Natural Language Processing: Assignment 01 date: 22/01/2018 **due date: 29/01/2018**

Marks: 10 (5 marks for summary and 5 marks for the word clouds)

1. **Using NLTK toolkit on Python, produce a 10 sentence summary of a text dataset.**

- Download the dataset (these are the reviews of a particular restaurant) **Result1.txt** from the **Google Drive's course assignments folder**
- Split the data into sentences, then into words and finally eliminate the stop words
- Now find frequency of each word (after eliminating Stop Words)
- The more common a word, the more frequent it is.
- The score of a sentence is sum of frequency of its constituent words. You have to calculate score of each sentence in the review dataset.
- Sort the sentences in ascending order as per the above score & Print the top 10 sentences
- Tag the word tokens and choose the word according to tokens
- Make **three word clouds** with the followings (they may not be very accurate as we are using tagging and the review text has many words that were not in corpus used for tagging)
    - Most frequent words
    - Nouns
    - Adjective

Instructions for coding:

1. Try to familiarize yourself with NLTK (if not already). The NLTK online resource is the best. You can see the jupyter notebook provided in code_used_in_class folder in shared Google Drive
2. In general, you should use NLTK's sent_tokenize to tokenize text into sentences, followed by word_tokenize to tokenize the sentence into words. At this point, you should check the words for stop words and then calculate the frequency of each word etc.
3. NLTK uses Penn Treebank tagset. https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html
4. You should install Python word cloud library (conda install). Note that it requires a string of words. https://github.com/amueller/word_cloud
5. In case you are using windows 10 with Anaconda 3, you may face some issue in word cloud (at least I faced). A possible workaround that I found from net and it worked for me :
    a. Download the .whl file compatible with your Python version and your windows distribution (32bit or 64bit) from here
    b. cd to the file path
    c. Run this command `python -m pip install <filename>`

Marks: 5

2. **Computing minimum edit distances by hand, figure out whether "drive" is closer to "brief" or to "divers" and what the edit distance is. Use 1-insertion, 1-deletion, 2-substitution costs.**

Marks: 5

3. **Design an FSA to recognize simple date expressions like March 15, the 22nd of November, Christmas. Extend this date FSA to handle deictic expressions like Yesterday, tomorrow, a week from tomorrow, the day before yesterday, Sunday, next Monday, three weeks from Saturday**