Queenie Guan
Professor Li
MIS3640 Problem Solving & Software Design
25 March 2020

**Assignment Two: Text-mining Project**

## I. Project Overview

I gained my data input from the ten most frequently downloaded Mark Twain books from Project Gutenberg, and fed them into a Markov text synthesis program to create new texts written in Mark Twain style.

## II. Implementation

Considering that I am using a large amount of texts, I decided to download all the text files from the website and store it in the project folder instead of fetching them off the Project Gutenberg's server every time I run the program. The original text files are stored in the "Mark Twain" folder.

In the "Markov analysis" folder, there are two python files and two corresponding text files as the output for each. The two major steps in this project are the following:

First, the "MT text.py" program processes all ten Project Gutenberg text files and combines them into one text file "Mark Twain.txt". The "process_file" function excludes everything that is not the official book content, unifies capitalization, processes punctuations, and returns the plain texts in a string format. Then, the "combine_text" function takes in a string tuple of processed text strings, and writes them into one text file "Mark Twain.txt," which will be the source data for the following Markov analysis.

Second, the "Markov.py" program creates a Markov analysis model to generate new texts. It first used the "make_dict" function to create a dictionary with all leading words, and the words appear next to it with their frequency. Then, with any word in mind, we can start to generate a new text. With every current word, we will randomly retrieve the next word from all the words appeared next to the word before based on the probability. After it generates the new text chain, the function "format_chain" will format the chain and return a reader friendly formatted string.

## III. Results

I spent a lot of time in improving the program so that it does a pretty good job in processing Gutenberg file and returning an easy-to-read text. However, the results are not as good as I expected. Occasionally, some sentences would make sense, but most of the time they don't. Sentences might make sense individually but there is no logic between

sentences at all. Here are a few examples produced by running the mash-up of ten books, starting with “he” and “she” **(**I bolded the part where I think the sentence makes sense):

> **He could not breathe not, and shiz, came across its unfolding his heels through the intrusion.** We could a storm cost me a hundred pounds of the stove if i would have been any a low voice “perchance he was returning from me. Money, to, too.
>
> **She had been dead silence,** he was just try to comprehend what my son is this man, just out of it down horse has ever since the child be at the “two minutes she was telling his business principles of them were making a fine porphyry pillars among casual as a pickaxe!

I thought that might because there’re a lot of differences among the topics of the ten books, hence mashing up those content don’t really produce meaningful new content. Thus, I ran another Markov analysis with just two books: *Adventures of Huckleberry Finn* and *The Adventures of Tom Sawyer*, since they are in the same series. Here are some results:

> Tom believed it ain't it 'cuz he could lock shows how companionless and hugged him.. **Tom found the united states, gold, but it was bright and offered entertainment and so he makes you i'll bless you!** She sews it looked down upon the back.
>
> **Huckleberry finn is what i would you fetch an indian territory robbers!** You in we want any more to comb, and don't you told tom hesitated and chained his head sore toe; and flung us. Then the deef and was as you don't come…
>
> Aunt polly was partly. The way to every day a blazing stick to cross, **he wrote the service proceeded to death, because nobody forbade that the road all that night.** Goshen's ten seconds he would i hear it was broad and whispered tom arrived. There and the towhead, and all intending to get a stone, did.
>
> **She was afraid to make him out after breakfast!** To her face lighted up and knees and then fell to foot long out against one, pretty reasonable, that belonged to make 'm all they said: “well, the wolves away she had gone? ” “i reckon i went out something like he was about the shore some in de yuther servants in a chicken bone, and talking about him understand it hid, but a snake up and jim put off; and never had no time in a monstrous bows to camp fire upon a pretty soon back a bad luck, with a lot of the new interest emphasized by the town to his mouth against ghosts.

The results are slightly better that it feels like sentences can somehow connected with each other. However, they still don’t make sense most of the time.

## IV. Reflection

In general, I think the program is working pretty well and perform its job. What effects more is the data that we feed into the Markov model. I think the results would be more

interesting with some other types of data source, for example, Donald Trump's tweets (I didn't used that because twitter declined my developer account application for not submitting the additional information they asked for one year ago when I applied for BI class, and there's no option to appeal their decision). I also tried to find other series of books on Project Gutenberg, like *A Song of Ice and Fire*, *The Hunger Game*, or *Harry Porter*, but unfortunately those are unavailable. I think if I understand the Markov text generator model better, I would have make a better decision in choosing the data source.

Our team decided that we will stay as a team to proceed with the project. We plan to start a bit later as we learn web page and APIs during class so that we have a better idea of what we're doing.