Team MarshMallow

STOR 565

9 March 2022

Project Proposal

1. Team Members

- Kyungjin Sohn
- Soohyun Kim
- Winnie Ren
- Yang Xiao
- Ziang Li

2. Data Source and Brief Description

Our dataset on real versus fake news comes from Kaggle. There, we found two files– one containing the data for real news articles and the other containing data for fake news articles. Each article in the dataset was run through a fact-checking website called PolitiFact, which determined the accuracy of the claims made in the articles using its Truth-O-Meter.

Our data contains 23,481 fake and 21,417 real news with four columns each: title, main text, category, and the date. Regarding the subject of the observation, Middle-east, News, politics, and US News. For the real news there are 2 subject categories; politics and the dataset for fake news contains the following 6 subject categories: Government News, left-news,worlds.

3. Motivation and Goals

The reason why we selected our dataset was because we wanted to perform natural language processing on a topic that was relevant to our everyday lives. At some point of our lives, we have all come across a news article that is fake. However, whether we realize that the article is fake or not, that is the question. Fake news, defined as news that is intentionally crafted to convey misleading information or totally fabricated information and presented in a way that mimics conventional news, propagates the spread of misinformation and disinformation.

Some instances of fake news include:

- The start of the Covid-19 pandemic was accompanied by the circulation of misinformation, myths, and conspiracy theories about the disease.
- The circulation of fake news during periods of great political activity, such as elections, that deliberately target political leaders.

We want to learn what aspects of textual data are used to determine whether a news article is truthful or not. Therefore, our goal is to build a fake news detection model that will be able to recognize whether a particular article is real or fake by using machine learning techniques. We will explore several potential classification methods on the textual data that we have of the real and fake news articles.

4. Exploratory Data Analysis

4.1 Data Pre-Processing and Cleaning

Prior to doing analysis on our data, we had to first inspect the data. We checked to make sure that there were no NA values present in our data that we had to eliminate. When inspecting the data, we made note that the categories of the subject variable were different for the real and fake datasets, therefore, we cannot make any inference on the type of article based on the subject.

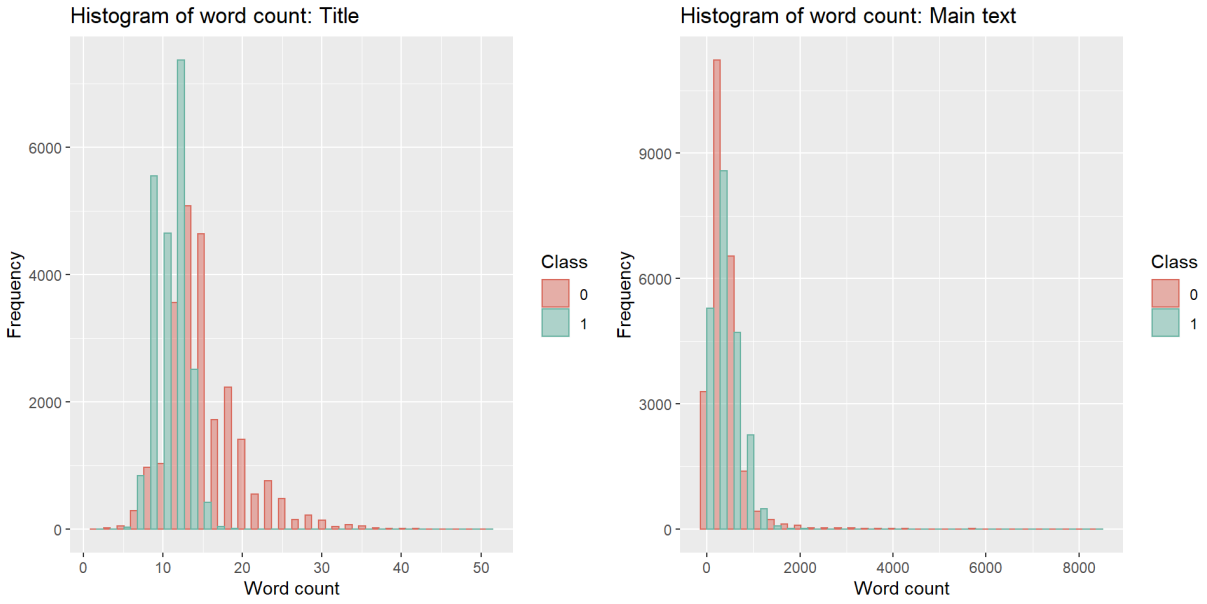
To perform EDA, it was necessary for us to clean the dataset. When inspecting the data, we found that in the main text of the real dataset, the majority of the observations start with the line: “Location (Reuters) -”. We found that this would interfere with our classification model, since these words could signify that the news is real. Therefore, we removed it from the main text. We combined the real and fake news by creating a binary variable called `Class`, where 0 indicates a fake news observation and 1 indicates a real news observation. In addition, from the title and the main text we removed all the special characters like “[][!#$%()*,.:;<=>@^_|~.{}]” and kept only the alphanumeric characters.

4.2 EDA of Fake and True News Text

4.2-1 Word Count Distribution of Fake and True News

Next, we look at the word count distribution of fake and true news. Looking at Figure 1, we find that the spread of fake news is wider compared to the true news. As for the word count distribution for the main text, we find that fake news and real news share similar distributions for word count.

Figure 1. Histogram of Title and Main Text Word Count for Fake and Real News



4.2-2 Top 20 of most used word for Fake and True News

As much as the length of a news article is an important factor to consider, we should also consider the actual content of the news itself. In order to look at the content being discussed, we looked at the most frequently used words of fake and true news in both the title and the main text. We first cleaned the text removing numbers, punctuations, white space, and special characters. Based on this we created a table containing the most frequently used words and their frequency. With the table we created histograms of the keywords.

Looking at Figure 2 we can compare the fake and the real news titles. We find that for both "trump" is the most commonly used word. It is important to note that "video" is a word that is only frequently used in fake news. Also, the word "just" only appears in fake news. This shows that fake news tends to use more casual words. Also, the word "breaking" seems to point to the fact that fake news is more likely to point to urgency. With real news they tend to have words like "house", "senate", and "republican". Yet, this may have to do with the category of the news. Where our sample of true news includes more domestic politics compared to fake news. We would need to investigate this issue further.

Figure 2. Histogram of Most Frequently Used Words in Title for Fake and Real News

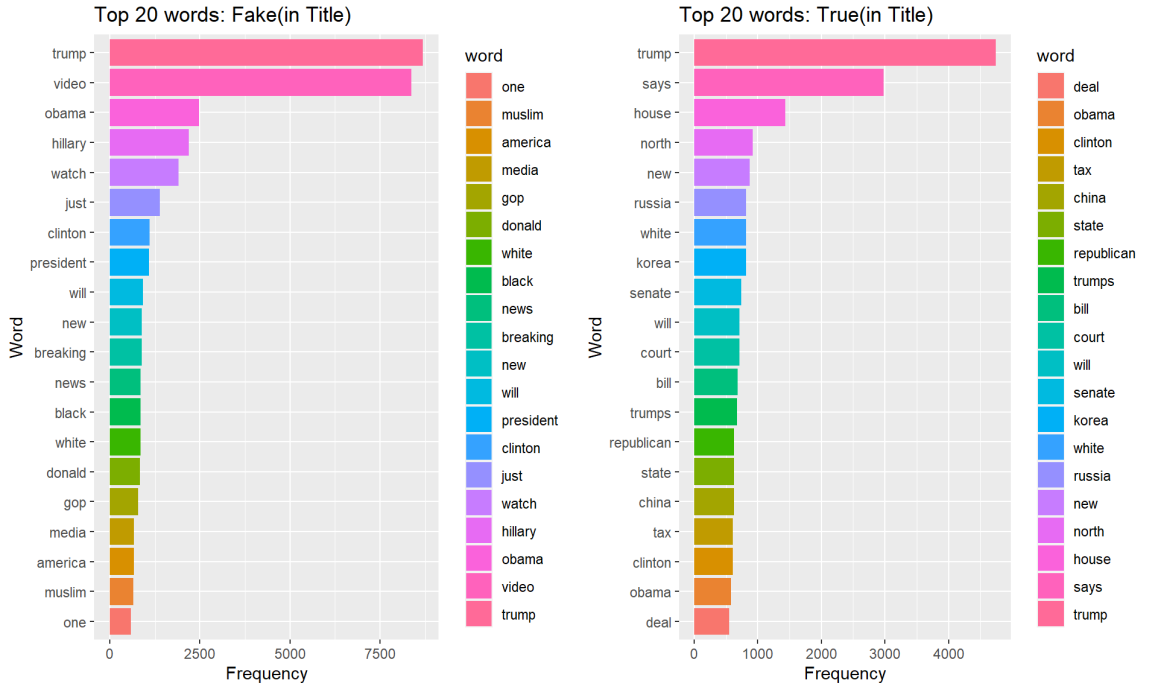
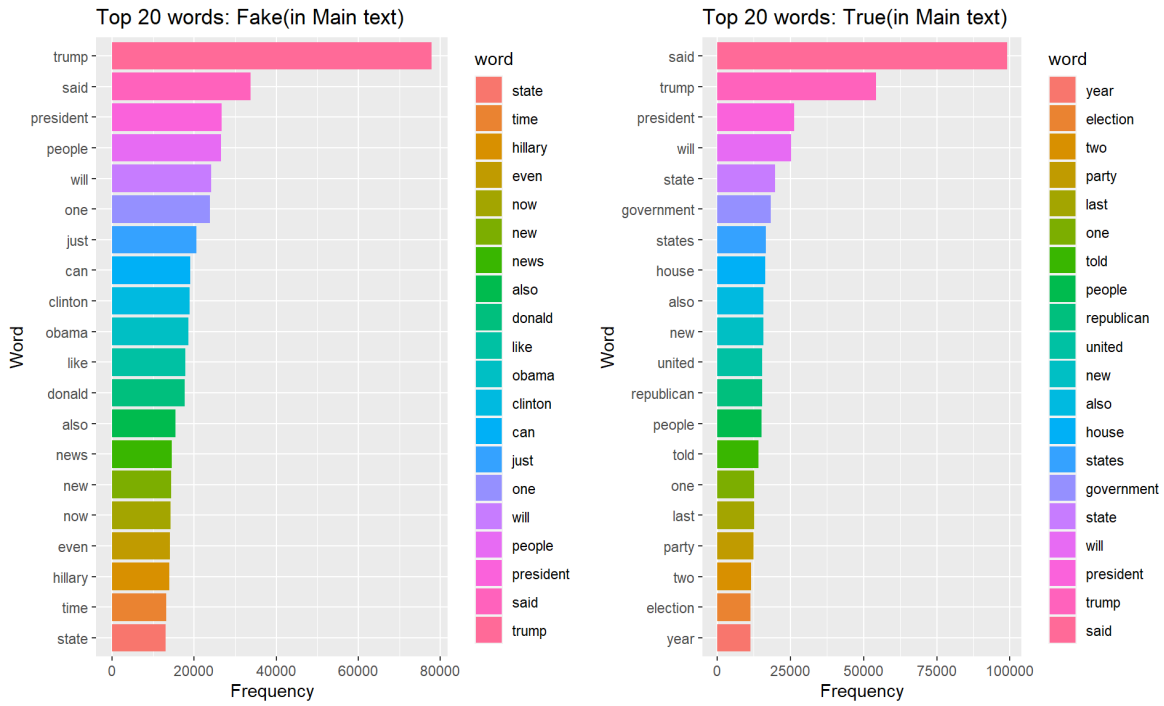


Figure 3. Histogram of Most Frequently Used Words in Main Text for Fake and Real News



With the main text of the news, given in Figure 3, similarly to the title we find that fake news tends to use more casual words such as “just”, “like”, and “even”. It is also interesting to note the difference in verb usage. For fake news, the word “can” is frequently used, while for real news “told” appears more often. Based on this we can speculate that fake news are more likely to make assumptions and hint to possibilities in their news content. However, with real news, they are more likely to quote and state facts based on another person, which we would assume to be a professional in the field. Again, we see that fake news also hints to urgency with words like “now” and “time”. For real news, the words “party” and “election” appear frequently. As stated above, this may have to do with the news categories.

4.3 EDA of Linguistic Inquiry and Word Count (LIWC) Data

4.3-1 Pre-Processing LIWC Result

Table 1. Description of Select LIWC Variables

Variable	Measurement	Variable	Measurement	Variable	Measurement
WC	Word count of title and text combined	conj	Conjunctions	bio	Biological processes (eat, blood, pain)
Tone	Sentiment of the text ranging between 0 to 100. 50 is neutral	negate	Negations (no, not, never)	drives	Drives (affiliation, achievement, power, reward)
WPS	Words per sentence	verb	Verb	relativ	Relativity (motion, space, time)
Sixltr	Words with more than six letters	adj	Common adjectives	work	Work (jobs, majors, xeros)
function	Total function words (it, to, no, very)	interrog	Interrogatives (how, when, what)	leisure	Leisure (cook, chat, movie)
ppron	Personal pronouns	quant	Quantifiers (few, many, much)	home	Home (kitchen, landlord)
article	article	affect	Affective processes (happy, cried)	money	Money (audit, cash, owe)
prep	prepositions	social	Social Process (mate, talk, they)	relig	Religion (altar, church)
auxverb	Auxiliary verbs	cogproc	Cognitive processes (cause, know, ought)	death	Death (bury, coffin, kill)
adverb	Common Adverbs	percept	Perceptual processes (look, heard, feeling)	informal	Informal language. Netspeak, Assent, Swear words

To better understand the characteristics of the text, we used a lexicon based method with the Linguistic Inquiry and Word Count(LIWC). Using the LIWC we were able to extract 95 variables. Detailed information about the variables can be found here. Among 95 variables, as some of them would overlap with each other we chose to look at 30 top summary variables. Except for `WC`, `Tone`, and `WPS`, the values show the percentage of words that fall into the category. Detailed information about 30 variables are given in Table 1.

4.3-2 Correlations between LIWC variables

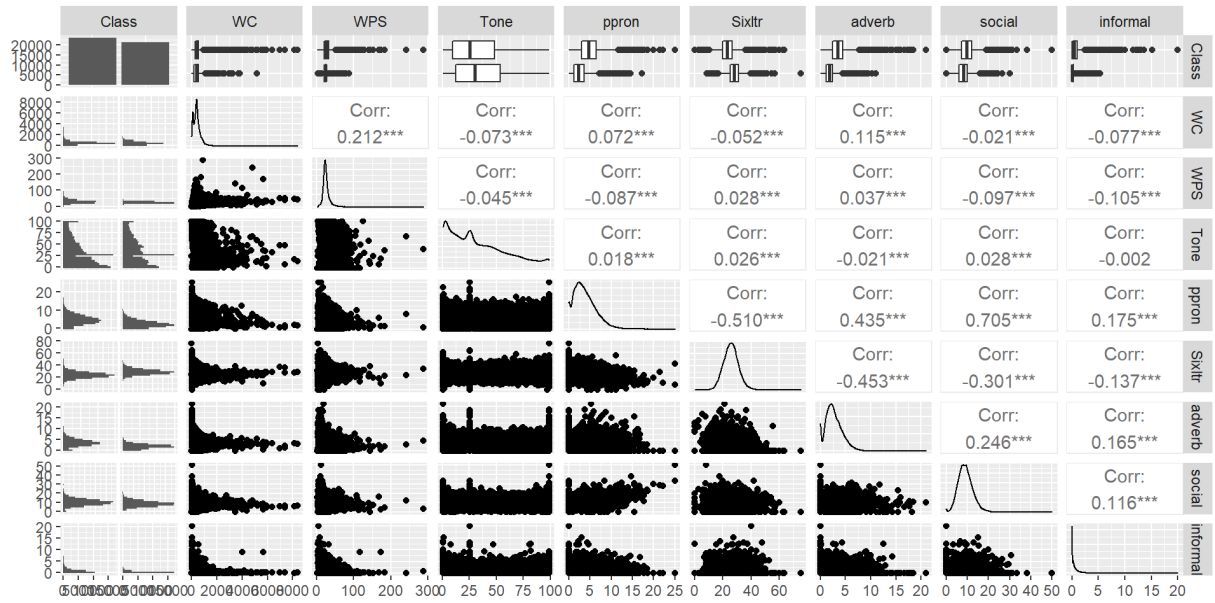
With the chosen 30 variables, we look at the correlation. Table 2 is the correlation table. Among the correlation we list the variables with the top 20 correlation. We find that `auxverb` and `verb` have the highest correlation of 0.726. This is a reasonable finding as `auxver` is the count of auxiliary verbs and `verb` is the number of verbs. Therefore, `auxverb` would be included in `verb`. The next is `ppron` and `social` with a correlation of 7.05. It shows that news articles with higher usage of personal pronouns have a higher number of social words. `function` also has a high correlation with `auxverb`, `ppron`, `adverb`, `verb`, `Sixtlr`, `cogproc`, and `conj`. With the `Class` variable, which indicates the fake and real news has a high correlation `adverb` of 0.491, `Sixltr` of 0.458, and `ppron` of 0.435. Based on this we find that real news has a higher use of adverbs, words with more than six letters, and personal pronouns compared to fake news.

Table 2. Top 20 Correlation LIWC Variables

Variable 1	Variable 2	Correlation	Variable 1	Variable 2	Correlation
auxverb	verb	0.726	function.	conj	0.466
ppron	social	0.705	Class	Sixltr	0.458
function.	auxverb	0.584	Sixltr	adverb	0.453
function.	ppron	0.56	percept	leisure	0.452
function.	adverb	0.533	work	money	0.45
function.	verb	0.523	verb	cogproc	0.44
Sixltr	ppron	0.51	ppron	adverb	0.435
Sixltr	function.	0.5	Class	ppron	0.435
Class	adverb	0.491	ppron	verb	0.43
function.	cogproc	0.478	adverb	interrog	0.426

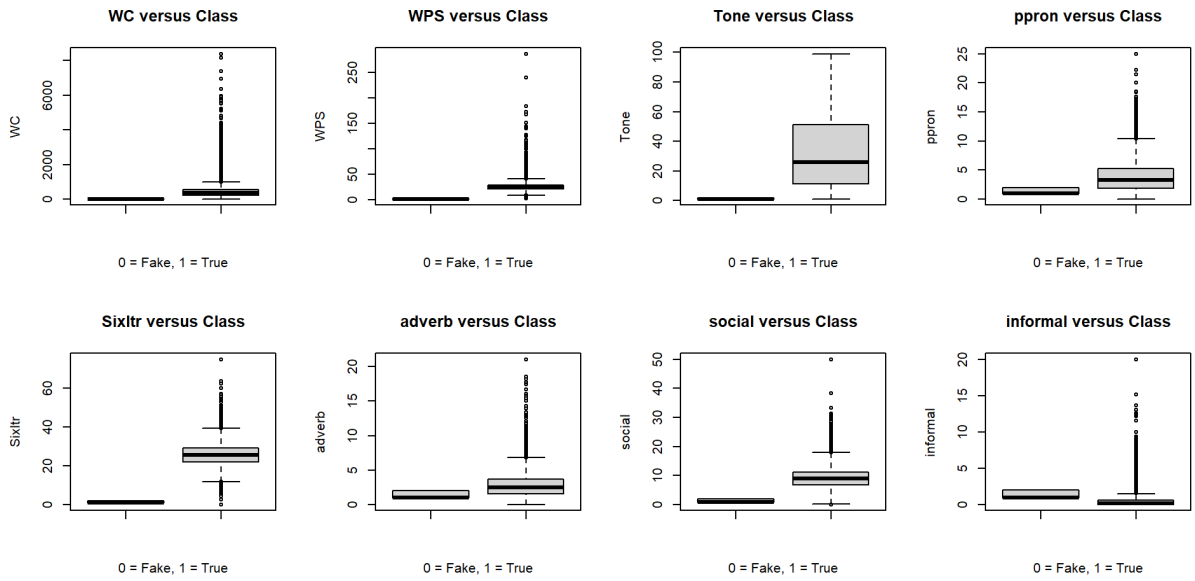
4.3-3 Visualizing Relationship among LIWC variables

Figure 4. Correlation Table of Select LIWC Variables



Based on the correlation and common sense, we chose to visualize the relationship for ‘Class’, `WC`, `WPS`, `Tone`, `ppron`, `Sixltr`, `adverb`, `social`, and `informal`. From Figure 4, we see that all the correlations are significant. To take a better look at the difference in fake and real news for the variables selected we draw a boxplot for each one in Figure 5. First looking and `WC`, the word count, we see that there is a higher variation in length for real news compared to fake news. Also, real news tends to be longer. This is in line with what we have found above. Second, looking at `WPS`, words per sentence, the length of a sentence is longer for real news. Meaning that fake news mostly consists of short sentences. With `Tone`, we can clearly see that fake news usually contains information about negative news, while for real news it is well-rounded. For `ppron`, real news uses more personal pronouns. Looking at `Sixltr`, we find that fake news has a higher proportion of shorter words, while real news has longer words. Similar findings are reported for `adverb` and `social` as well. Finally, fake news consists of more `informal` words.

Figure 5. Boxplot of Select LIWC Variables



4.3-4 Testing means between two different groups

Table 3. Testing Difference of Means of Fake and Real News Variables

Variable	T-Statistic	Degrees of Freedom	P-value	95% CI	Sample Estimate of Fake	Sample Estimate of Real
WC	15.727	40917	< 2.2e-16	[45.913,58.986]	451.0233	398.5744
WPS	53.82	29425	< 2.2e-16	[4.492,4.831]	28.20337	23.54204
Tone	-13.638	44535	< 2.2e-16	[-3.941,-2.951]	32.23441	35.68022
ppron	103.99	42262	< 2.2e-16	[-3.941,-2.951]	4.805203	2.575414
social	46.795	44635	< 2.2e-16	[1.431,1.556]	9.828172	8.334765
adverb	122.02	39679	< 2.2e-16	[1.650, 1.703]	3.495365	1.818927
Sixltr	-109.77	44890	< 2.2e-16	[-5.087,-4.908]	23.36330	28.36098
informal	69.832	30397	< 2.2e-16	[0.553, 0.585]	0.7851523	0.2159682

In the previous section we have visually checked for the differences of means in the variables based on the `Class` (Fake = 0, Real = 1) of the news. To substantiate our findings on a statistical level we run t-tests on the 8 variables. The results are given in Table 3. For all 8, as the p-values are small, we can conclude that the difference is statistically significant, allowing us to accept the alternative hypothesis that the true difference in means is not equal to 0. Based on our findings, we would be able to implement the differences into our classification model.

5. Potential Questions and Ideas

As we have highlighted the importance of being able to sort out fake news we aim to answer the question: “How do we classify fake news?”. In a realistic setting of a person that is exposed to fake news, they only have the title and the text of the news. Therefore, to mimic this setting we would like to build a classification model that can accurately distinguish fake and real news using the title and the main text of the news.

In our EDA, we implemented a lexicon based method (LIWC) to get a basic understanding of the difference in fake and real news. For our final project, we would like to expand on our findings from the lexicon based method and incorporate various NLP techniques like n-grams to extract important features that may help us to distinguish fake and real news. Based on the extracted features using the NLP techniques, as we do not primarily know the structure of the data, we are not able to choose one model that would be able to clearly distinguish fake and real news. We will compare the results from KNN, LDA, Logistic Regression, Naive Bayes, and Quadratic Classifier to find the best performing method.

Project Proposal (Code for EDA)

Marshmallow

2022 3 9

1. Data Pre-processing and Cleaning

1-1. Data Pre-processing

```
# 1. Read Data
df.fake <- read.csv('../data/Fake.csv', encoding = 'UTF-8',
                    header = TRUE)
df.true <- read.csv('../data/True.csv', encoding = 'UTF-8',
                    header = TRUE)

# 2. Look through the data
          # dimension
dim(df.fake)
```

```
## [1] 23481     4
```

```
dim(df.true)
```

```
## [1] 21417     4
```

```
          # check NA values
table(is.na(df.fake))
```

```
##
## FALSE
## 93924
```

```
table(is.na(df.true))
```

```
##
## FALSE
## 85668
```

```
          # simple view of the data
colnames(df.fake)
```

```
## [1] "title"   "text"    "subject" "date"
```

```
colnames(df.true)
```

```
## [1] "title"   "text"    "subject" "date"
```

```
head(df.fake, 1)
```

```
## title
## 1 Donald Trump Sends Out Embarrassing New Year's Eve Message; This is Disturbing
## text
## 1 Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that. Instead, he had to give a shout out to his enemies, haters and the very dishonest fake news media.  The former reality show star had just one job to do and he couldn t do it. As our Country rapidly grows stronger and smarter, I want to wish all of my friends, supporters, enemies, haters, and even the very dishonest Fake News Media, a Happy and Healthy New Year, President Angry Pants tweeted.  2018 will be a great year for America! As our Country rapidly grows stronger and smarter, I want to wish all of my friends, supporters, enemies, haters, and even the very dishonest Fake News Media, a Happy and Healthy New Year. 2018 will be a great year for America!  Donald J. Trump (@realDonaldTrump) December 31, 2017Trump s tweet went down about as welll as you d expect.What kind of president sends a New Year s greeting like this despicable, petty, infantile gibberish? Only Trump! His lack of decency won t even allow him to rise above the gutter long enough to wish the American citizens a happy new year!  Bishop Talbert Swan (@TalbertSwan) December 31, 2017no one likes you Calvin (@calvinstowell) December 31, 2017Your impeachment would make 2018 a great year for America, but I ll also accept regaining control of Congress.  Miranda Yaver (@mirandayaver) December 31, 2017Do you hear yourself talk? When you have to include that many people that hate you you have to wonder? Why do the they all hate me?  Alan Sandoval (@AlanSandoval13) December 31, 2017Who uses the word Haters in a New Years wish??  Marlene (@marlene399) December 31, 2017You can t just say happy new year?  Koren pollitt (@Korencarpenter) December 31, 2017Here s Trump s New Year s Eve tweet from 2016.Happy New Year to all, including to my many enemies and those who have fought me and lost so badly they just don t know what to do. Love!  Donald J. Trump (@realDonaldTrump) December 31, 2016This is nothing new for Trump. He s been doing this for years.Trump has directed messages to his enemies and haters for New Year s, Easter, Thanksgiving, and the anniversary of 9/11. pic.twitter.com/4FPAe2KypA Daniel Dale (@ddale8) December 31, 2017Trump s holiday tweets are clearly not presidential.How long did he work at Hallmark before becoming President? Steven Goodine (@SGoodine) December 31, 2017He s always been like this . . . the only difference is that in the last few years, his filter has been breaking down.  Roy Schulze (@thbthttt) December 31, 2017Who, apart from a teenager uses the term haters?  Wendy (@WendyWhistles) December 31, 2017he s a fucking 5 year old Who Knows (@rainyday80) December 31, 2017So, to all the people who voted for this a hole thinking he would change once he got into power, you were wrong! 70-year-old men don t change and now he s a year older.Photo by Andrew Burton/Getty Images.
## subject date
## 1 News December 31, 2017
```

```
head(df.true, 1)
```

```
## title
```

```
## 1 As U.S. budget fight looms, Republicans flip their fiscal script
## text
## 1 WASHINGTON (Reuters) - The head of a conservative Republican faction in the U.S.
Congress, who voted this month for a huge expansion of the national debt to pay for tax
cuts, called himself a "fiscal conservative" on Sunday and urged budget restraint in
2018. In keeping with a sharp pivot under way among Republicans, U.S. Representative Mark
Meadows, speaking on CBS' "Face the Nation," drew a hard line on federal spending,
which lawmakers are bracing to do battle over in January. When they return from the
holidays on Wednesday, lawmakers will begin trying to pass a federal budget in a fight
likely to be linked to other issues, such as immigration policy, even as the November
congressional election campaigns approach in which Republicans will seek to keep control
of Congress. President Donald Trump and his Republicans want a big budget increase in
military spending, while Democrats also want proportional increases for non-defense
"discretionary" spending on programs that support education, scientific research,
infrastructure, public health and environmental protection. "The (Trump) administration
has already been willing to say: 'We're going to increase non-defense discretionary
spending ... by about 7 percent,'" Meadows, chairman of the small but influential House
Freedom Caucus, said on the program. "Now, Democrats are saying that's not enough, we
need to give the government a pay raise of 10 to 11 percent. For a fiscal conservative, I
don't see where the rationale is. ... Eventually you run out of other people's money,"
he said. Meadows was among Republicans who voted in late December for their party's
debt-financed tax overhaul, which is expected to balloon the federal budget deficit and
add about $1.5 trillion over 10 years to the $20 trillion national debt. "It's
interesting to hear Mark talk about fiscal responsibility," Democratic U.S.
Representative Joseph Crowley said on CBS. Crowley said the Republican tax bill would
require the United States to borrow $1.5 trillion, to be paid off by future generations,
to finance tax cuts for corporations and the rich. "This is one of the least ...
fiscally responsible bills we've ever seen passed in the history of the House of
Representatives. I think we're going to be paying for this for many, many years to
come," Crowley said. Republicans insist the tax package, the biggest U.S. tax overhaul
in more than 30 years, will boost the economy and job growth. House Speaker Paul Ryan,
who also supported the tax bill, recently went further than Meadows, making clear in a
radio interview that welfare or "entitlement reform," as the party often calls it,
would be a top Republican priority in 2018. In Republican parlance, "entitlement"
programs mean food stamps, housing assistance, Medicare and Medicaid health insurance for
the elderly, poor and disabled, as well as other programs created by Washington to assist
the needy. Democrats seized on Ryan's early December remarks, saying they showed
Republicans would try to pay for their tax overhaul by seeking spending cuts for social
programs. But the goals of House Republicans may have to take a back seat to the Senate,
where the votes of some Democrats will be needed to approve a budget and prevent a
government shutdown. Democrats will use their leverage in the Senate, which Republicans
narrowly control, to defend both discretionary non-defense programs and social spending,
while tackling the issue of the "Dreamers," people brought illegally to the country as
children. Trump in September put a March 2018 expiration date on the Deferred Action for
Childhood Arrivals, or DACA, program, which protects the young immigrants from
deportation and provides them with work permits. The president has said in recent Twitter
messages he wants funding for his proposed Mexican border wall and other immigration law
changes in exchange for agreeing to help the Dreamers. Representative Debbie Dingell told
CBS she did not favor linking that issue to other policy objectives, such as wall
funding. "We need to do DACA clean," she said.  On Wednesday, Trump aides will meet
with congressional leaders to discuss those issues. That will be followed by a weekend of
strategy sessions for Trump and Republican leaders on Jan. 6 and 7, the White House said.
Trump was also scheduled to meet on Sunday with Florida Republican Governor Rick Scott,
who wants more emergency aid. The House has passed an $81 billion aid package after
```

hurricanes in Florida, Texas and Puerto Rico, and wildfires in California. The package far exceeded the $44 billion requested by the Trump administration. The Senate has not yet voted on the aid.
subject date
1 politicsNews December 31, 2017

```
          # category table of the data
table(df.fake$subject)
```

```
##
## Government News       left-news     Middle-east            News        politics
##            1570            4459             778            9050            6841
##         US_News
##             783
```

```
table(df.true$subject)
```

```
##
## politicsNews    worldnews
##        11272        10145
```

1-2. Data Cleaning

```
# 1. Remove redundant strings
          # Average text containing "(Reuters)" in True v. Fake Dataset
mean(grepl("(Reuters)", df.fake$text, fixed = TRUE))
```

```
## [1] 0.0003832886
```

```
mean(grepl("(Reuters)", df.true$text, fixed = TRUE))
```

```
## [1] 0.9920624
```

```
          # Drop Prefix (Reuters) Function
drop_prefix <- function(text, prefix = '(Reuters)', n = 5) {
          # splits textual data into separate words + symbols
          ts = strsplit(text, " ")
          ifelse(prefix %in% ts[[1]][1:5],
                 strsplit(text,"(Reuters) - ", fixed = TRUE)[[1]][-1], text)
}
new_Text <- apply(df.true["text"], 1, drop_prefix)
new_Text <- as.data.frame(new_Text)
df.true["text"] <- new_Text
          # result
df.true$text[1]
```

[1] "The head of a conservative Republican faction in the U.S. Congress, who voted this month for a huge expansion of the national debt to pay for tax cuts, called himself a "fiscal conservative" on Sunday and urged budget restraint in 2018. In keeping with a sharp pivot under way among Republicans, U.S. Representative Mark Meadows, speaking on CBS' "Face the Nation," drew a hard line on federal spending, which lawmakers are bracing to do battle over in January. When they return from the holidays on Wednesday, lawmakers will begin trying to pass a federal budget in a fight likely to be linked to other issues, such as immigration policy, even as the November congressional election campaigns approach in which Republicans will seek to keep control of Congress. President Donald Trump and his Republicans want a big budget increase in military spending, while Democrats also want proportional increases for non-defense "discretionary" spending on programs that support education, scientific research, infrastructure, public health and environmental protection. "The (Trump) administration has already been willing to say: 'We're going to increase non-defense discretionary spending ... by about 7 percent,'" Meadows, chairman of the small but influential House Freedom Caucus, said on the program. "Now, Democrats are saying that's not enough, we need to give the government a pay raise of 10 to 11 percent. For a fiscal conservative, I don't see where the rationale is. ... Eventually you run out of other people's money," he said. Meadows was among Republicans who voted in late December for their party's debt-financed tax overhaul, which is expected to balloon the federal budget deficit and add about $1.5 trillion over 10 years to the $20 trillion national debt. "It's interesting to hear Mark talk about fiscal responsibility," Democratic U.S. Representative Joseph Crowley said on CBS. Crowley said the Republican tax bill would require the United States to borrow $1.5 trillion, to be paid off by future generations, to finance tax cuts for corporations and the rich. "This is one of the least ... fiscally responsible bills we've ever seen passed in the history of the House of Representatives. I think we're going to be paying for this for many, many years to come," Crowley said. Republicans insist the tax package, the biggest U.S. tax overhaul in more than 30 years, will boost the economy and job growth. House Speaker Paul Ryan, who also supported the tax bill, recently went further than Meadows, making clear in a radio interview that welfare or "entitlement reform," as the party often calls it, would be a top Republican priority in 2018. In Republican parlance, "entitlement" programs mean food stamps, housing assistance, Medicare and Medicaid health insurance for the elderly, poor and disabled, as well as other programs created by Washington to assist the needy. Democrats seized on Ryan's early December remarks, saying they showed Republicans would try to pay for their tax overhaul by seeking spending cuts for social programs. But the goals of House Republicans may have to take a back seat to the Senate, where the votes of some Democrats will be needed to approve a budget and prevent a government shutdown. Democrats will use their leverage in the Senate, which Republicans narrowly control, to defend both discretionary non-defense programs and social spending, while tackling the issue of the "Dreamers," people brought illegally to the country as children. Trump in September put a March 2018 expiration date on the Deferred Action for Childhood Arrivals, or DACA, program, which protects the young immigrants from deportation and provides them with work permits. The president has said in recent Twitter messages he wants funding for his proposed Mexican border wall and other immigration law changes in exchange for agreeing to help the Dreamers. Representative Debbie Dingell told CBS she did not favor linking that issue to other policy objectives, such as wall funding. "We need to do DACA clean," she said. On Wednesday, Trump aides will meet with congressional leaders to discuss those issues. That will be followed by a weekend of strategy sessions for Trump and Republican leaders on Jan. 6 and 7, the White House said. Trump was also scheduled to meet on Sunday with Florida Republican Governor Rick Scott, who wants more emergency aid. The House has passed an $81 billion aid package after hurricanes in Florida, Texas and Puerto Rico, and wildfires in California. The package far exceeded the $44 billion requested by the Trump administration. The Senate has not yet voted on the aid. "

```
# 2. Combine Data
df.fake$Class <- 0
df.true$Class <- 1
df.full<- rbind(df.fake, df.true)
          # give column names which are missing
colnames(df.full) <- c("Title", "MainText", "Subject", "Date", "Class")
colnames(df.full)

# 3. Clean up Data
df.full$Title <- gsub("[][!#$%()*,.:;<=>@^_|~.{}]", " ", df.full$Title)
df.full$Title <- gsub("‘|’|"|"|-", " ", df.full$Title)

df.full$MainText <- gsub("[][!#$%()*,.:;<=>@^_|~.{}]", " ", df.full$MainText)
df.full$MainText <- gsub("‘|’|"|"|-", " ", df.full$MainText)
          # Save the result
write.csv(df.full, "../data/Full.csv", row.names = FALSE)
```

```
# 1. Read the saved data
df.full <- read.csv("../data/Full.csv", encoding = 'UTF-8',
                    header = TRUE)
df.full$Class <- as.factor(df.full$Class)

# 2. Check null data
table(is.na(df.full))
```

```
##
##  FALSE
## 224490
```

```
# 3. Split the data
df.split <- split(df.full, f = df.full$Class)
df.fake <- df.split[['0']]
df.true <- df.split[['1']]
```

2. Exploratory data analysis(EDA)

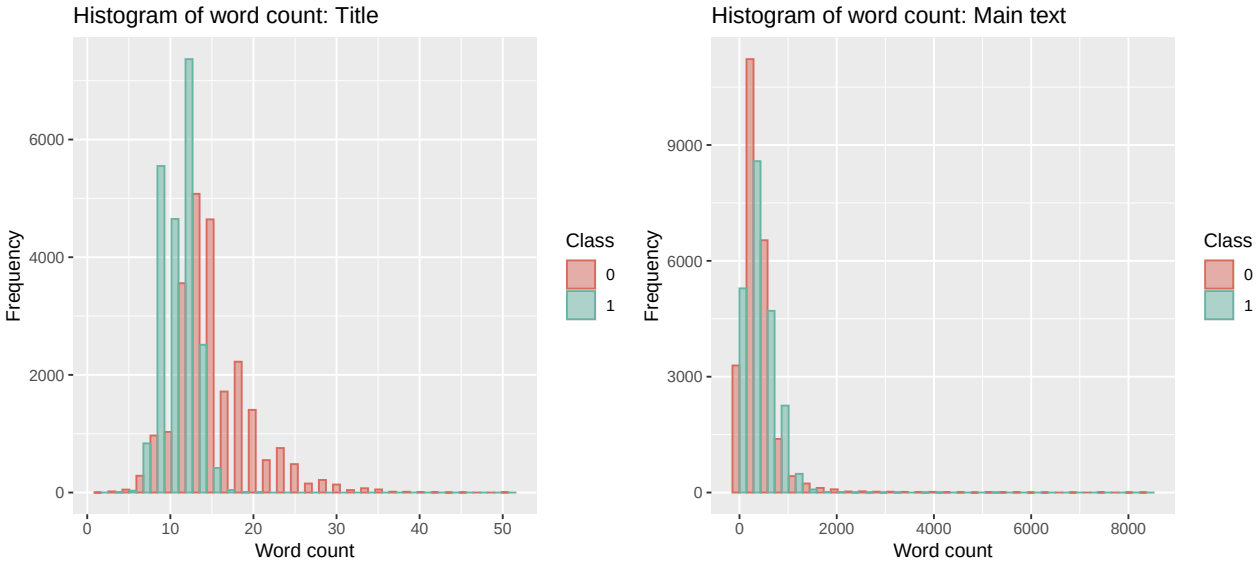
2-1. Word Count Distribution of Fake and True News

```
hist.title <- df.full %>%
                    mutate(WC.Title = str_count(Title, '\\w+')) %>%
                    ggplot(aes(x = WC.Title, fill = Class, color = Class)) +
                    geom_histogram(position = 'dodge', alpha=0.5) +
                    scale_color_manual(values=c("#d96a5d", "#69b3a2")) +
                    scale_fill_manual(values=c("#d96a5d", "#69b3a2")) +
                    labs(title = "Histogram of word count: Title",
                         x ="Word count", y = "Frequency") +
                    theme(legend.position = "right")

hist.main <- df.full %>%
```

```
                    mutate(WC.Main = str_count(MainText, '\\w+')) %>%
                    ggplot(aes(x = WC.Main, fill = Class, color = Class)) +
                    geom_histogram(position = 'dodge', alpha=0.5) +
                    scale_color_manual(values=c("#d96a5d", "#69b3a2")) +
                    scale_fill_manual(values=c("#d96a5d", "#69b3a2")) +
                    labs(title = "Histogram of word count: Main text",
                         x ="Word count", y = "Frequency") +
                    theme(legend.position = "right")

ggarrange(hist.title, hist.main,
          ncol = 2, nrow = 1)
```



2-2. Top 20 of most used word for Fake and True News

```
# 1. Data cleaning and calculate the word frequency
cleanup.text <- function(docs){
  toSpace <- content_transformer(function (x , pattern) gsub(pattern, " ", x))

  docs <- docs %>%
    tm_map(removeNumbers) %>%
    tm_map(removePunctuation) %>%
    tm_map(stripWhitespace) %>%
    tm_map(content_transformer(tolower)) %>%
    tm_map(toSpace, "/") %>%
    tm_map(toSpace, "@") %>%
    tm_map(toSpace, "\\|")
  docs <- tm_map(docs, removeWords, c("the"))
  docs <- tm_map(docs, removeWords, c("['’]s\b|[^[:alnum:][:blank:]@_]"))

  stopwords_regex <- paste(stopwords('en'), collapse = '\\b|\\b')
  stopwords_regex <- paste0('\\b', stopwords_regex, '\\b')
  docs <- str_replace_all(docs, stopwords_regex, '')
  docs <- Corpus(VectorSource(docs))
```

```
  doc_mat <- TermDocumentMatrix(docs)
  m <- as.matrix(doc_mat)
  v <- sort(rowSums(m), decreasing = TRUE)
  d_Rcran <- data.frame(word = names(v), freq = v)
  return(d_Rcran)
}

# 2. Save results of Title
          # Fake News
fake.title <- paste(df.fake$Title, collapse = " ")
docs <- Corpus(VectorSource(fake.title))
fake.title.freq <- cleanup.text(docs)
write.csv(fake.title.freq, "../data/Fake_wordFreq_Title.csv", row.names = FALSE)
          # True News
true.title <- paste(df.true$Title, collapse = " ")
docs <- Corpus(VectorSource(true.title))
true.title.freq <- cleanup.text(docs)
write.csv(true.title.freq, "../data/True_wordFreq_Title.csv", row.names = FALSE)

# 3. Save results of Main Text
          # Fake News
fake.MainText <- paste(df.fake$MainText, collapse = " ")
docs <- Corpus(VectorSource(fake.MainText))
fake.MainText.freq <- cleanup.text(docs)
write.csv(fake.MainText.freq, "../data/Fake_wordFreq_MainText.csv", row.names = FALSE)
          # True News
true.MainText <- paste(df.true$MainText, collapse = " ")
docs <- Corpus(VectorSource(true.MainText))
true.MainText.freq <- cleanup.text(docs)
write.csv(true.MainText.freq, "../data/True_wordFreq_MainText.csv", row.names = FALSE)
```

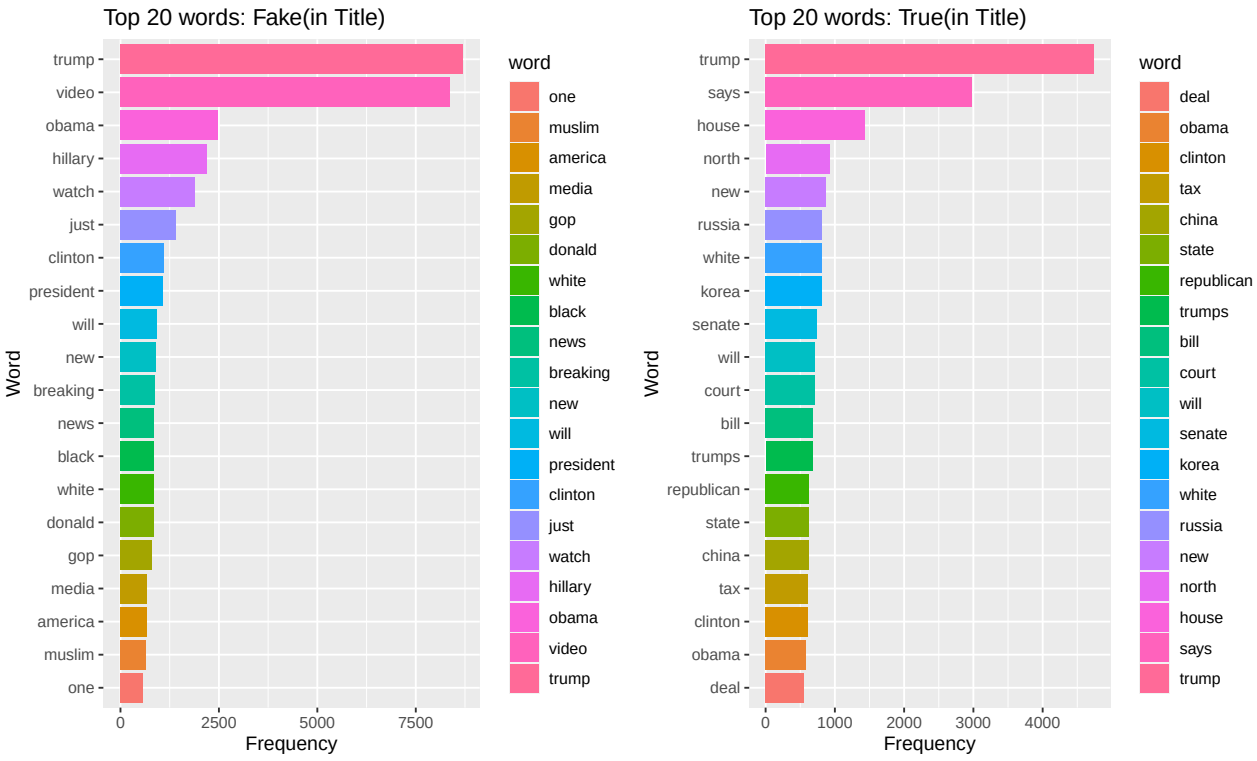
```
# 1. Read frequency data for title
df.fakeTitle.freq <- read.csv('../data/Fake_wordFreq_Title.csv', encoding = 'UTF-8',
                    header = TRUE)
df.trueTitle.freq <- read.csv('../data/True_wordFreq_Title.csv', encoding = 'UTF-8',
                    header = TRUE)

# 2. Draw bar plot of top 20 for title
myColors <- brewer.pal(11, "Spectral")
fake.bar <- top_n(df.fakeTitle.freq, 20, freq) %>%
                    arrange(freq) %>%
                    mutate(word = factor(word, levels = word)) %>%
                    ggplot(aes(x = word, y = freq, fill = word)) +
                    geom_bar(stat="identity") +
                    scale_colour_manual(name = "word", values = myColors) +
                    labs(title = "Top 20 words: Fake(in Title)",
                                  x ="Word", y = "Frequency") +
                    coord_flip()

true.bar <- top_n(df.trueTitle.freq, 20, freq) %>%
                    arrange(freq) %>%
                    mutate(word = factor(word, levels = word)) %>%
                    ggplot(aes(x = word, y = freq, fill = word)) +
```

```
                    geom_bar(stat="identity") +
                    scale_colour_manual(name = "word", values = myColors) +
                    labs(title = "Top 20 words: True(in Title)",
                         x ="Word", y = "Frequency") +
                    coord_flip()

ggarrange(fake.bar, true.bar,
          ncol = 2, nrow = 1)
```

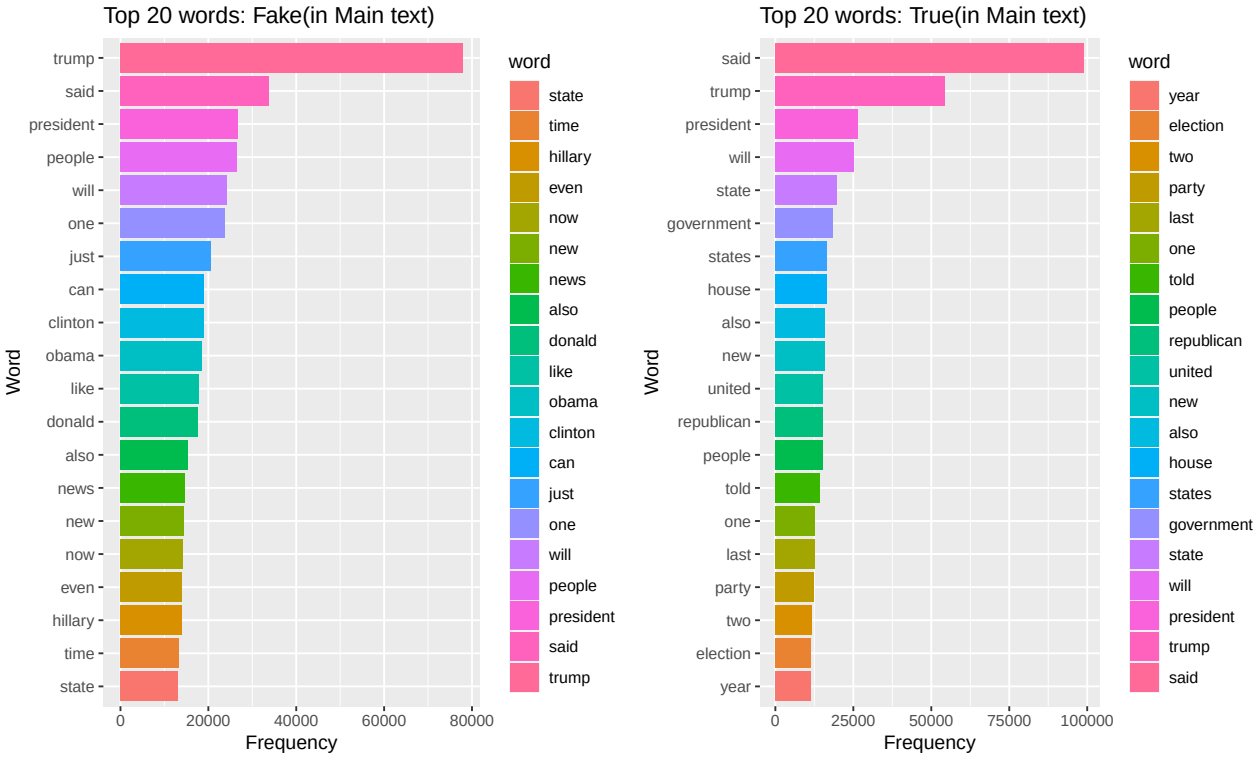



```
# 3. Read frequency data for MainText
df.fakeMainText.freq <- read.csv('../data/Fake_wordFreq_MainText.csv', encoding =
↪  'UTF-8',
                    header = TRUE)
df.trueMainText.freq <- read.csv('../data/True_wordFreq_MainText.csv', encoding =
↪  'UTF-8',
                    header = TRUE)

# 2. Draw bar plot of top 20 for MainText
myColors <- brewer.pal(11, "Spectral")
fake.bar <- top_n(df.fakeMainText.freq, 20, freq) %>%
                    arrange(freq) %>%
                    mutate(word = factor(word, levels = word)) %>%
                    ggplot(aes(x = word, y = freq, fill = word)) +
                    geom_bar(stat="identity") +
                    scale_colour_manual(name = "word", values = myColors) +
                    labs(title = "Top 20 words: Fake(in Main text)",
                                 x ="Word", y = "Frequency") +
                    coord_flip()
```

```
true.bar <- top_n(df.trueMainText.freq, 20, freq) %>%
                    arrange(freq) %>%
                    mutate(word = factor(word, levels = word)) %>%
                    ggplot(aes(x = word, y = freq, fill = word)) +
                    geom_bar(stat="identity") +
                    scale_colour_manual(name = "word", values = myColors) +
                    labs(title = "Top 20 words: True(in Main text)",
                         x ="Word", y = "Frequency") +
                    coord_flip()

ggarrange(fake.bar, true.bar,
          ncol = 2, nrow = 1)
```



3. EDA of Linguistic Inquiry and Word Count(LIWC) data

3-1. Pre-processing LIWC Result

```
# 1. Read Data
df.fake.LIWC <- read.csv('../data/565_Fake_LIWC.csv', encoding = 'UTF-8',
                    header = TRUE)
df.true.LIWC <- read.csv('../data/565_True_LIWC.csv', encoding = 'UTF-8',
                    header = TRUE)

# 2. Combine Data
df.fake.LIWC$Class <- 0
```

```
df.true.LIWC$Class <- 1
df.full.LIWC <- rbind(df.fake.LIWC, df.true.LIWC)
df.full.LIWC$Class <- as.factor(df.full.LIWC$Class)
          # give column names which are missing
colnames(df.full.LIWC)[1:4] <- c("Title", "MainText", "Category", "Date")
colnames(df.full.LIWC)
```

```
##  [1] "Title"        "MainText"     "Category"     "Date"         "WC"
##  [6] "Tone"         "WPS"          "Sixltr"       "Dic"          "function."
## [11] "pronoun"      "ppron"        "i"            "we"           "you"
## [16] "shehe"        "they"         "ipron"        "article"      "prep"
## [21] "auxverb"      "adverb"       "conj"         "negate"       "verb"
## [26] "adj"          "compare"      "interrog"     "number"       "quant"
## [31] "affect"       "posemo"       "negemo"       "anx"          "anger"
## [36] "sad"          "social"       "family"       "friend"       "female"
## [41] "male"         "cogproc"      "insight"      "cause"        "discrep"
## [46] "tentat"       "certain"      "differ"       "percept"      "see"
## [51] "hear"         "feel"         "bio"          "body"         "health"
## [56] "sexual"       "ingest"       "drives"       "affiliation"  "achieve"
## [61] "power"        "reward"       "risk"         "focuspast"    "focuspresent"
## [66] "focusfuture"  "relativ"      "motion"       "space"        "time"
## [71] "work"         "leisure"      "home"         "money"        "relig"
## [76] "death"        "informal"     "swear"        "netspeak"     "assent"
## [81] "nonflu"       "filler"       "AllPunc"      "Period"       "Comma"
## [86] "Colon"        "SemiC"        "QMark"        "Exclam"       "Dash"
## [91] "Quote"        "Apostro"      "Parenth"      "OtherP"       "Class"
```

```
# 3. Clean up Data
df.full.LIWC$Title <- gsub("[][!#$%()*,.:;<=>@^_|~.{}]", " ", df.full.LIWC$Title)
df.full.LIWC$Title <- gsub("‘|’|"|"|-", " ", df.full.LIWC$Title)

df.full.LIWC$MainText <- gsub("[][!#$%()*,.:;<=>@^_|~.{}]", " ", df.full.LIWC$MainText)
df.full.LIWC$MainText <- gsub("‘|’|"|"|-", " ", df.full.LIWC$MainText)

# 4. Look through the data
          # dimension
dim(df.full.LIWC)
```

```
## [1] 44898    95
```

```
          # check NA values
table(is.na(df.full.LIWC))
```

```
##
##   FALSE
## 4265310
```

```
          # simple view of the data
head(df.full.LIWC, 1)
```

```
## Title
## 1 Donald Trump Sends Out Embarrassing New Year s Eve Message This is Disturbing
## MainText
## 1 Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that Instead he had to give a shout out to his enemies haters and the very dishonest fake news media The former reality show star had just one job to do and he couldn t do it As our Country rapidly grows stronger and smarter I want to wish all of my friends supporters enemies haters and even the very dishonest Fake News Media a Happy and Healthy New Year President Angry Pants tweeted 2018 will be a great year for America As our Country rapidly grows stronger and smarter I want to wish all of my friends supporters enemies haters and even the very dishonest Fake News Media a Happy and Healthy New Year 2018 will be a great year for America Donald J Trump realDonaldTrump December 31 2017Trump s tweet went down about as welll as you d expect What kind of president sends a New Year s greeting like this despicable petty infantile gibberish? Only Trump His lack of decency won t even allow him to rise above the gutter long enough to wish the American citizens a happy new year Bishop Talbert Swan TalbertSwan December 31 2017no one likes you Calvin calvinstowell December 31 2017Your impeachment would make 2018 a great year for America but I ll also accept regaining control of Congress Miranda Yaver mirandayaver December 31 2017Do you hear yourself talk? When you have to include that many people that hate you you have to wonder? Why do the they all hate me?  Alan Sandoval AlanSandoval13 December 31 2017Who uses the word Haters in a New Years wish??  Marlene marlene399 December 31 2017You can t just say happy new year?  Koren pollitt Korencarpenter December 31 2017Here s Trump s New Year s Eve tweet from 2016 Happy New Year to all including to my many enemies and those who have fought me and lost so badly they just don t know what to do Love Donald J Trump realDonaldTrump December 31 2016This is nothing new for Trump He s been doing this for years Trump has directed messages to his enemies and haters for New Year s Easter Thanksgiving and the anniversary of 9/11 pic twitter com/4FPAe2KypA Daniel Dale ddale8 December 31 2017Trump s holiday tweets are clearly not presidential How long did he work at Hallmark before becoming President?  Steven Goodine SGoodine December 31 2017He s always been like this the only difference is that in the last few years his filter has been breaking down Roy Schulze thbthttt December 31 2017Who apart from a teenager uses the term haters?  Wendy WendyWhistles December 31 2017he s a fucking 5 year old Who Knows rainyday80 December 31 2017So to all the people who voted for this a hole thinking he would change once he got into power you were wrong 70-year-old men don t change and now he s a year older Photo by Andrew Burton/Getty Images
## Category Date WC Tone WPS Sixltr Dic function. pronoun
## 1 News December 31, 2017 516 14.96 17.79 20.54 72.09 36.82 9.5
## ppron i we you shehe they ipron article prep auxverb adverb conj negate
## 1 6.2 1.55 0.39 1.55 2.33 0.39 3.29 5.23 9.3 5.62 4.07 4.46 0.39
## verb adj compare interrog number quant affect posemo negemo anx anger sad
## 1 11.43 8.33 2.71 1.55 4.65 1.74 8.91 4.07 4.84 0.39 3.29 0.19
## social family friend female male cogproc insight cause discrep tentat certain
## 1 10.85 0 0.39 0 2.52 7.95 1.16 1.74 2.13 0.39 1.74
## differ percept see hear feel bio body health sexual ingest drives
## 1 0.97 1.36 0.78 0.58 0 0.39 0 0 0.19 0 6.98
## affiliation achieve power reward risk focuspast focuspresent focusfuture
## 1 2.52 0.58 2.71 0.97 0.39 2.71 7.36 1.55
## relativ motion space time work leisure home money relig death informal swear
## 1 15.7 1.36 3.49 11.05 1.36 0.39 0 0 0.19 0 1.74 0.19
## netspeak assent nonflu filler AllPunc Period Comma Colon SemiC QMark Exclam
## 1 0.39 0 0 0 23.84 4.46 7.56 0 0.19 1.74 1.16
## Dash Quote Apostro Parenth OtherP Class
## 1 0.39 0 0.19 5.04 3.1 0
```

3-2. Correlations between LIWC Variables

```
# 1. Select upper-level variables
top_var <- c("Class", "WC", "Tone", "WPS", "Sixltr", "function.", "ppron",
             "article", "prep", "auxverb", "adverb", "conj", "negate", "verb",
             "adj", "interrog", "quant", "affect", "social", "cogproc",
             "percept", "bio", "drives", "relativ", "work", "leisure", "home",
             "money", "relig", "death", "informal")

# 2. Build correlation table
df.full.LIWC.top <- df.full.LIWC[, top_var]
df.full.LIWC.top$Class <- as.numeric(df.full.LIWC.top$Class)
LIWC.cor <- cor(df.full.LIWC.top)
head(round(LIWC.cor, 3), 10)
```

```
##            Class     WC   Tone    WPS Sixltr function.  ppron article   prep
## Class      1.000 -0.073  0.064 -0.238  0.458    -0.290 -0.435   0.235  0.300
## WC        -0.073  1.000 -0.073  0.212 -0.052     0.259  0.072   0.122  0.056
## Tone       0.064 -0.073  1.000 -0.045  0.026    -0.040  0.018  -0.010 -0.047
## WPS       -0.238  0.212 -0.045  1.000  0.028     0.125 -0.087   0.149  0.149
## Sixltr     0.458 -0.052  0.026  0.028  1.000    -0.500 -0.510   0.115  0.114
## function. -0.290  0.259 -0.040  0.125 -0.500     1.000  0.560   0.190  0.146
## ppron     -0.435  0.072  0.018 -0.087 -0.510     0.560  1.000  -0.326 -0.245
## article    0.235  0.122 -0.010  0.149  0.115     0.190 -0.326   1.000  0.216
## prep       0.300  0.056 -0.047  0.149  0.114     0.146 -0.245   0.216  1.000
## auxverb   -0.127  0.099 -0.013 -0.040 -0.287     0.584  0.286  -0.081 -0.222
##            auxverb adverb   conj negate   verb    adj interrog  quant affect
## Class      -0.127 -0.491 -0.190 -0.063 -0.114 -0.049   -0.277 -0.080 -0.214
## WC          0.099  0.115  0.244  0.022  0.010  0.062    0.077  0.159 -0.027
## Tone       -0.013 -0.021 -0.018  0.009  0.026  0.123   -0.036  0.022  0.042
## WPS        -0.040  0.037  0.170 -0.078 -0.121  0.057    0.016  0.097 -0.083
## Sixltr     -0.287 -0.453 -0.223 -0.129 -0.387 -0.060   -0.277 -0.124 -0.111
## function.   0.584  0.533  0.466  0.272  0.523  0.003    0.358  0.160  0.045
## ppron       0.286  0.435  0.259  0.184  0.430 -0.021    0.301  0.004  0.176
## article    -0.081 -0.225 -0.063 -0.105 -0.149 -0.020   -0.102  0.002 -0.155
## prep       -0.222 -0.235 -0.075 -0.158 -0.206  0.013   -0.168  0.040 -0.187
## auxverb     1.000  0.304  0.173  0.289  0.726 -0.003    0.244  0.094  0.031
##             social cogproc percept    bio drives relativ   work leisure   home
## Class     -0.214  -0.199  -0.103 -0.167  0.221   0.294  0.319  -0.278  0.065
## WC        -0.021   0.142  -0.234  0.007 -0.023  -0.032 -0.002  -0.157 -0.041
## Tone       0.028   0.039   0.015 -0.099  0.082  -0.073  0.132   0.088  0.018
## WPS       -0.097   0.034  -0.231  0.017 -0.022   0.029  0.036  -0.152  0.015
## Sixltr    -0.301  -0.214  -0.144 -0.125  0.244   0.023  0.412  -0.137  0.009
## function.  0.342   0.478  -0.182  0.020 -0.127  -0.106 -0.232  -0.238 -0.077
## ppron      0.705   0.284   0.103  0.092 -0.093  -0.232 -0.292   0.075 -0.049
## article   -0.241  -0.104  -0.234 -0.056  0.051   0.155  0.118  -0.242  0.040
## prep      -0.178  -0.188  -0.209 -0.033  0.079   0.395  0.090  -0.233  0.043
## auxverb    0.169   0.426  -0.047 -0.031 -0.088  -0.192 -0.119  -0.104 -0.086
##             money  relig  death informal
## Class       0.153  0.057  0.003   -0.303
## WC          0.035 -0.037  0.015   -0.077
## Tone        0.111 -0.061 -0.251   -0.002
```

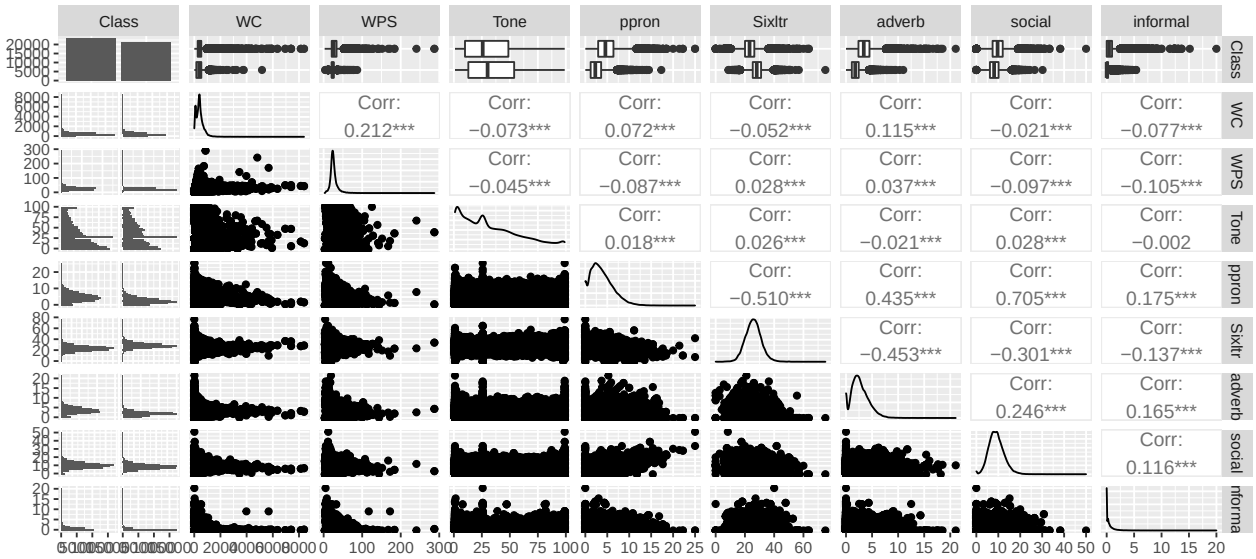
```
## WPS        0.028 -0.009 -0.003   -0.105
## Sixltr     0.125  0.063 -0.050   -0.137
## function. -0.098 -0.058 -0.002   -0.139
## ppron     -0.160 -0.061 -0.031    0.175
## article    0.053  0.019  0.054   -0.285
## prep       0.049  0.050  0.043   -0.309
## auxverb   -0.052 -0.039 -0.012   -0.032
```

```
# 3. Show first 20 biggest correlation (in absolute value)
LIWC.cor[lower.tri(LIWC.cor, diag=TRUE)] <- 0
LIWC.cor.sorted <- sort(abs(LIWC.cor), decreasing=T)
LIWC.cor.top20 <- data.frame()
for (val in 1:20){
  vars.big.cor <- arrayInd(which(abs(LIWC.cor) == LIWC.cor.sorted[val]),
                           dim(LIWC.cor))
  LIWC.cor.top20 <- rbind(LIWC.cor.top20,
                          c(colnames(df.full.LIWC.top)[vars.big.cor],
                            round(LIWC.cor.sorted[val], 3)))
}
colnames(LIWC.cor.top20) <- c("Var1", "Var2", "Correlation")
LIWC.cor.top20
```

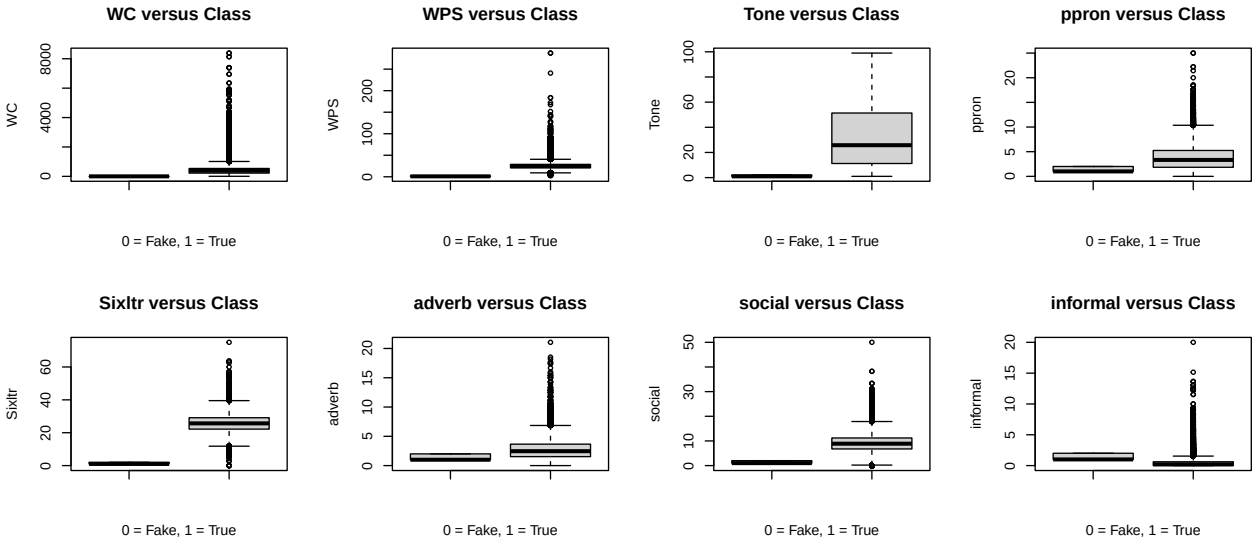
```
##         Var1      Var2 Correlation
## 1    auxverb      verb       0.726
## 2      ppron    social       0.705
## 3  function.   auxverb       0.584
## 4  function.     ppron        0.56
## 5  function.    adverb       0.533
## 6  function.      verb       0.523
## 7     Sixltr     ppron        0.51
## 8     Sixltr function.         0.5
## 9      Class    adverb       0.491
## 10 function.   cogproc       0.478
## 11 function.      conj       0.466
## 12     Class    Sixltr       0.458
## 13    Sixltr    adverb       0.453
## 14   percept   leisure       0.452
## 15      work     money        0.45
## 16      verb   cogproc        0.44
## 17     ppron    adverb       0.435
## 18     Class     ppron       0.435
## 19     ppron      verb        0.43
## 20    adverb  interrog       0.426
```

3-3. Visualizing Relationships among LIWC variables

```
df.subfull.LIWC <- df.full.LIWC %>%
          select("Class", "WC", "WPS", "Tone", "ppron", "Sixltr", "adverb",
                 "social", "informal")
ggpairs(df.subfull.LIWC)
```




```
par(mfrow=c(2,4))
boxplot(df.subfull.LIWC$Class, df.subfull.LIWC$WC, main="WC versus Class",
     xlab="0 = Fake, 1 = True", ylab="WC")
boxplot(df.subfull.LIWC$Class, df.subfull.LIWC$WPS, main="WPS versus Class",
     xlab="0 = Fake, 1 = True", ylab="WPS")
boxplot(df.subfull.LIWC$Class, df.subfull.LIWC$Tone, main="Tone versus Class",
     xlab="0 = Fake, 1 = True", ylab="Tone")
boxplot(df.subfull.LIWC$Class, df.subfull.LIWC$ppron, main="ppron versus Class",
     xlab="0 = Fake, 1 = True", ylab="ppron")
boxplot(df.subfull.LIWC$Class, df.subfull.LIWC$Sixltr, main="Sixltr versus Class",
     xlab="0 = Fake, 1 = True", ylab="Sixltr")
boxplot(df.subfull.LIWC$Class, df.subfull.LIWC$adverb, main="adverb versus Class",
     xlab="0 = Fake, 1 = True", ylab="adverb")
boxplot(df.subfull.LIWC$Class, df.subfull.LIWC$social, main="social versus Class",
     xlab="0 = Fake, 1 = True", ylab="social")
boxplot(df.subfull.LIWC$Class, df.subfull.LIWC$informal, main="informal versus Class",
     xlab="0 = Fake, 1 = True", ylab="informal")
```



```r
par(mfrow=c(1,1))
```

3-4. Testing means between two different groups

```r
t.test(df.subfull.LIWC$WC[df.subfull.LIWC$Class==0],
       df.subfull.LIWC$WC[df.subfull.LIWC$Class==1])
```

```
##
## Welch Two Sample t-test
##
## data: df.subfull.LIWC$WC[df.subfull.LIWC$Class == 0] and
df.subfull.LIWC$WC[df.subfull.LIWC$Class == 1]
## t = 15.727, df = 40917, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 45.91219 58.98568
## sample estimates:
## mean of x mean of y
## 451.0233 398.5744
```

```r
t.test(df.subfull.LIWC$WPS[df.subfull.LIWC$Class==0],
       df.subfull.LIWC$WPS[df.subfull.LIWC$Class==1])
```

```
##
## Welch Two Sample t-test
##
## data: df.subfull.LIWC$WPS[df.subfull.LIWC$Class == 0] and
df.subfull.LIWC$WPS[df.subfull.LIWC$Class == 1]
## t = 53.82, df = 29425, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 4.491578 4.831096
## sample estimates:
## mean of x mean of y
## 28.20337 23.54204
```

```r
t.test(df.subfull.LIWC$Tone[df.subfull.LIWC$Class==0],
       df.subfull.LIWC$Tone[df.subfull.LIWC$Class==1])
```

```
##
## Welch Two Sample t-test
##
## data: df.subfull.LIWC$Tone[df.subfull.LIWC$Class == 0] and
df.subfull.LIWC$Tone[df.subfull.LIWC$Class == 1]
## t = -13.638, df = 44535, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -3.941043 -2.950586
```

```
## sample estimates:
## mean of x mean of y
## 32.23441 35.68022
```

```
t.test(df.subfull.LIWC$ppron[df.subfull.LIWC$Class==0],
       df.subfull.LIWC$ppron[df.subfull.LIWC$Class==1])
```

```
##
## Welch Two Sample t-test
##
## data: df.subfull.LIWC$ppron[df.subfull.LIWC$Class == 0] and
df.subfull.LIWC$ppron[df.subfull.LIWC$Class == 1]
## t = 103.99, df = 42262, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 2.187761 2.271816
## sample estimates:
## mean of x mean of y
## 4.805203 2.575414
```

```
t.test(df.subfull.LIWC$social[df.subfull.LIWC$Class==0],
       df.subfull.LIWC$social[df.subfull.LIWC$Class==1])
```

```
##
## Welch Two Sample t-test
##
## data: df.subfull.LIWC$social[df.subfull.LIWC$Class == 0] and
df.subfull.LIWC$social[df.subfull.LIWC$Class == 1]
## t = 46.795, df = 44635, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 1.430855 1.555959
## sample estimates:
## mean of x mean of y
## 9.828172 8.334765
```

```
t.test(df.subfull.LIWC$adverb[df.subfull.LIWC$Class==0],
       df.subfull.LIWC$adverb[df.subfull.LIWC$Class==1])
```

```
##
## Welch Two Sample t-test
##
## data: df.subfull.LIWC$adverb[df.subfull.LIWC$Class == 0] and
df.subfull.LIWC$adverb[df.subfull.LIWC$Class == 1]
## t = 122.02, df = 39679, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 1.649508 1.703367
## sample estimates:
## mean of x mean of y
## 3.495365 1.818927
```

```
t.test(df.subfull.LIWC$Sixltr[df.subfull.LIWC$Class==0],
       df.subfull.LIWC$Sixltr[df.subfull.LIWC$Class==1])
```

```
##
## Welch Two Sample t-test
##
## data: df.subfull.LIWC$Sixltr[df.subfull.LIWC$Class == 0] and
df.subfull.LIWC$Sixltr[df.subfull.LIWC$Class == 1]
## t = -109.77, df = 44890, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -5.086918 -4.908448
## sample estimates:
## mean of x mean of y
## 23.36330 28.36098
```

```
t.test(df.subfull.LIWC$informal[df.subfull.LIWC$Class==0],
       df.subfull.LIWC$informal[df.subfull.LIWC$Class==1])
```

```
##
## Welch Two Sample t-test
##
## data: df.subfull.LIWC$informal[df.subfull.LIWC$Class == 0] and
df.subfull.LIWC$informal[df.subfull.LIWC$Class == 1]
## t = 69.832, df = 30397, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 0.5532084 0.5851598
## sample estimates:
## mean of x mean of y
## 0.7851523 0.2159682
```