R Notebook Data Story on US Presidents' Inaugural Speeches

Chengcheng Yuan

This is an R Notebook on the analysis of US presidents' inaugural speeches. It mainly focuses on the comparison and commonality wordcloud of the speeches based on the results of clustering of topic modeling and aims at figuring out the most popular words of presidents' speeches in each cluster. More detailed results have been provided about the exact frequency of words in each cluster including the word frequency plot and two csv files.

Step 0 - Install and load libraries

```
packages.used=c("tm", "wordcloud", "RColorBrewer", "reshape", "ggplot2",
                "dplyr")

# 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(reshape)
library(ggplot2)
```

This notebook was prepared with the following environmental settings.

```
print(R.version)
```

```
##                 _
## platform        x86_64-apple-darwin13.4.0
## arch            x86_64
## os              darwin13.4.0
## system          x86_64, darwin13.4.0
## status
## major           3
## minor           3.1
## year            2016
## month           06
## day             21
## svn rev         70800
## language        R
## version.string R version 3.3.1 (2016-06-21)
## nickname       Bug in Your Hair
```

Step 1 - Read in the speeches and the clustering results of top modeling

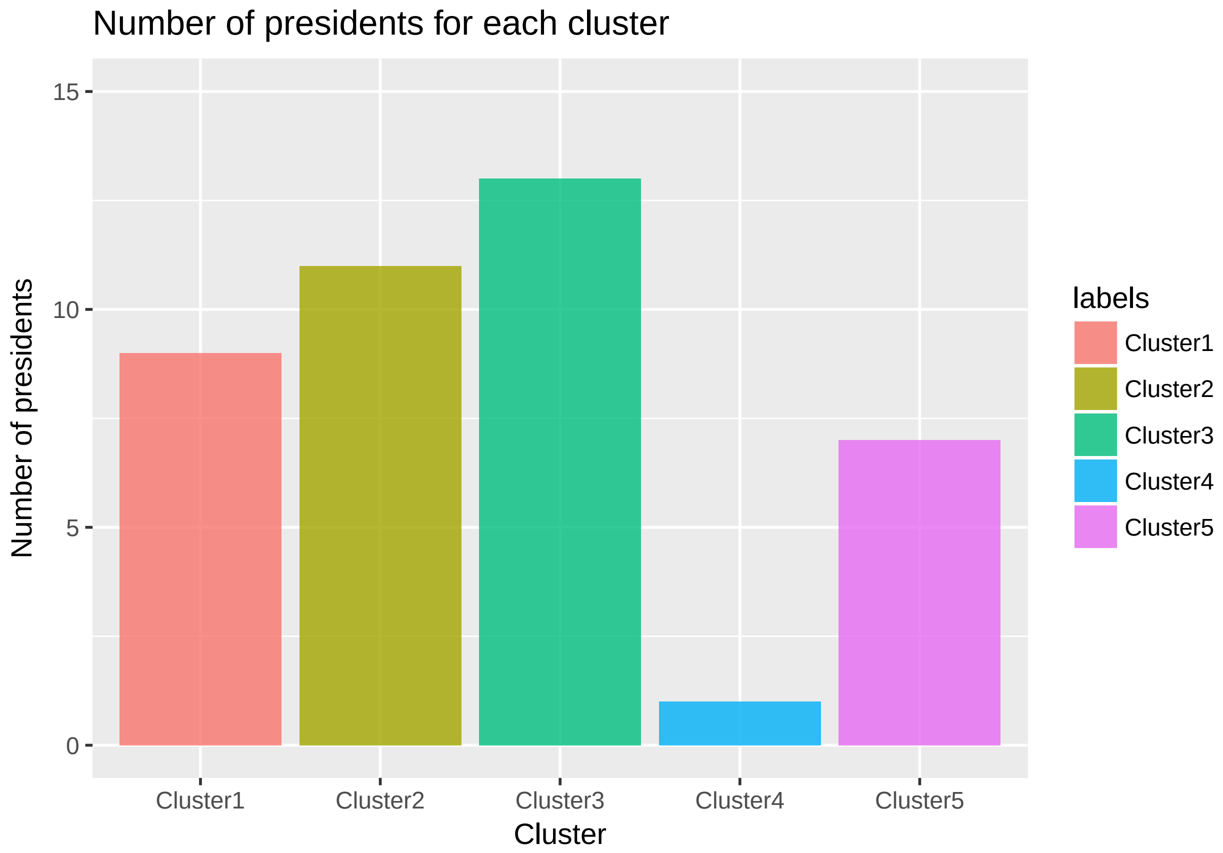
Based on the clustering results of topic modeling, we classify the speeches of presidents into five clusters in preparation of the following analysis.

```
currentwd <- getwd()
currentwd <- substr(currentwd,1,nchar(currentwd)-4)
relativepath<- "/data/InauguralSpeeches/"
folder.path <- paste(currentwd,relativepath,sep="")
speeches=list.files(path = folder.path, pattern = "*.txt")
clustering <- read.csv(paste(currentwd,"/data/cluster.csv",sep=""),header=F)
colnames(clustering) <- c("President","Cluster")
index <- substr(speeches,6,nchar(speeches)-6) %in% clustering[,1]
speeches <- speeches[index]
n <- length(speeches)
dataset <- c()
labels <- c()
for (i in 1:n){
  dataset[i] <- readLines(paste(folder.path,speeches[i],sep=""))
  labels[i] <- clustering$Cluster[clustering[,1]==substr(speeches[i],6,nchar(speeches[i])-6)]
}
labels <- unlist(labels)
dataset <- data.frame(dataset)
labels <- paste("Cluster",labels,sep="")
```

step 2 - Calculate the number of presidents in each cluster and draw a bar plot

```
pdf(paste(currentwd,"/output/Number of presidents each cluster.pdf",sep=""))
ggplot(as.data.frame(table(labels))) +
  aes(x=labels,y=Freq) +
  geom_bar(stat="identity",aes(fill=labels),alpha=0.8) +
  ylim(0,15) +
  labs(title="Number of presidents for each cluster",
       x="Cluster",y="Number of presidents")
dev.off()
```

```
ggplot(as.data.frame(table(labels))) +
  aes(x=labels,y=Freq) +
  geom_bar(stat="identity",aes(fill=labels),alpha=0.8) +
  ylim(0,15) +
  labs(title="Number of presidents for each cluster",
       x="Cluster",y="Number of presidents")
```



Step 3 - Text processing

For each speeches, we remove extra white space, numbers, punctuation and stop words. We also remove a word list which may not be of much importance and the words with length shorter than four. Then we compute the Document-Term Matrix (DTM).

```
unique_labels <- sort(unique(labels))
dataset_merge <- sapply(unique_labels,function(label) list( dataset[labels %in% label,1] ) )
ff <- lapply(dataset_merge, function(x) Corpus(VectorSource( toString(x) )))
ff_all <- ff[[1]]
for (i in 2:length(unique_labels)) {
  ff_all <- c(ff_all,ff[[i]])
}

ff_all <- tm_map(ff_all, removePunctuation)
ff_all <- tm_map(ff_all, removeNumbers)
ff_all <- tm_map(ff_all, function(x) removeWords(x,stopwords("english")))

words_to_remove <- c("said","from","what","told","over","more","other","have","last",
                     "with","this","that","such","when","been","says","will","also",
                     "where","why","would","today")
ff_all <- tm_map(ff_all, removeWords, words_to_remove)

dtm <- TermDocumentMatrix(ff_all)
dtm_mat <- as.matrix(dtm)
```

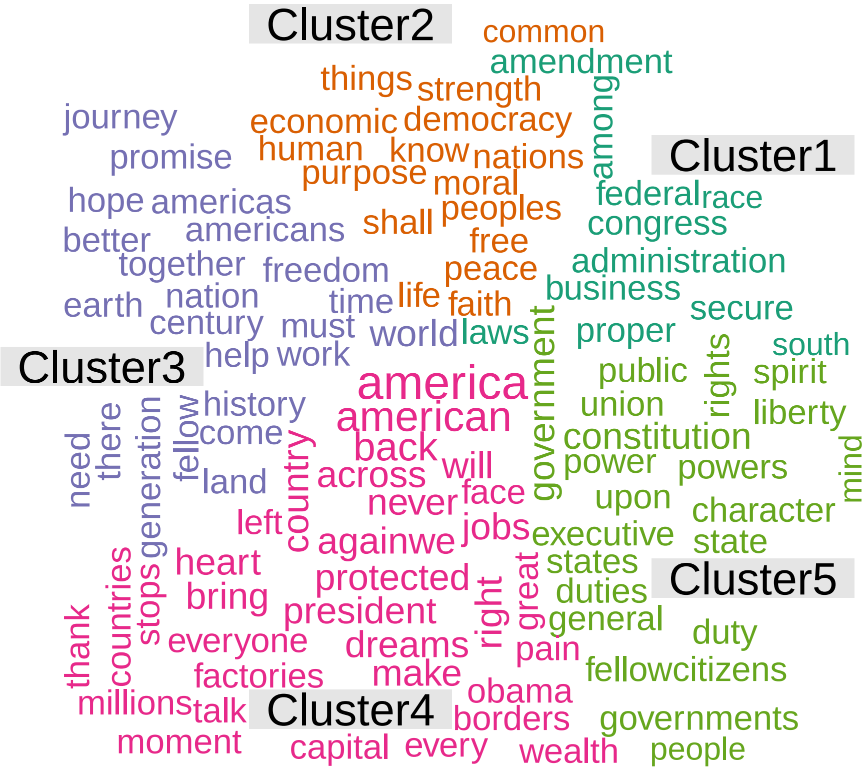
```r
colnames(dtm_mat) <- unique_labels
index <- as.logical(sapply(rownames(dtm_mat), function(x) (nchar(x)>3) ))
dtm_mat_long <- dtm_mat[index,]
```

Step 4 - Inspect a comparison wordcloud

Using the tools of wordcloud package, we can calculate a comparison wordcloud to show the difference of important words across the clusters. In this notebook, we give two results with maximum words of 200 and 2000 respectively.

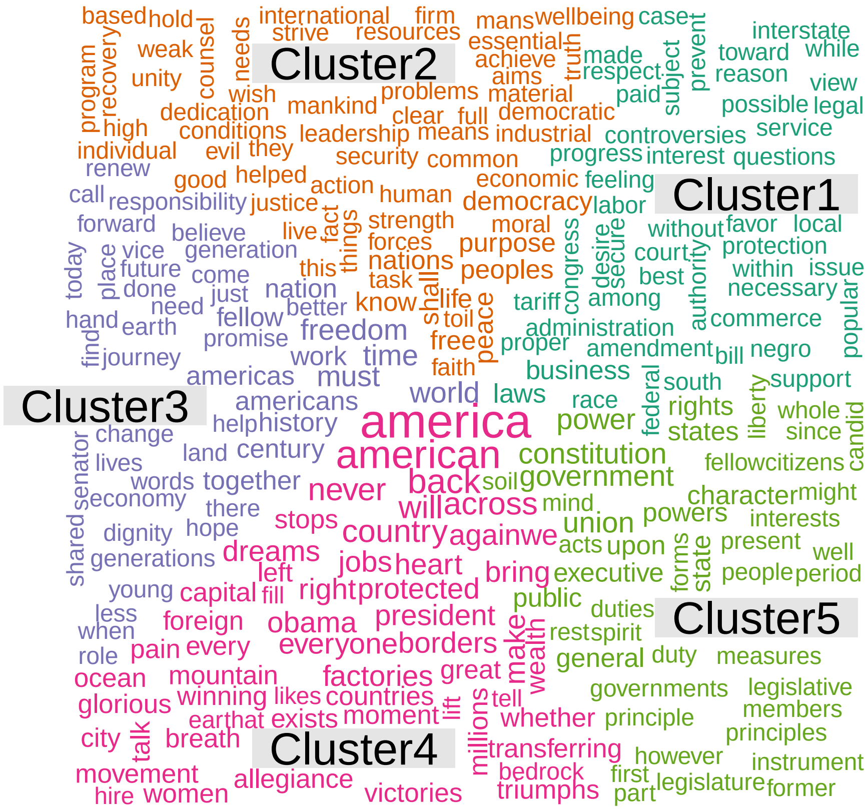
```r
pdf(paste(currentwd,"/output/Comparison_cloud_200.pdf",sep=""),height=8,width=8)
comparison.cloud(dtm_mat_long,
                 max.words=200,
                 random.order=FALSE,c(3,1.5),
                 title.size=1.4)
dev.off()
```

```r
comparison.cloud(dtm_mat_long,
                 max.words=200,
                 random.order=FALSE,c(1.5,1),
                 title.size=1.4)
```



```r
pdf(paste(currentwd,"/output/Comparison_cloud_2000.pdf",sep=""),height=8,width=8)
comparison.cloud(dtm_mat_long,
                 max.words=2000,
                 random.order=FALSE,c(4,0.8),
                 title.size=1.4)
dev.off()
```

```
comparison.cloud(dtm_mat_long,
                 max.words=2000,
                 random.order=FALSE,c(1.5,0.7),
                 title.size=1.4)
```

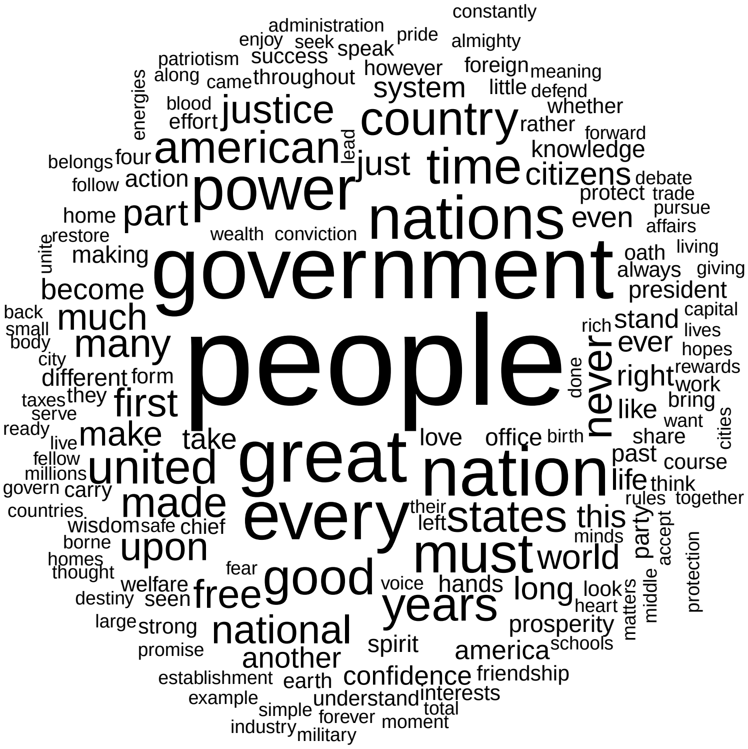


```
pdf(paste(currentwd, "/output/Commonality cloud.pdf",sep=""),height=5,width=5)
commonality.cloud(dtm_mat_long,
                  max.words=2000,
                  random.order=FALSE)
dev.off()
```

Step 5 - Inspect a commonality wordcloud

We calculate this wordcloud to show the top common words across the five clusters. The results is the following.

```
commonality.cloud(dtm_mat_long,
                  max.words=2000,
                  random.order=FALSE)
```



We can see from the result that the top common words presidents in every cluster seem to emphasis are 'people', 'government', 'nations' and so on.

Step 6 - Calculate the word frequency within each cluster

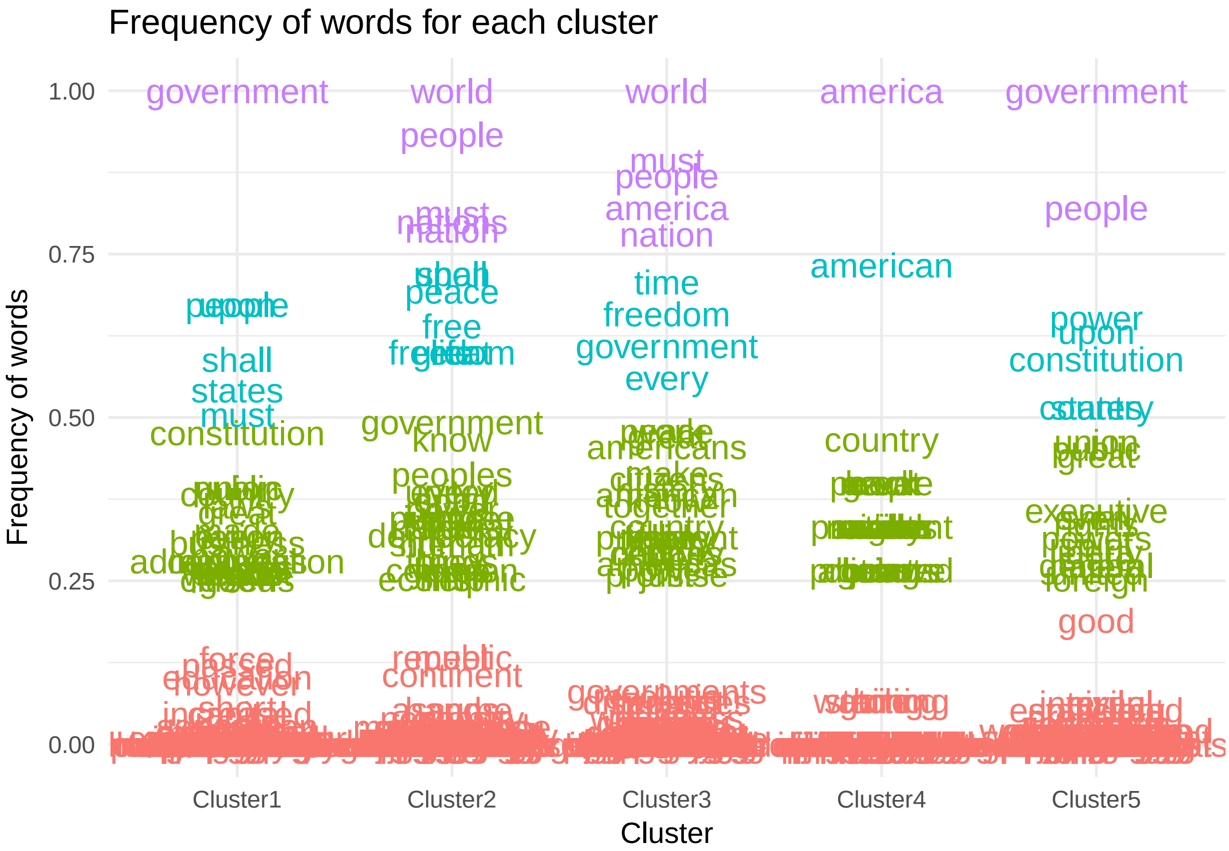
We set four frequency levels for this calculation which are 0.25, 0.5, 0.75 and 1. Since a large amount of words are of frequency between 0 and 0.25, we draw a random samples from this class for plotting the word frequency graph.

```
dtm_mat_long_norm <- apply(dtm_mat_long,2,function(col) col/max(col) )

word_frequency <- melt(dtm_mat_long_norm)
word_frequency <- cbind(word_frequency,
                    # add colors depending on the score
                    category=ifelse(word_frequency$value<=0.25,"0.25",
                    ifelse(word_frequency$value<=0.5, "0.5",
                    ifelse(word_frequency$value<=0.75, "0.75",
                    ifelse(word_frequency$value<=1, "1.0","lol"))))
)

index <- !word_frequency$category %in% "0.25"
word_frequency2 <- word_frequency[index,]
index <- sample(rownames(word_frequency[word_frequency$category %in% "0.25",]),
                500,replace=FALSE)
word_frequency2 <- rbind(word_frequency2,word_frequency[index,])
write.csv(word_frequency, file=paste(currentwd,"/output/Word frequency.csv",sep=""))
write.csv(word_frequency, file=paste(currentwd,"/output/Word frequency_simplified.csv",sep=""))
```

Step 7 - Plot the word frequency graph of each cluster

```
pdf(paste(currentwd, "/output/Frequency of words.pdf",sep=""),width=8,height=6)
ggplot(word_frequency2) +
  aes(x=Docs,y=value) +
  geom_text(aes(label=Terms,size=1,colour=category)) +
  labs(title="Frequency of words for each cluster",
       x="Cluster",y="Frequency of words") +
  theme_minimal() +
  theme(legend.position="none")
dev.off()
```

```
ggplot(word_frequency2) +
  aes(x=Docs,y=value) +
  geom_text(aes(label=Terms,size=1,colour=category)) +
  labs(title="Frequency of words for each cluster",
       x="Cluster",y="Frequency of words") +
  theme_minimal() +
  theme(legend.position="none")
```



From the frequency plot, we can see that Cluster4 which is consisted of the speech of Donald Trump mainly emphasise America, American and Country while the speeches of presidents in other four clusters are about government, world and people.