CS3216/AI5203
In-class Assignment 1- NLP

Deadline: 9th February 2024

Time: 4:30 pm, 1630 hrs IST

Total Marks: 40

Instructions: -

- Please don't copy from the internet or any other student.
- Plagiarism will thoroughly be checked for all submissions.
    - Refer to the policy on Academic Unfair means.
- You may use any libraries such as spacy, nltk, regex libraries.
- Allowed Programming Language: Python
- It is recommended to use Jupyter Notebook/Google colab
- **Submission link:** https://forms.gle/aJmFTxm1AVNzpD9dA
- The format of the submission file should be: **RollNo-A1.ipynb**
    - The python notebook file should contain:
      - The code in a step-by-step format with the results of each stage
      - Readable comments for each stage
      - Short 1-2 lines analysis/observations of the results

Problem Statement:

Suppose you're tasked with building a program to analyze text data using regular expressions in Python. Your program needs to preprocess a given text by tokenizing it into individual words, performing normalization such as converting all characters to lowercase, and applying specific regular expression patterns for pattern matching.

Dataset: Sherlock Holmes.txt

Task: Using the Sherlock Holmes.txt file, perform the following tasks in python:

1. Design and implement a preprocessing module for the problem that effectively tokenizes the document, normalizes them, and extracts relevant information using regular expressions. [10 marks]

Deliverables:

- List of all the tokens, stemmed, lemmatized, normalized, and extracted relevant words (you are free to design your own metric to find relevancy of word to document) [10 marks]
- Plot the analysis of both stemmed and lemmatized words and compare the results:
    - a) Word frequency [10 marks]

      (Hint: Select the top 10 most commonly occurring words)
    - b) Most common words ending with 'ing' [10 marks]

Bonus: [15 marks]

2. Identify and address the difficulties associated with building the preprocessing module, such as designing robust regular expressions, handling linguistic variations, and optimizing for efficiency.

3. Test the preprocessing module and discuss potential strategies for mitigating the identified difficulties and improving the performance of the preprocessing module in a real-world scenario.

---------All the best--------