Assignment 1: Text analysis, processing and representation

Submission process: Do this assignment in your preferred notebook (Jupyter, Colab any will do) and submit the notebook after you run all the blocks. Name the notebook using only your student ID-- no other words or characters (this is mandatory and I will deduct marks if you do not do it). You will have one chance to submit, so don't make a mistake. Your submission deadline is **December 27, 2024, 11:59pm** and no extension will be given whatsoever.
Submission link: https://forms.gle/No1hBgzNFkJugsqcA You will need to be logged in using your bracu email. Do not include the IMDB file or the GloVe file (they are way too big).

Preamble: Setting up environments for this assignment

Install NLTK: pip install nltk in terminal; OR !{sys.executable} -m pip install numpy in notebook
Download NLTK data:

1. From terminal or notebook: python -m nltk.downloader popular
2. Using python script: import nltk; nltk.download('popular')

Question 1: Analyzing word occurrences
To download Moby Dick, follow these steps:

- Download books (that we will use to analyze)
- From nltk.book import *
- After you do this, variable *text1* will contain the entire novella Moby Dick in it.

Now answer these questions:

a. What is the vocabulary size for Moby Dick? That is, how many unique words are in Moby Dick? [1]

b. How many total words are there in Moby Dick? [1]

c. Count the number of times a word is present in the novella. Create a python dictionary where the key will be a word and the value will be the number of times that word appeared in Moby Dick. We will consider punctuation marks as words here. [1]

d. List top 10 most frequent words based on what you found in C. [1]

e. List top 10 most frequent words that are not punctuation marks. [1]

Question 2: Loading, cleaning and processing text files (5 marks)

- Install Pandas: pip install pandas in terminal
- Download this file: https://drive.google.com/file/d/1dbwZBzlFQnuc1tqv0lXvBj9BqTJ_aZMe/view?usp=drive_link

- Unzip the file imdb_440.zip. It will contain the actual data file in .csv format. This is a set of reviews from IMDB.

Now answer these questions:

a. Load the csv file using pandas. How many reviews do we have here? Which row has the longest review in terms of words? How about in terms of sentences? [2]
   i. Tips: Search on google how to use NLTK for sentence tokenization and word tokenization. Specifically, how to use NLTK sent_tokenize and word_tokenize

b. Cleaning step 1: Now, from each review, remove stopwords and punctuations [2]
   i. Tips: Stop words are those words that do not contribute to the deeper meaning of the phrase. For example, articles. It is often necessary to remove these words during the text processing period. Search on google how to remove stopwords and punctuations effectively. There can be multiple ways, as long as it works, it is ok.

c. Cleaning step 2: Convert the reviews to lowercase and save these clean reviews (only the reviews, not their classes from the csv file) in a text file where each line will be a single review (that is, line 1 will contain the entire first review, line 2 will contain the entire second review and so on. Check whether you have saved all the reviews, or did you miss out on anything. [1]

Question 3: Representation

a. Now, use scikit-learn's TF-IDF vectorizer to perform the same task in 3a. Use this tutorial to your advantage: https://melaniewalsh.github.io/Intro-Cultural-Analytics/05-Text-Analysis/03-TF-IDF-Scikit-Learn.html [2]

b. We have read about word embeddings, and now we will see some practically. Download this file: https://drive.google.com/file/d/14DWN7qZTfTttV1rkeUMJhQ2rLjx7AdD9/view?usp=drive_link [3]
   This will have almost all the words in the English language and their GLoVe word embedding in 100-space (that is, each word is represented by 100-dimensional vectors). Load this file and extract the vectors for all the words. Then, find cosine similarities between these vectors:
   - Man and Woman
   - Cat and Dog
   - King and Queen

   Do they reflect what we learnt in class? Show that (in your code, using cosine similarity), that the vector you get from King - Man + Woman is close to the vector of Queen, that is, prove this equation: King - Man + Woman = Queen.