Homework 2 - Due February 15th, 2021 at 11:59 Eastern Standard Time

How to do this assignment: Using Google Colab (https://colab.research.google.com/) please answer the following questions in a new notebook. You are to write the question on a Text field, and your programming answer on a Code field. Please write your name and student ID number on a Text field at the very beginning of this notebook. Note that when you are testing your code, the output needs to be shown

What to submit: Via iCollege you are to download a copy of the executed Colab notebook in PDF format, and submit this file ALONGSIDE a link (in the comments/description section) to your own GitHub repository where this Colab notebook is stored. In other words, you are expected to keep your own personal GitHub repository for this class and place ALL your assignments and project code in that location.

Questions (150 total points): ***NOTE: You can only use Python's built-in regular functions, scikit-learn as your ML library for these exercises, NLTK as your NLP library. Any code using any other ML or NLP libraries will not be graded. Note: You can use pandas and/or numpy as these are NOT machine learning libraries.***

TEXT: Using all of Shakespeare's works found in iCollege titled: shakespeares-works_TXT_FolgerShakespeare.zip. Write the following pieces of code:

1. Create a cosine similarity matrix for all of Shakespeare's works found in the provided file. This will result in a 42 by 42 matrix with the cosine similarity between each of his works. In other words, calculate the document-wise cosine similarity between all of Shakepeare's works. (50 points). ***Use TF_IDF for this. Note, you can use the Cosine Similarity function on scikit-learn or implement your own, but no other library/package is allowed.***

2. Write a function that takes the previous matrix and a number *n* as parameters (nothing else will be accepted) and return the top *n* similar works. Use the function to output the top 10 similar works. (30 points).

3. Using the code from the Language Models II class, train two simple language models using all of the files (together) in shakespeares-works_TXT_FolgerShakespeare.zip. One model should be trained using bigrams, the other using trigrams. (40 points).

4. Write a function that takes the following three parameters: model, list of start words, number of sentences to generate. This function should return the sentences generated as a list. DO NOT print anything to the screen from within the function. Use this function to generate 10 sentences with the bigram model from the previous question, and 5 sentences with the trigram model from the previous question. (30 points)

Bonus (20 points): Using the same methodology from questions 1 and 2, create a similarity matrix between the 20 newsgroups corpus. And find the top 5 similar newsgroups.