**Lab – Text Analysis**

1. **Primitive text visualizations**

* Quick and dirty word cloud: <http://www.wordle.net/>
* Type in “I love data visualization” and visualize it.
* Type in “I love love love love love love love data visualization visualization” and visualize it.
* Explore text visualization collection: <http://textvis.lnu.se>

1. **Text mining in R**

|  |
| --- |
| library(tm)  library(wordcloud)  #create some sample data  sample.tennis = c("Oh, I absolutely love Roger, even he lost to Djokovic in IndianWells Yesterday.",  "Tennis! It must be love!",  "USOpen tickets cost about 100 per ticket.",  "I love tennis I can not lie! ")  corp <- Corpus(VectorSource(sample.tennis))  dtm <- DocumentTermMatrix(corp)  inspect(dtm)  wordcloud(corp, min.freq=25, color=brewer.pal(6, "Dark2") )  **#plot word relations**  library(igraph)  tdm = as.matrix(tdm)  termMatrix <- tdm %\*% t(tdm)  g <- graph.adjacency(termMatrix, weighted=T, mode = "undirected")  plot(g)  **#Text processing**  corp <- tm\_map(corp, removePunctuation)  corp <- tm\_map(corp, removeNumbers)  corp <- tm\_map(corp, removeWords, stopwords("english"))  corp <- tm\_map(corp, stripWhitespace)  library(SnowballC)  corp <- tm\_map(corp, stemDocument)  corp <- tm\_map(corp, content\_transformer(tolower))  #corp <- tm\_map(corp, PlainTextDocument) #avoid tm 0.6 issue  #remove your OWN stopwords  myStopwords <- c("can")  myCorpus <- tm\_map(corpus, removeWords, myStopwords)  wordcloud(myCorpus, colors=brewer.pal(5,"Set1"),random.order=FALSE, max.words=50)  **#now, load trump\_debate.txt & clinton\_debate.txt**  **#process both corpora and generate two wordcloud figures.**  **#generate topic models based the debate data, and generate wordclouds again. Would you observe the differences between these two worldclouds?**  library(lda)  doclines <- lexicalize(corpus)  result <- lda.collapsed.gibbs.sampler(doclines$documents, 10, doclines$vocab,  250, 0.1, 0.1, compute.log.likelihood = TRUE)  cloud.data <- sort(result$topics[1, ], decreasing = TRUE)[1:50]  wordcloud(names(cloud.data), freq = cloud.data, scale = c(4, 0.1), min.freq = 1,  rot.per = 0, random.order = FALSE,colors=brewer.pal(5,"Set1"))  **#Bonus (2 pt for assignment4, submit it with assignment 4): how do you generate both candidates debate word relation network figures?**  **MacOS:private:var:folders:zs:b7tvvzn971g14jwvjskttx4jz5czbg:T:TemporaryItems:strimling-attacks.png** |

1. **Sentiment analysis**

* Download dictionaries from <http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html> ([A list of positive and negative opinion words or sentiment words for English](http://www.cs.uic.edu/~liub/FBS/opinion-lexicon-English.rar) (around 6800 words)) We will add our own words later!
* Import Sentiment.R

|  |
| --- |
| #load sentiment score function (Sentiment.R)  #load lexicon: positive & negative words into R  pos = scan('opinion-lexicon-English/positive-words.txt', what='character', comment.char=';')  neg = scan('opinion-lexicon-English/negative-words.txt', what='character', comment.char=';')  #customized dictionary (optional)  pos.words = c(pos, 'upgrade','lol')  neg.words = c(neg, 'wtf', 'wait', 'waiting', 'epicfail', 'mechanical')  tennis.sentiment = score.sentiment(sample.tennis, pos.words, neg.words)  tennis.sentiment$score |

**Questions: (Submit text-vis-lab by the end of 3/24 (Tuesday))**

Q1. How do you compare word clouds?

Q2. What types of visualizations are good for showing polarities?

Q3. What are both candidates’ first debate sentiment scores?