Recreating Literary Author's Styles With Markov Chains

Daniel Olivieri With Professor Chun and Professor Elkins
Artificial Intelligence for the Humanities

Motivation

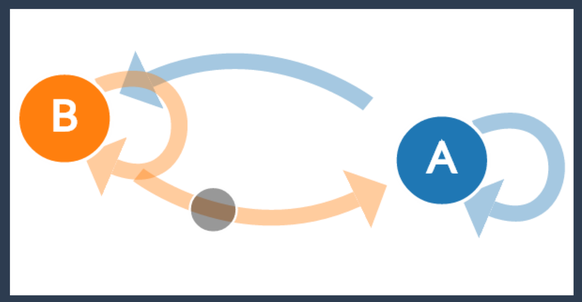
Style is an essential part of literature. What words a writer uses, how many adjectives and adverbs, their punctuation, are all important facets to great writing. Thus, to capture and recreate this style marks a significant achievement in artificial writing ability as it shows some level of proficiency at imitation. In this project, we wrote a program that uses Markov chains to reproduce the style of various authors. We then used Twitter's API to have the program post its generated sentences online in real time.

Markov Chains

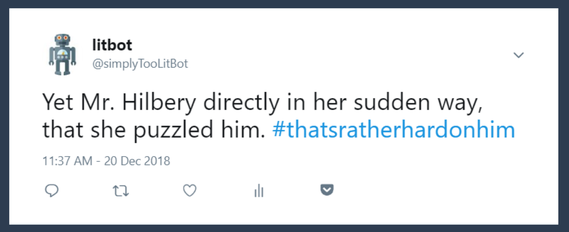
The most important underlying technological method for this project is the Markov chain. Markov chains are a series of stochastic decisions in which the chance of each option being chosen is not based on past occurrences, but solely on the current state of the Markov chain.[4] This memoryless aspect that defines how Markov Chains operate is especially interesting when one is trying to produce text. Not taking into account what's been said previously in the sentence means that few of the generated sentences will make logical sense. However, the Markov chains' ability to find and reproduce patterns of which words follow one another allow it to effectively mimic an author's style, even if the content rarely makes sense. Markov chains have been used to better understand international politics,[3] make musical compositions,[1] and understand sociology.[5]

Choice of Twitter as a Platform

As mentioned in this poster's section on Markov Chains, while it is capable of reproducing style, the method I am using rarely produces text that has any clear meaning. Were the text to go on for a paragraph it would read like an incoherent jumble of unrelated sentences. Thus, it is important to apply this to a platform that does not require full paragraphs. With its limit of just 280 characters, Twitter was the perfect environment for this project. It allows readers to enjoy the imitation of an author's style without requiring long passages of text.



This is a visual representation f the stochastic process of the Markov chain.



This is a tweet based on *Pride and Prejudice.*

Methodology

We chose to work in the Python language because it had a robust number of libraries for working with text. We start by using the urllib python library to send a request to Project Gutenberg for the text file we are interested in. For example, Jane Austen's *Pride and Prejudice*. In some cases, the data are not available on Project Gutenberg and the text is instead saved in the project's folder. These are accessed through the file system. Once we have imported the data, we then decode the response using the UTF8 character encoding.We then use the Markovify library[7] to instantiate a Markov chain model object based on the text to be imitated, in this case *Pride and Prejudice*. We use the "make_short_sentence" function of this object to create the main Tweet. We pass the integer argument "180" to the function so that it generates text that is no more than 180 characters. To make the hashtag, we then call this function again, this time limiting it to only 30 characters and removing all punctuation and spaces afterwards and adding an octothorpe at the beginning. The next step is to use Twitter's Tweepy API to post the completed Tweet.

Conclusion

I feel that this Twitterbot represents a significant advancement over other similar projects. There have been several attempts to make literary twitter accounts using Markov chains. Such attempts include the Murkami generator and the James Joyce generator.[11] These previous attempts have been admirable, but have lacked the ability to make hashtags for their tweets. We found this to be a major oversight on their part, as the use of hashtags is a defining feature of the Tweet format. By using the Markovify library to generate short strings, we could then combine the words to allow for hashtags. Thus, this project now adds a new stochastically generated hashtag to each one of its tweets.

References

1. Ames, Charles. "The Markov Process as a Compositional Model: A Survey and Tutorial." Leonardo, vol. 22, no. 2, 1989, pp. 175–187. JSTOR, JSTOR, www.jstor.org/stable/1575226.
2. Duncan, George T., and Randolph M. Siverson. "Markov Chain Models for Conflict Analysis: Results from Sino-Indian Relations, 1959-1964." International Studies Quarterly, vol. 19, no. 3, 1975, pp. 344–374. JSTOR, JSTOR, www.jstor.org/stable/2600315.
3. Gagniuc, Paul A. (2017). Markov Chains: From Theory to Implementation and Experimentation. USA, NJ: John Wiley & Sons. pp. 1–235. ISBN 978-1-119-38755-8.
4. McFarland, David D. "Intragenerational Social Mobility as a Markov Process: Including a Time- Stationary Mark-Ovian Model That Explains Observed Declines in Mobility Rates Over Time." American Sociological Review, vol. 35, no. 3, 1970, pp. 463–476. JSTOR, JSTOR, www.jstor.org/stable/2092989.
5. Robert, Christian, and George Casella. "A Short History of Markov Chain Monte Carlo: Subjective Recollections from Incomplete Data." Statistical Science, vol. 26, no. 1, 2011, pp. 102–115. JSTOR, JSTOR, www.jstor.org/stable/23059158.
6. Singer-Vine, Jeremy. Markovify, (2013), GitHub repository, https://github.com/jsvine/markovify