Lab #4: Named Entity Recognition and Part-of-Speech Tagging

Due in Sakai February 22, 2019 (by 10:00a)

Introduction

For this lab, you will practice using named entity recognition (NER) and part-of-speech (POS) classifiers.

Getting Started

First, set your working directory. Open up a fresh RStudio session (if you open RStudio and a previous work session is loaded, be sure you save any R and/or RData files before you close them out). Then set your working directory:

```
setwd("working_directory_here")
```

Then open up a fresh R script. Save it using your first initial, full last name, and then "_lab4." So my script would be titled **MTaylor_lab4.R**.

You should always strive to keep your scripts tidy. At the top of your R script, type this (substituting in your first initial and last name):

```
##############################
##  MTaylor_lab4.R
##  Note: Code for lab assignment #4
##  Author: Marshall A. Taylor
##############################

###BEGIN###
```

You are ready to begin your code. Be sure to include all the code necessary for me to check your work. Document your code thoroughly (using "#").

Remember that saving your R work is a two-step process. You save your R script using Cmd+Enter (Mac) or Cntrl+Enter (Windows). You save your R data objects like this:

```
save.image("MTaylor_lab4.RData")
```

Lastly, prep a Word, Pages, or LaTeX document that has the same title structure: e.g., **MTaylor_lab4.docx**. This is where you put your write-ups and visualizations.

You should turn in three documents to Sakai: your **R script** that shows the code you used, **RData file** that provides the data your scraped, and **Word document** (or whatever text processor you choose to use) with your write-ups.

Assignment

There are two components to this lab. First, you will identify the "entities"" in a corpus and classify them. Second, you will classify the parts of speech in that same corpus. Recall that the packages we are using for NER and POS tagging are built around the OpenNLP library (The Apache Software Foundation 2017),

which is Java-based. So if you cannot get Java to play nice with R on your personal machine (most likely for Mac users), you may need to complete the lab on a campus PC. Plan your time accordingly.

Section #1: Named Entity Recognition

For this section, you'll apply the NER classifier to the provided corpus. The corpus is a sample of the State of the Union dataset (The American Presidency Project 2018). I have included only those SOTUs from 2000 through 2018.

The corpus file is called "sotu_corpus.rds." You can read in the corpus using the `readRDS()` function.

1. Apply the "person" tagger separately to two subsets of the corpus: speeches given by Democratic versus Republican presidents. Get the frequency distributions of the person entities for each subset, setting the minimum frequency for any given person entity to be retained in the table to two mentions. Plot the results in side-by-side bar graphs, with the entities in descending order of frequency. Save the graph as a PDF and put it in your Word document.
2. What are some interesting conclusions you might draw from this graph?
3. Repeat step #1, this time using the "organization" tagger.
4. What are some interesting conclusions you might draw from this graph?
5. Are there any entities included in either the person or organization frequency tables that you think reflect a classification error? If so, are there any that you think you might now why it was misclassified as a person/organization?
6. What are the most distinguishing organization tags by political party? Create a plot to visualize the distinguishing tags.

Section #2: Part-of-Speech Tagging

Now let's do some POS tagging using the same data.

1. Apply the POS tagger—using the "basic" categories—separately to the Democrat and Republican speeches Get the frequency distributions of the POS tags for each subset. Plot the results in side-by-side bar graphs, with the tags in descending order of frequency. Save the graph as a PDF and put it in your Word document.
2. What are some interesting conclusions you might draw from this graph?
3. What proportion of total words in the Democratic speeches are nouns, and what proportion are verbs? What about for the Republican speeches?
4. Extra Credit: What are the most common nouns in the Democrat speeches? What about the Republican speeches? Creat a side-by-side bar graph that shows the most frequent nouns in descending order of frequency. Save the graph as a PDF and put it in your Word document.
5. Extra Credit: What are the most common verb lemmas in the Democrat speeches? What about the Republican speeches? Creat a side-by-side bar graph that shows the most frequent verb lemmas in descending order of frequency. Save the graph as a PDF and put it in your Word document. Why might it be more beneficial to use verb lemmas here instead of the raw verbs?

Hints

The R scripts for the dictionary and sentiment analysis slides (pres51.R and pres52.R) might be very helpful. Like, *really really helpful*.

References

The American Presidency Project. 2018. *Annual Messages to Congress on the State of the Union (Washington 1790 - Trump 2018)*. Retrieved February 3, 2019 (link).

The Apache Software Foundation. 2017. *openNLP*. Retrieved February 11, 2019 (link).