Bot or Not Assignment

Applied Computing Assignment: 1 point

Due: Start of class **this Thursday**, April 4. Over Google Forms; see below.

For this assignment, you'll be building and analyzing the computational part of a bot by automatically generating text; then you'll be trying to tell apart bot-created text from human-generated text. This involves doing two things drawing on core skills we have emphasized this class.

1. Thinking *creatively* about finding sources to be the *input* to computation.
2. Thinking *critically* about sources that someone else has created as the *output* of computation.

Steps

We'll be using a Markov chain to write these bots. R code is below.

Making

1. **Choose a source** as input to the Markov chain; download it, and save it as a text file. (Or just pull it from the web if it's a text file there).
2. Run it through the Markov chain generator several times until you get a sentence or two you like. Always do this changing the random seed.
3. Choose *both* a bot-like snippet from your original text, and a non-bot-like snippet you've generated, and submit them via the form at https://docs.google.com/forms/d/169GzC-QJmvdBchzHiAk3TqKt8tIomPzuo-idR3-previewform You do not need to use an entire snippet generated; you can start at word 5 and end at word 28 of a 40 word run, for instance. For the real segment, you should be choosing something that actually exists in the original text you used, that you think is likely to fool your classmates as something that might have been produced by a Markov chain.

Reading

Once the bots are up, I'll put a website where you can guess *bot or not*; we'll collect statistics on which ones do the best in class Thursday.

Grading

This is a two-step grading process.

1. For *submission*, you'll receive an A if you include all elements in your online submission, including some evidence you put thought into the source: or a B if it's incomplete.

2. In class on Thursday, your attempts to evaluate other students's efforts will count as a reading response. If you confuse the most other students, you'll get bragging rights.

Appendix 1: Markov Generator Generator code.

This is a program that creates a Markov chain generator from a text file.

```
library(tidytext)
library(tidyverse)

markov_generator_generator = function(filename, order) {

  labels = str_c("word", 1:order, sep = "")

  transitions =
    tibble(text = read_lines(filename)) %>%
      filter(!is.na(text)) %>%
      unnest_tokens(ngram, text, token = "ngrams", n = order, to_lower = F) %>%
      separate(ngram, labels, " ") %>%
      # This is some special syntax for grouping. See ?syms for an explanation.
      count(!!!syms(labels))

  add_rows = function(frame, length) {
    next_row = frame %>%
      tail(1) %>%
     inner_join(transitions, by = intersect(names(frame), names(transitions))) %>%
      sample_n(1, weight = n) %>%
      select(-word1, -n)
    names(next_row) = labels[1:order - 1]
    output = bind_rows(frame, next_row)
    if (length > 1) {
      return(add_rows(output, length - 1))
    } else {
      return(output)
    }
  }

  generator = function(..., length, seed) {
    set.seed(seed)

    words = unlist(list(...))
    data_frame(col = paste("word", 1:length(words), sep = ""), word = words) %>%
      spread(col, word) %>%
      add_rows(length)
  }
}
```

To use it, you'll have to change just a few things: in the line below, both the filename (don't use Henry Adams!)

and the depth (order=3 here means to create a level three Markov chain; use the last two words to predict the next 1 word.)

As part of your submission, copy the version of this code you end up using; it need only have two lines like below.

`seed` makes the randomness deterministic; you'll have to change the number for every run. `The` is just a random starting word; you can change it to anything else, or make it a longer list if you like.

```
generator = markov_generator_generator("http://www.gutenberg.org/files/2044/2044.txt", order = 2)

generator("The", length = 45, seed = 20) %>% pull(word1) %>% paste(collapse = " ")
```