CIS/STA 9665: Assignment 3

Applied Natural Language Processing

Guidelines:

- Use Python as a programming language and finish this assignment in Jupyter Notebook
- Work is to be done individually for this assignment
- Students handing in similar work will both receive a grade of 0 and will face disciplinary actions.

Chapter 4. Writing Structured Programs

1. Create a list of words = ['is', 'it', 'good', '?']. a) Use a series of assignment statements (e.g. words[1] = words[2]) and a temporary variable tmp to transform this list into the list ['it', 'is', 'good', '!']. b) Now do the same transformation using tuple assignment.

2. Write code that removes whitespace at the beginning and end of a string (' this is a sample sentence '), and normalizes whitespace between words to be a single space character.
a) do this task using split() and join()
b) do this task using regular expression substitutions

3. sent1= ['The', 'dog', 'gave', 'John', 'the', 'newspaper']. Now assign sent2=sent1. Modify sent1[1]='monkey'. Please review section 4.1 -Assignment in Chapter 4 to answer the following questions:
a) verify that sent2 has changed
b) Now try the same exercise but instead assign sent2=sent1[:]. Modify sent1[1]='monkey' and see what happens to sent2. Explain.
c) Now define text1=[['The', 'dog', 'gave', 'John', 'the', 'newspaper'], ['John', 'is', 'happy']]. Now assign text2=text1[:], assign a new value to one of the words (text1[0][1]='monkey'). Check what happens to text2. Explain.
d) Extract successive overlapping 4-grams from ['The', 'dog', 'gave', 'John', 'the', 'newspaper'].

4. Write a function that prints any word that appeared in the last 20% of a text that had not been encountered earlier. Use text1 from nltk.book to call this function.

5. Write a program that takes the sentence ("we have seen two kinds of two sequence objects") expressed as a single string, splits it and counts up the tokens. Get it to print out each token and the token's frequency, one per line, in alphabetical order. You should write a function and call that function to process the sentence.

6. Write a function shorten(word, n) to process a text (“big big big world today tomorrow good Today good”), omitting the n most frequently occurring words of the text. You should use w.lower() to normalize the text first. Please call this shorten function.

7. Please use the sample code from Lab 4 about the TF-IDF to summarize your own text.

8. Write a function that takes a list of words (containing duplicates) (i.e. words=['table','chair','desk','table','table','chair']) and returns a list of words (with no duplicates) sorted by decreasing frequency. E.g. if the input list contained 10 instances of the word table and 9 instances of the word chair, then table would appear before chair in the output list. You should use **lambda** in the sorted( ) function.

9. Write a function that takes a text (e.g. text3 from nltk.book) and a vocabulary (e.g. nltk.corpus.words.words()) as its arguments and returns the set of words that appear in the text but not in the vocabulary. Both arguments can be represented as lists of strings. Can you do this by using set.difference()?

10. Choose your own webpage (in html format) and output the 20 most common words in this web page. You should use w.lower() to normalize the text. Please get rid of stop words, numbers, and punctuations. Please define a function and call that function.

What to Submit

a. Use Python as a programming language and finish this assignment in Jupyter Notebook
b. I have created an ipynb file with questions. Please add your code and answers in this ipynb file
c. After completion, please save your finalized ipynb file as a PDF file
d. Submit both **PDF file** and **ipynb file** to Blackboard
e. Please answers questions clearly, concisely, and completely. To answer some questions, the code is not sufficient. You should complement your answers in words by using **comments (#)**
f. The assignment will be graded on the correctness of the answers, comprehensiveness of the analysis, clarity of results' presentation and neatness of the report.