Week 4 Homework: Data and Text

Vincent Porretta

9/24/2019

This assignment uses data that were scraped from a browser extension that attempts to classify online articles as fake news or not. For this assignment you will use two separate files: `FakeNewsMeta.cvs` (which contains meta data for the posts) and `FakeNewsTitles.txt` (which contains the raw text of the article titles. Files available on Avenue to Learn. Record your completion of the steps of this assignment in an R scipt. Use comments to add additional information for answering the questions. 36 points. Due by the start of class on 10/1/2019.

Data

1. Read in `FakeNewsMeta.cvs` to a data frame called `Meta` and examine the contents.
2. Create a new column that represents only the date on which the article was published. Hint: similar to Part 2, Step 3 on the in-class exercise.
3. Find the average number of shares for each classification type. Which was the highest?
4. Find the number of articles per author. Who were the top 10 most prodigious posters?
5. Find each author's number of post per day. Exclude those posts with an empty author field (Hint: strings that only have a beginning and an end). Arrange the authors from post frequent poster to least. Save this to a new object called `PPD`. Who were the ten posters with the highest posting frequency?
6. `Eddy Lavine` was among the top posters and had some of the fastest posting speed. Select his data from PPD and examine his frequency over time. When did his peak posting frequency occur?
7. Using the information contained in `Meta`, find the country Eddy was posting from.
8. Knowing what we now know about Eddy's peak posting frequency (as determined above) and location (google his country code), what event might this correspond with?

Text

9. Read in `FakeNewsTitles.cvs` by lines to a vector called `Titles`. Please be sure to specify `fileEncoding = "UTF8"`. Examine the contents.
10. Remove the following using regex:

a) all text between | or » and the end of the string (regex: `"(\\||»).*$"`)
b) any text within parentheses or square brackets (regex: `"(\\[|\\()[a-zA-Z\\s]*(\\]|\\))"`)
c) any url ending in .com that appears at the end of the string (regex: `"\\s[a-zA-Z0-9]*\\.com$"`)

Note that (a) and (b) may be completed in multiple steps

11. Complete the following:

a) Convert to lowercase
b) Replace all instances of `...` with a space (note that this is a single UTF8 character, not three periods!)
c) Remove all punctuation
d) Remove excess whitespace (both around and within strings)

Note that (d) may be completed in multiple steps

12. Split `Titles` to a vector of individual words called `Titles_words`