes of notched boxplots overlap, there is not a significant
difference in the median. Let's view that.

#set up a color palette
myColors = c('skyblue4','thistle4')


boxplot(metoo$total_verbs, knee$total_verbs,  notch=TRUE,
  names=c("#metoo","#takeaknee"),
  col=myColors,
  main="Distribution of Total Verbs (not split)",
  horizontal = TRUE)
```




```
boxplot(metoo_split$total_verbs, knee_split$total_verbs,  notch=TRUE,
  names=c("me too","take a knee"),
  col=myColors,
  main="Distribution of Total Verbs (split)",
  horizontal = TRUE)
```



Step Six Results - Results of Wilcoxon Test

The null hypothesis is that the location shift of verbs in each hashtag is equal to zero. The alternative hypothesis is that the location shift of verbs is not equal to zero. At the significance level of .01, we see similar results as the t-test. When the hashtags are treated as hashtags, there is not enough evidence to assert that the distribution varies significantly. When the hashtags are treated as individual words, there is enough evidence to assert that the distributions are of different shapes. In both instances, the median is the same (3), indicating that the difference lies truly in the shape of the distribution, and not in the measure of central tendency.

Step Seven - Review Verb Tense

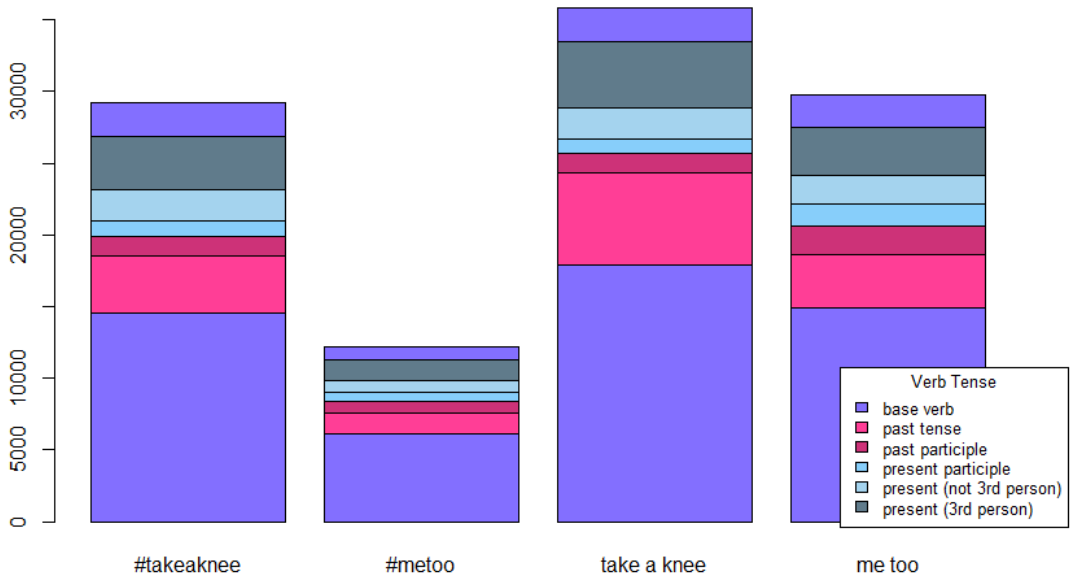
This section could be interesting as a mechanism for further analysis.

```
verb_cols = c("total_verbs", "base_verb", "past_tense",  "past_participle",
"present_participle", "present_not_third", "present_third")

metoo_sums = apply(metoo[verb_cols],2, sum)
knee_sums = apply(knee[verb_cols],2, sum)
metoo_split_sums = apply(metoo_split[verb_cols],2, sum)
knee_split_sums = apply(knee_split[verb_cols],2, sum)

combined_verbs = cbind(knee_sums, metoo_sums, knee_split_sums,
metoo_split_sums)
```

```r
colnames(combined_verbs) = c("#takeaknee", "#metoo", "take a knee", "me too")
combined_verbs
```

```
##                    #takeaknee #metoo take a knee me too
## total_verbs             14586   6079       17904  14889
## base_verb                3981   1454        6411   3696
## past_tense               1367    850        1336   2014
## past_participle          1054    670        1032   1577
## present_participle       2131    807        2131   2013
## present_not_third        3787   1407        4702   3299
## present_third            2266    891        2292   2290
```

```r
#https://stats.stackexchange.com/questions/14118/drawing-multiple-barplots-on-a-graph-in-r
#https://stackoverflow.com/questions/12481430/how-to-display-the-frequency-at-the-top-of-each-factor-in-a-barplot-in-r

#set up a color palette
stackedColors = c('slateblue1', 'violetred1', 'violetred3', 'lightskyblue', 'lightskyblue2', 'lightskyblue4')


#get the matrix of combined verbs for takeaknee and metoo
combined_verbs_minus = as.matrix(combined_verbs)


barplot(as.matrix(combined_verbs_minus), col=stackedColors, main="Total Verb Usage by Verb Tense",  bty='L')

#getting the legend off the plot:
#https://stackoverflow.com/questions/3932038/plot-a-legend-outside-of-the-plotting-area-in-base-graphics
par(xpd=NA)

legend("bottomright", c("base verb", "past tense",  "past participle", "present participle", "present (not 3rd person)", "present (3rd person)"), cex=.8, pt.cex=.8, fill=stackedColors, inset=c(-0.05,0), title="Verb Tense" )
```


Total Verb Usage by Verb Tense
0
5000
10000
20000
30000
#takeaknee
#metoo
take a knee
me too
Verb Tense
base verb
past tense
past participle
present participle
present (not 3rd person)
present (3rd person)