# War and Peace

Inaugural speeches are delivered in inaugural events signifying induction of a President into office. They set forth the political principles that will guide the new administration, and are very insightful when it comes to learning about the intentions of the new government.

We are given 58 inaugural speeches by 45 presidents of the United States of America. Inspired by an article in Washington Post, titled-*"Here's how much of your life the United States has been at war"*,I decided to study the speeches of American Presidents by categorizing them into "War time" Presidents and "Peace time" Presidents and see if there is a remarkable difference in the sentence construction, emotions, and topic of their speeches.

For the purpose of this study, a war time president is the one who was involved in signing a war declaration, and actively engaged in taking the US to war. While, a peace time president is the one who either did not have any war during his term or was involved in ending an ongoing war. I took the following presidents in the "War Time Presidents" cohort:- Woodrow Wilson (World War I), Franklin D. Roosevelt (World War II), Lyndon B. Johnson (Vietnam War), and George W. Bush (Iraq War). The following are the presidents, I have included in the "Peace Time Presidents":- John Quincy Adams, Ulysses S. Grant, Dwight D. Eisenhower, and Jimmy Carter.

We begin the analysis by loading the relevant packages, and preparing data for analysis. The study is divided into the following parts: 1. Load packages and data preparation 2. Analysis- (2.1.) Most commonly used words (2.2.) Sentence lengths (2.3.) Sentiment Analysis (2.4.)Topic Modeling 3. Conclusion

## 1. Load packages and data preparation

```
packages.used=c("rvest", "tibble", "qdap",
                "sentimentr", "gplots", "dplyr",
                "tm", "syuzhet", "factoextra",
                "beeswarm", "scales", "RColorBrewer",
                "RANN", "tm", "topicmodels","readxl")

# check packages that need to be installed.
packages.needed=setdiff(packages.used,
                        intersect(installed.packages()[,1],
                                  packages.used))
# install additional packages
if(length(packages.needed)>0){
  install.packages(packages.needed, dependencies = TRUE)
}

# load packages
library("rvest")
library("tibble")

library("qdap")
library("sentimentr")
library("gplots")
library("dplyr")
library("tm")
library("syuzhet")
library("factoextra")
```

```r
library("beeswarm")
library("scales")
library("RColorBrewer")
library("RANN")
library("tm")
library("topicmodels")
library(readxl)
```

Cloud Package

```r
packages.used=c("tm", "wordcloud", "RColorBrewer",
                "dplyr", "tidytext")

# check packages that need to be installed.
packages.needed=setdiff(packages.used,
                        intersect(installed.packages()[,1],
                                  packages.used))
# install additional packages
if(length(packages.needed)>0){
  install.packages(packages.needed, dependencies = TRUE,
                   repos='http://cran.us.r-project.org')
}

library(tm)
library(wordcloud)
library(RColorBrewer)
library(dplyr)
library(tidytext)
```

```r
dates<-read.table("/Users/anshuma/Documents/GitHub/Spring2018-Project1-anshumachandak/data/InauguationD
info<-read_excel("/Users/anshuma/Documents/GitHub/Spring2018-Project1-anshumachandak/data/InaugurationI
info<-as.data.frame(info)
#merging both data sets
colnames(dates)[1]<-"President"
all_info<-merge(info,dates,by="President")

f.all<-list.files(path="/Users/anshuma/Documents/GitHub/Spring2018-Project1-anshumachandak/data/Inaugura
f.list<-as.character(lapply(f.all,function(i) readLines(i)))
speech_df<-data_frame(text=f.list)
speech_df$ID <- seq.int(nrow(speech_df))

p_names<-read_excel("/Users/anshuma/Desktop/Columbia/1. Applied Data Science/R/Presidents_names.xlsx")
p_names<-as.data.frame(p_names)
speech_master<-merge(p_names,speech_df,by="ID") #adding a president column in dataframe

Term<-NULL    #adding terms to the speech_master
for (i in 1:58){
  text.name<-list.files(path="/Users/anshuma/Documents/GitHub/Spring2018-Project1-anshumachandak/data/I
  term<-substring(text.name,nchar(text.name)-4,nchar(text.name)-4)
  term<-as.numeric(term)
  Term<-c(Term,term)
}
Term<-data.frame(Term)
speech_master <- cbind(speech_master,Term)
```

Making two data sets- war_p for "War Time" Presidents and peace_df for "Peace Time"

```r
war_p<-c("Woodrow Wilson","Franklin D. Roosevelt","Lyndon B. Johnson","George W. Bush")
war_df<-speech_master[speech_master$President %in% war_p,]
peace_p<-c("Ulysses S. Grant","Jimmy Carter","Dwight D. Eisenhower","John Quincy Adams")
peace_df<-speech_master[speech_master$President %in% peace_p,]
```

Clean files: WAR FILES

```r
war_clean <- Corpus(VectorSource(war_df))
war_clean <- tm_map(war_clean, tolower)
war_clean<-tm_map(war_clean, stripWhitespace)
war_clean<-tm_map(war_clean, content_transformer(tolower))
war_clean<-tm_map(war_clean, removeWords, stopwords("english"))
war_clean<-tm_map(war_clean, removeWords, c("will","shall","can","upon","let","day","like")) #removing
war_clean<-tm_map(war_clean, removeWords, character(0))
war_clean<-tm_map(war_clean, removePunctuation)

tdm.all_war<-TermDocumentMatrix(war_clean)

tdm.tidy_war=tidy(tdm.all_war)

tdm.overall_war=summarise(group_by(tdm.tidy_war, term), sum(count))

dtm <- DocumentTermMatrix(war_clean,
                        control = list(weighting = function(x)
                                               weightTfIdf(x,
                                                          normalize =FALSE),
                                        stopwords = TRUE))
ff.dtm=tidy(dtm)
```

Clean Files: PEACE FILES

```r
peace_clean <- Corpus(VectorSource(peace_df))
peace_clean  <- tm_map(peace_clean , tolower)
peace_clean <-tm_map(peace_clean , stripWhitespace)
peace_clean <-tm_map(peace_clean , content_transformer(tolower))
peace_clean <-tm_map(peace_clean , removeWords, stopwords("english"))
peace_clean<-tm_map(peace_clean, removeWords, c("will","shall","can","upon","let","day","like")) #remov
peace_clean <-tm_map(peace_clean , removeWords, character(0))
peace_clean <-tm_map(peace_clean , removePunctuation)

tdm.peace<-TermDocumentMatrix(peace_clean )

tdm.tidy_peace=tidy(tdm.peace)

tdm.overall_peace=summarise(group_by(tdm.tidy_peace, term), sum(count))

dtm_peace <- DocumentTermMatrix(peace_clean,
                        control = list(weighting = function(x)
                                               weightTfIdf(x,
                                                          normalize =FALSE),
                                        stopwords = TRUE))
ff.dtm_peace=tidy(dtm_peace)

head(ff.dtm_peace$count)
```

```
## [1] 2 2 2 2 2 4
```

FDR:

```
FDR<-speech_master[speech_master$President == "Franklin D. Roosevelt",]
FDR.t1<-FDR[FDR$Term==1,]
FDR.t2<-FDR[FDR$Term==2,]
FDR.t3<-FDR[FDR$Term==3,]
FDR.t4<-FDR[FDR$Term==4,]
```

Clean FDR term 1

```
t1_clean <- Corpus(VectorSource(FDR.t1))
t1_clean <- tm_map(t1_clean, tolower)
t1_clean<-tm_map(t1_clean, stripWhitespace)
t1_clean<-tm_map(t1_clean, content_transformer(tolower))
t1_clean<-tm_map(t1_clean, removeWords, stopwords("english"))
t1_clean<-tm_map(t1_clean, removeWords, character(0))
t1_clean<-tm_map(t1_clean, removeWords, c("will","shall","can","upon","let","day","like")) #removing so
t1_clean<-tm_map(t1_clean, removePunctuation)

t1_clean.all<-TermDocumentMatrix(t1_clean)

t1_clean.tidy=tidy(t1_clean.all)

t1_clean.overall=summarise(group_by(t1_clean.tidy, term), sum(count))
```

Clean FDR term 3

```
t3_clean <- Corpus(VectorSource(FDR.t3))
t3_clean <- tm_map(t3_clean, tolower)
t3_clean<-tm_map(t3_clean, stripWhitespace)
t3_clean<-tm_map(t3_clean, content_transformer(tolower))
t3_clean<-tm_map(t3_clean, removeWords, stopwords("english"))
t3_clean<-tm_map(t3_clean, removeWords, character(0))
t3_clean<-tm_map(t3_clean, removePunctuation)
t3_clean<-tm_map(t3_clean, removeWords, c("will","shall","can","upon","let","day","like")) #removing so
t3_clean.all<-TermDocumentMatrix(t3_clean)

t3_clean.tidy=tidy(t3_clean.all)

t3_clean.overall=summarise(group_by(t3_clean.tidy, term), sum(count))
```

Clean FDR term 4

```
t4_clean <- Corpus(VectorSource(FDR.t4))
t4_clean <- tm_map(t4_clean, tolower)
t4_clean<-tm_map(t4_clean, stripWhitespace)
t4_clean<-tm_map(t4_clean, content_transformer(tolower))
t4_clean<-tm_map(t4_clean, removeWords, stopwords("english"))
t4_clean<-tm_map(t4_clean, removeWords, character(0))
t4_clean<-tm_map(t4_clean, removePunctuation)
t4_clean<-tm_map(t4_clean, removeWords, c("will","shall","can","upon","let","day","like")) #removing so
t4_clean.all<-TermDocumentMatrix(t4_clean)

t4_clean.tidy=tidy(t4_clean.all)
```

```
t4_clean.overall=summarise(group_by(t4_clean.tidy, term), sum(count))
```

Words per sentence:

Words per sentence for peace time presidents:

```
sentence.list=NULL
for(i in 1:nrow(peace_df)){
  sentences=sent_detect(peace_df$text[i],
                        endmarks = c("?", ".", "!", "|",";")) #the sentences ending with these mark the
  if(length(sentences)>0){
    emotions=get_nrc_sentiment(sentences)
    word.count=word_count(sentences)
    # colnames(emotions)=paste0("emo.", colnames(emotions))
    # in case the word counts are zeros?
    emotions=diag(1/(word.count+0.01))%*%as.matrix(emotions)
    sentence.list=rbind(sentence.list,
                        cbind(peace_df[i,-ncol(peace_df)],
                              sentences=as.character(sentences),
                              word.count,
                              emotions,
                              sent.id=1:length(sentences)
                              )
    )
  }
}
```

Some non-sentences exist in raw data due to erroneous extra end-of sentence marks.

```
sentence.list=
  sentence.list%>%
  filter(!is.na(word.count))
```

Words per sentence for War time presidents:

```
sentence.list.w=NULL
for(i in 1:nrow(war_df)){
  sentences.w=sent_detect(war_df$text[i],
                          endmarks = c("?", ".", "!", "|",";")) #the sentences ending with these mark the
  if(length(sentences.w)>0){
    emotions.w=get_nrc_sentiment(sentences.w)
    word.count.w=word_count(sentences.w)
    # colnames(emotions)=paste0("emo.", colnames(emotions))
    # in case the word counts are zeros?
    emotions.w=diag(1/(word.count.w+0.01))%*%as.matrix(emotions.w)
    sentence.list.w=rbind(sentence.list.w,
                          cbind(war_df[i,-ncol(war_df)],
                                sentences.w=as.character(sentences.w),
                                word.count.w,
                                emotions.w,
                                sent.id=1:length(sentences.w)
                                )
    )
  }
}
```

Some non-sentences exist in raw data due to erroneous extra end-of sentence marks.

```
sentence.list.w=
  sentence.list.w%>%
  filter(!is.na(word.count.w))
```

## 2. Analysis

Franklin D. Roosevelt, popularly known as FDR, served 4 terms as the President of the USA. He was the longest serving president of the USA, and also played an instrumental role in shaping America, and world politics. Although, in this study we are not analysing a single President, it is definitely worth studying his speeches to know how his thoughts and usage of words changed over time- i.e. from becoming the President for the first time in 1933 to serving his third term amidst the World War II and then finally, serving his fourth term- also the end of the war. We have made word clouds of the speeches of his first term, third term, and fourth term inauguration speeches.

```
par(mfrow=c(1,3))

wordcloud(t1_clean.overall$term, t1_clean.overall$`sum(count)`,
          scale=c(2.8,0.2),
          max.words=50,
          min.freq=1,
          random.order=FALSE,
          rot.per=0.3,
          use.r.layout=T,
          random.color=FALSE,
          colors=brewer.pal(9,"YlGn"))
```

```
## Warning in wordcloud(t1_clean.overall$term, t1_clean.overall
## $`sum(count)`, : important could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(t1_clean.overall$term, t1_clean.overall
## $`sum(count)`, : stricken could not be fit on page. It will not be plotted.

## Warning in wordcloud(t1_clean.overall$term, t1_clean.overall
## $`sum(count)`, : without could not be fit on page. It will not be plotted.
```

```
wordcloud(t3_clean.overall$term, t3_clean.overall$`sum(count)`,
          scale=c(2.8,0.2),
          max.words=50,
          min.freq=1,
          random.order=FALSE,
          rot.per=0.3,
          use.r.layout=T,
          random.color=FALSE,
          colors=brewer.pal(9,"YlGn"))
```

```
## Warning in strwidth(words[i], cex = size[i], ...): conversion failure on
## '-' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in strwidth(words[i], cex = size[i], ...): conversion failure on
## '-' in 'mbcsToSbcs': dot substituted for <80>

## Warning in strwidth(words[i], cex = size[i], ...): conversion failure on
## '-' in 'mbcsToSbcs': dot substituted for <94>

## Warning in text.default(x1, y1, words[i], cex = size[i], offset = 0, srt =
```

```
## rotWord * : conversion failure on '-' in 'mbcsToSbcs': dot substituted for
## <e2>

## Warning in text.default(x1, y1, words[i], cex = size[i], offset = 0, srt =
## rotWord * : conversion failure on '-' in 'mbcsToSbcs': dot substituted for
## <80>

## Warning in text.default(x1, y1, words[i], cex = size[i], offset = 0, srt =
## rotWord * : conversion failure on '-' in 'mbcsToSbcs': dot substituted for
## <94>

## Warning in text.default(x1, y1, words[i], cex = size[i], offset = 0, srt =
## rotWord * : font metrics unknown for Unicode character U+2014
```

```r
wordcloud(t4_clean.overall$term, t4_clean.overall$`sum(count)`,
          scale=c(2.8,0.2),
          max.words=50,
          min.freq=1,
          random.order=FALSE,
          rot.per=0.3,
          use.r.layout=T,
          random.color=FALSE,
          colors=brewer.pal(9,"YlGn"))
```



It is fascinating to see how the most commonly used words by President Roosevelt changed so drastically over these three terms. The speeches are very much representative of his thoughts, and the popular sentiments around that period. In his first term (left most word cloud), he frequently used a variety of words like national, people, help, world, money, etc. In his third term, his speech seemed more succint and focussed with words like democracy, freedom, spirit, life, America. In his fourth term, which was the year World War II ended, the word "Peace" took the spotlight, which makes sense because the war had left millions of people dead, as well as homeless.

## 2.1. Most commonly used words

Coming back to the focus of the study i.e. whether usage of words and sentence construction in inaugural speeches of the Presidents give an insight into the president being a "war time" president or a "peace time" president. We start with visualizing the most frequently used words in the speeches of war-time and peace-time presidents.

```r
par(mfrow=c(1,2))
wordcloud(tdm.overall_war$term, tdm.overall_war$`sum(count)`,
          scale=c(2.5,0.2),
          max.words=60,
```

```
        min.freq=1,
        random.order=FALSE,
        rot.per=0.3,
        use.r.layout=T,
        random.color=FALSE,
        colors=brewer.pal(9,"GnBu"))
wordcloud(tdm.overall_peace$term, tdm.overall_peace$`sum(count)`,
        scale=c(2.5,0.2),
        max.words=60,
        min.freq=1,
        random.order=FALSE,
        rot.per=0.3,
        use.r.layout=T,
        random.color=FALSE,
        colors=brewer.pal(9,"GnBu"))
```



The speeches of war-time presidents (left) have less diversity in terms of vocabulary, and we see the following words the most- people, nation, world,government. Looking at the word cloud of peace-time presidents (right), there is more diversity in the words used, with the following words having the highest frequency of showing up- peace, nation, people, freedom, free, union, government. There is more usage of words that signify solidarity in the speeches of peace-time presidents. So, does that mean times of urgency lead to speeches that are less ornate and more direct?

Next, we make a comparative analysis on the basis of sentence lengths of the speeches.

## 2.2. Sentence Lengths

Short sentences make action move faster.There is more emotional impact, when sentences are blunt.Following this, we hypothesize that war-time presidents' speeches would have shorter sentences compared to that of peace-time, as the quickening pace of shorter sentences create a sense of urgency.

```
getmode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
```

```
table.des<-cbind(c("War","Peace"),c(round(mean(sentence.list.w$word.count.w)),round(mean(sentence.list$w
colnames(table.des)<-c("President","Mean","Mode")
table.des
```

```
##      President Mean Mode
## [1,] "War"     "20" "9"
## [2,] "Peace"   "22" "14"
```

The average sentence length of the speeches by wartime presidents is 20, while it is 22 for peacetime presidents. Most of the sentences of war time presidents is of lenth 9, while it is 14 for peacetime presidents. This leads us to conclude that war-time presidents' speeches were more direct and blunt, thus exhibiting promptness.

**2.3.Sentiment Analysis**

Emotions unconsciously integrate sensory input from within, and manifest themselves in facial, body, and speech displays. War times are stressful times leading to greater display of negative emoitions.In this section, we hypothesize that war-time presidents would have more negative sentiment in their speeches.

```
#Preparing data for visualization
em_peace<-sentence.list[,6:15]
em_war<-sentence.list.w[,6:15]
d<-rbind(colSums(em_war),colSums(em_peace))
em_score<-as.data.frame(cbind(c("War time Presidents","Peace time Presidents"),d))
prez <- read.csv("prez.csv", header = T, as.is = T)
ggplot(prez)+
geom_point(aes(y = War.time.Presidents, x = Emotion,
color = "War time Presidents"))+
geom_point(aes(y = Peace.time.Presidents, x = Emotion,
color = "Peace time Presidents"))+
theme(axis.text.x=element_text(angle=90,hjust=1))+ggtitle("Emotion scores for War Time Presidents and P
```

Emotion scores for War Time Presidents and Peace Time Presidents
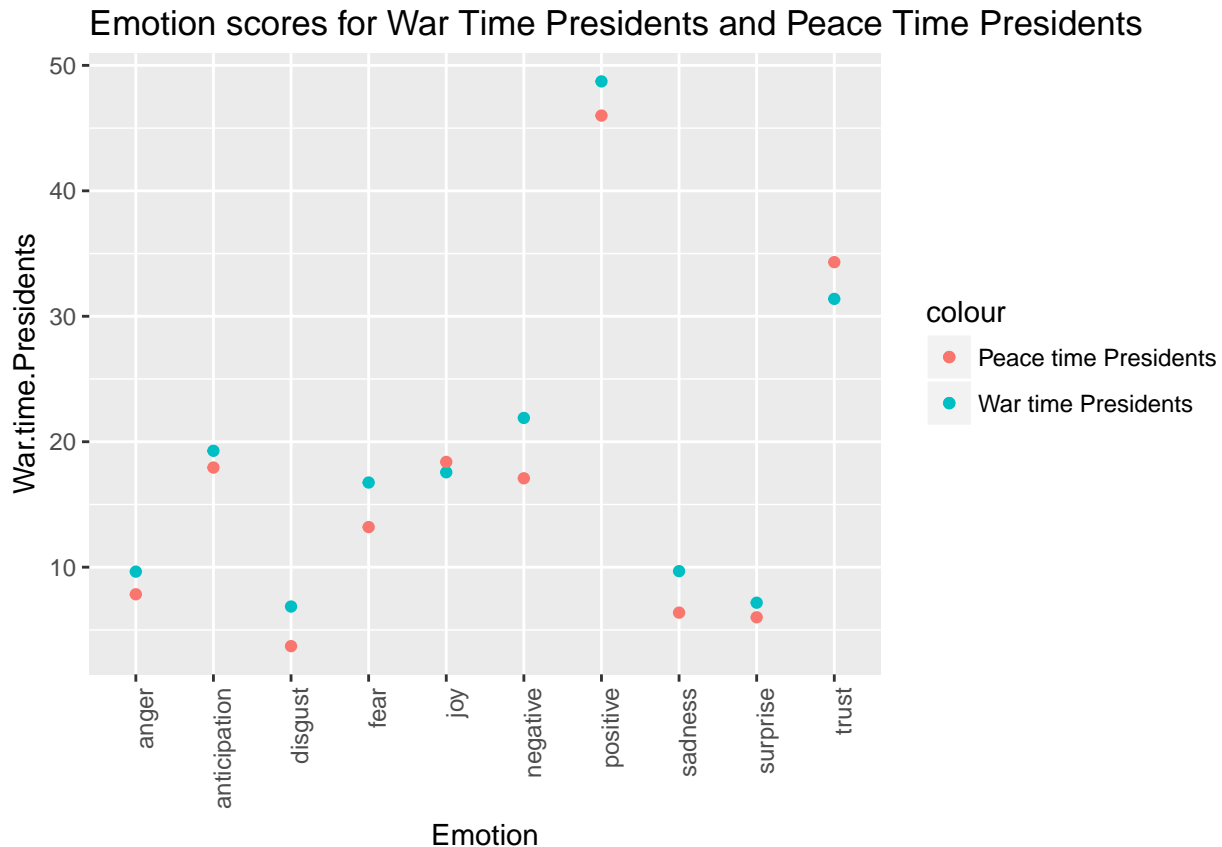
From the plot above, we can conclude that the speeches of war time presidents are more emotionally charged with higher scores for anger, disgust, fear, and negative sentiments. There is 30 per cent more negative sentiment, 23 per cent more anger, and 85 per cent more disgust in war-time presidents' speech than the peace-time presidents' speech. There is lesser trust and positive sentiment in their speeches as compared to the speeches of the peace-time presidents.

### 2.3.Topic Modeling

We are now aware of the kind of words the different presidents used, and the emotions their speeches represented. We now find out the topics that Presidents from both the groups talked about the most in their speeches. We achieve this by a tool of text mining,called topic modeling which is used to discover hidden semantic structures in a text body. We use the "Latent Dirichlet allocation" algorithm to achieve this.

For Peace-time presidents:

```
corpus.list=sentence.list[2:(nrow(sentence.list)-1), ]
sentence.pre=sentence.list$sentences[1:(nrow(sentence.list)-2)]
sentence.post=sentence.list$sentences[3:(nrow(sentence.list)-1)]
corpus.list$snipets=paste(sentence.pre, corpus.list$sentences, sentence.post, sep=" ")
rm.rows=(1:nrow(corpus.list))[corpus.list$sent.id==1]
rm.rows=c(rm.rows, rm.rows-1)
corpus.list=corpus.list[-rm.rows, ]
```

```
docs_p<- Corpus(VectorSource(corpus.list$snipets))
```

```
#remove potentially problematic symbols
docs_p<-tm_map(docs_p,content_transformer(tolower))
```

```
#remove punctuation
docs_p<- tm_map(docs_p, removePunctuation)


#Strip digits
docs_p<- tm_map(docs_p, removeNumbers)


#remove stopwords
docs_p <- tm_map(docs_p, removeWords, stopwords("english"))


#remove whitespace
docs_p<- tm_map(docs_p, stripWhitespace)


#Stem document
docs_p<- tm_map(docs_p,stemDocument)
```

Gengerate document-term matrices

```
dtm_p<- DocumentTermMatrix(docs_p)

rownames(dtm_p) <- paste(corpus.list$type, corpus.list$File,
                         corpus.list$Term, corpus.list$sent.id, sep="_")

rowTotals <- apply(dtm_p, 1, sum)

dtm_p  <- dtm_p[rowTotals> 0, ]
corpus.list=corpus.list[rowTotals>0, ]
```

Run LDA (Latent Dirichlet allocation)

```
burnin <- 4000
iter <- 2000
thin <- 500
seed <-list(2003,5,63,100001,765)
nstart <- 5
best <- TRUE


k <- 15

#Run LDA using Gibbs sampling
ldaOut_p<-LDA(dtm_p, k, method="Gibbs", control=list(nstart=nstart,
                                                seed = seed, best=best,
                                                burnin = burnin, iter = iter,
                                                thin=thin))
#write out results
#docs to topics
ldaOut.topics_p<- as.matrix(topics(ldaOut_p))
table(c(1:k, ldaOut.topics_p))
```

```
## 
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
```

```
## 39 51 32 42 35 43 26 29 28 30 38 27 37 18 19
```

```r
write.csv(ldaOut.topics_p,file="DocsToTopics.csv")

#top 6 terms in each topic
ldaOut.terms_p <- as.matrix(terms(ldaOut_p,20))
write.csv(ldaOut.terms_p,file="TopicsToTerms_p.csv")

#probabilities associated with each topic assignment
topicProbabilities_p <- as.data.frame(ldaOut_p@gamma)
write.csv(topicProbabilities_p,file="TopicProbabilities_p.csv")
```

```r
terms.beta_p=ldaOut_p@beta
terms.beta_p=scale(terms.beta_p)
topics.terms_p=NULL
for(i in 1:k){
  topics.terms_p=rbind(topics.terms_p, ldaOut_p@terms[order(terms.beta_p[i,], decreasing = TRUE)[1:7]])
}
```

Based on the most popular terms and the most salient terms for each topic, we assign a hashtag to each topic.We are going to use the hashtags that were developed in the tutorial, in order to have greater comparability.

```r
topics.hash=c("Economy", "America", "Defense", "Belief", "Election", "Patriotism", "Unity", "Government
corpus.list$ldatopic=as.vector(ldaOut.topics_p)
corpus.list$ldahash=topics.hash[ldaOut.topics_p]

colnames(topicProbabilities_p)=topics.hash
corpus.list.df=cbind(corpus.list, topicProbabilities_p)
```

For Wartime presidents

```r
corpus.list.w=sentence.list.w[2:(nrow(sentence.list.w)-1), ]
sentence.pre.w=sentence.list.w$sentences.w[1:(nrow(sentence.list.w)-2)]
sentence.post.w=sentence.list.w$sentences.w[3:(nrow(sentence.list.w)-1)]
corpus.list.w$snipets=paste(sentence.pre.w, corpus.list.w$sentences.w, sentence.post.w, sep=" ")
rm.rows=(1:nrow(corpus.list.w))[corpus.list.w$sent.id==1]
rm.rows=c(rm.rows, rm.rows-1)
corpus.list.w=corpus.list.w[-rm.rows, ]
```

```r
docs_w<- Corpus(VectorSource(corpus.list.w$snipets))

#remove potentially problematic symbols
docs_w<<-tm_map(docs_w,content_transformer(tolower))


#remove punctuation
docs_w<- tm_map(docs_w, removePunctuation)


#Strip digits
docs_w<- tm_map(docs_w, removeNumbers)


#remove stopwords
docs_w <- tm_map(docs_w, removeWords, stopwords("english"))
```

```r
#remove whitespace
docs_w<- tm_map(docs_w, stripWhitespace)


#Stem document
docs_w<- tm_map(docs_w,stemDocument)
```

Gengerate document-term matrices.

```r
dtm_w<- DocumentTermMatrix(docs_w)
#convert rownames to filenames#convert rownames to filenames
rownames(dtm_w) <- paste(corpus.list.w$type, corpus.list.w$File,
                         corpus.list.w$Term, corpus.list.w$sent.id, sep="_")

rowTotals_w <- apply(dtm_w, 1, sum) #Find the sum of words in each Document

dtm_w  <- dtm_w[rowTotals_w> 0, ]
corpus.list.w=corpus.list.w[rowTotals>0, ]
```

Run LDA (Latent Dirichlet allocation)

```r
#Set parameters for Gibbs sampling
burnin <- 4000    #drop the first 4000 samples
iter <- 2000
thin <- 500       #drop every 499 guesses and only pick every 500
seed <-list(2003,5,63,100001,765)  #setting a random seed because algo is random. imp for reproducibili
nstart <- 5
best <- TRUE  #sample with largest posterior distribution

#Number of topics
k <- 15 #need to come up on your own

#Run LDA using Gibbs sampling
ldaOut_w<-LDA(dtm_w, k, method="Gibbs", control=list(nstart=nstart,
                                            seed = seed, best=best,
                                            burnin = burnin, iter = iter,
                                            thin=thin))
#write out results
#docs to topics
ldaOut.topics_w<- as.matrix(topics(ldaOut_w))
table(c(1:k, ldaOut.topics_w))

##
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
## 57 52 41 35 45 57 22 38 20 42 38 33 28 51 30
write.csv(ldaOut.topics_w,file="DocsToTopics_w.csv")

#top 6 terms in each topic
ldaOut.terms_w <- as.matrix(terms(ldaOut_w,20))
write.csv(ldaOut.terms_w,file="TopicsToTerms_w.csv")

#probabilities associated with each topic assignment
topicProbabilities_w <- as.data.frame(ldaOut_w@gamma)
write.csv(topicProbabilities_w,file="TopicProbabilities_w.csv")
```

```
terms.beta_w=ldaOut_w@beta
terms.beta_w=scale(terms.beta_w)
topics.terms_w=NULL
for(i in 1:k){
  topics.terms_w=rbind(topics.terms_w, ldaOut_w@terms[order(terms.beta_w[i,], decreasing = TRUE)[1:7]])
}
```

```
topics.hash=c("Economy", "America", "Defense", "Belief", "Election", "Patriotism", "Unity", "Government"
corpus.list.w$ldatopic=as.vector(ldaOut.topics_w)
corpus.list.w$ldahash=topics.hash[ldaOut.topics_w]

colnames(topicProbabilities_w)=topics.hash
corpus.list.df.w=cbind(corpus.list.w, topicProbabilities_w)
```

**Topics in War Time Presidents' speeches**
```
par(mfrow=c(1,2))
wordcloud(corpus.list.df.w$ldahash,
          scale=c(2.4,0.2),
          max.words=100,
          min.freq=1,
          random.order=FALSE,
          rot.per=0.3,
          use.r.layout=T,
          random.color=FALSE,
          colors=brewer.pal(9,"Blues"))

wordcloud(corpus.list.df$ldahash,
          scale=c(2.4,0.2),
          max.words=100,
          min.freq=1,
          random.order=FALSE,
          rot.per=0.3,
          use.r.layout=T,
          random.color=FALSE,
          colors=brewer.pal(9,"Blues"))
```
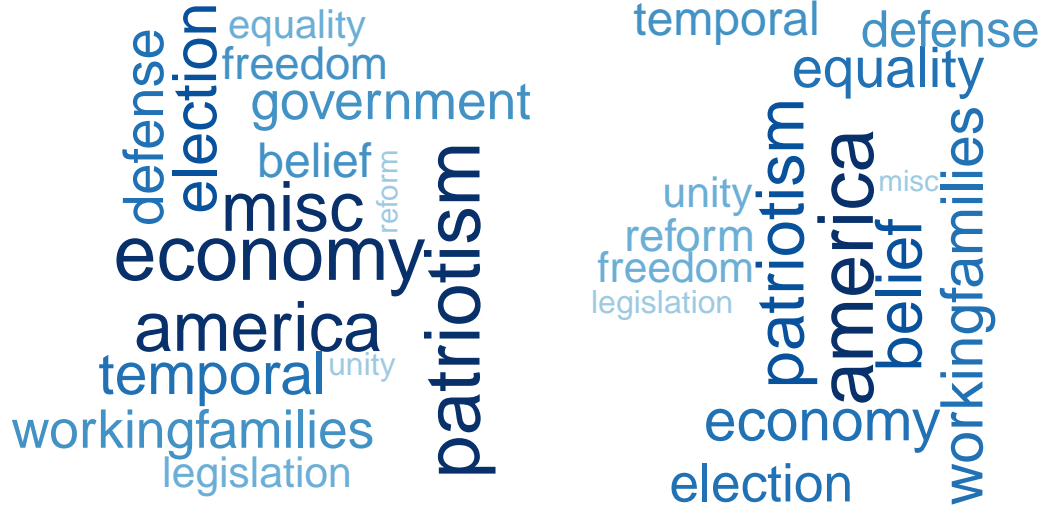
```
## Warning in wordcloud(corpus.list.df$ldahash, scale = c(2.4, 0.2), max.words
## = 100, : government could not be fit on page. It will not be plotted.
```

Looking at the most frequently used topics in speeches by both war-time presidents(left word cloud) and peace-time presidents(right word cloud), the war-time Presidents talked more about patriotism, economy,America and defense. On the other hand, the peace-time presidents have a more equitible distribution of topics in their speech. In addition to patriotism and America, they speak a lot more about reforms and beliefs as compared to the war-time presidents.

## 3. Conclusion

The literary theorist Kenneth Burke, in his work Grammar of Motives, argued that the precision of one's language reflects that individual's philosophy;for example,a perfectionist would use words that are free of any ambiguity. Therefore, understanding the text of the inaugural speeches is important in order to understand the course of action a President will take. From this study, we can conclude that the War-time presidents were more direct and their speeches contained shorter sentences with more negative tone. They spoke more about economy and nation while peace-time presidents spoke more about ideals. This study opens up another question for further research, which is, do certain kind of Presidents bring upon war?