Chloe Grubb
Software Design
Fall 2016

Alice in Wonderland Text Synthesizer

For this project, I decided to develop a generative text model based off Markov Text Analysis. This is a fairly well established approach to analyzing the style of text and using a random generation that mimics the writer's style. For this I used access to Project Gutenberg texts, particularly Charles Dodgson's Alice in Wonderland, developed a Python script to perform Markov Analysis, and then generate a new text based on it.

My Markov script has six functions in it, beginning with skipping the header of the Project Gutenberg file. Without this in place, it would also analyze the informative section for Project Gutenberg included in all of their files. I did not actually include this at first, and soon realized that my program was including the words and style that did not quite match up with those from Dodgson. It then begins processing each word individually, at first, it just begins storing the words, and then it begins forming a dictionary of them. This includes creating a map of possible suffixes to use.

Now that there the piece is processes, we can then begin using it to generate a new text, of "n" length. It starts by creating a random prefix from the stored dictionary, and then uses that as a baseline. Random words are then generated to follow it. This is a simple approach to Markov analysis and because it is purely random, the generated text does not always make grammatical sense. For example:

"It's the most confusing thing I ever heard! These were the cook, to see if there are, nobody attends to them and you've no idea what you're at! While the Duchess said in a few things indeed were really impossible. Exactly so, said Alice. Soles and eels, of course, the Mock Turtle interrupted, if you only kept on puzzling about it in less than a mile high, said Alice. That's different from what I used to say; once tasted but checked herself hastily, and said what else had you to."

"Its tongue hanging out of the party were placed along the passage into the roof bear? Mind that loose slate. Oh, it's coming down! Heads below! (a loud crash) Now, who did that? It was Bill, the Lizard. She could not possibly reach it: she could not be used. However, if you hold it too long; and that is rather a handsome pig, I think. And she squeezed herself up and beg for its dinner."

The results are entertaining, and retain some of the whimsical nature of Alice in Wonderland. On average, it produces a clean paragraph as a result; however, I had some issues with it producing random characters between letters of a single word. Overall, I was pleased with how the program turned out and what is generates.

However, if I were to revisit this project I would like to delve a little farther into what is causing the random errors. My best guess is that there is some remainder of the Project Gutenberg watermarking that I did not filter out entirely. I would also look into smarter algorithms for Markov Analysis so that I could generate text that makes more sense. These were not things I had the time or skill set to do this time around. That being said, I feel like my project was reasonably scoped and proportioned for time. I also had to do quite a bit of debugging throughout, which will help me moving forward with future SoftDes endeavors.