**Name: Hamad Nasir**

**Roll No: 120312**

**Section: BSCS-6C**

## **Speech and Image Processing - Lab 2**

**Task description:**

**Part (a)**

Write a python program that takes two single-line lowercase English files reference.txt and hypothesis.txt, and outputs the file result.txt containing Levenshtein distance of these two files as below. The distance should be word level and not character level.

**Part (b)**

Modify the above program so that it ignores 10 common words such that

* Insertions and deletions involving these common words are ignored
* Substitutions are ignored when both initial and final word are one of 10 common words

List of 10 common words:

the, of, and, a, be, this, there, an, been, some

**Source Code:**

**import** nltk

#function for calculating levenshtein distance

**def** levenshtein**(**s1**,** s2**):**

**if** len**(**s1**)** **<** len**(**s2**):**

**return** levenshtein**(**s2**,** s1**)**

# len(s1) >= len(s2)

**if** len**(**s2**)** **==** 0**:**

**return** len**(**s1**)**

previous\_row **=** range**(**len**(**s2**)** **+** 1**)**

**for** i**,** c1 **in** enumerate**(**s1**):**

current\_row **=** **[**i **+** 1**]**

**for** j**,** c2 **in** enumerate**(**s2**):**

insertions **=** previous\_row**[**j **+** 1**]** **+** 1 # j+1 instead of j since previous\_row and current\_row are one character longer

deletions **=** current\_row**[**j**]** **+** 1 # than s2

substitutions **=** previous\_row**[**j**]** **+** **(**c1 **!=** c2**)**

current\_row**.**append**(**min**(**insertions **,** deletions **,** substitutions**))**

previous\_row **=** current\_row

**print(**"Levenshtein distance is:"**,**previous\_row**[-**1**]** **)**

**print(**"Insertions:"**,**insertions **-** 1**)**

**print(**"Deletions:"**,**deletions **-** 1**)**

**print(**"Substitutions:"**,**substitutions **-** 1**)**

#function for removing the common words and calculating levenshtein values

**def** filter\_common\_words**(**sent1**,**sent2**):**

list **=** **[**'the '**,** 'of '**,** 'and '**,** 'a '**,** 'be '**,** 'this '**,** 'there '**,** 'an '**,** 'been '**,** 'some '**]**

#removing the common words

**for** i **in** list**:**

sent1 **=** sent1**.**replace**(**i**,**''**)**

sent2 **=** sent2**.**replace**(**i**,**''**)**

#tekenizing the filtered strings

tok1 **=** nltk**.**word\_tokenize**(**sent1**)**

tok2 **=** nltk**.**word\_tokenize**(**sent2**)**

#calculating levenshtein values for filtered strings

levenshtein**(**tok1**,**tok2**)**

#--------------------------------------------------------------------------

#Task: 1

#reading text from files

f**=**open**(**"reference.txt"**,** "r"**)**

sent1 **=** f**.**read**()**

f1**=**open**(**"hypothesis.txt"**,** "r"**)**

sent2 **=** f1**.**read**()**

#tekenizing the filtered strings

tk1 **=** nltk**.**word\_tokenize**(**sent1**)**

tk2 **=** nltk**.**word\_tokenize**(**sent2**)**

**print(**"To Check:\nLevenshtein Distance = Insertions + Substitutions + Deletions / 3\n "**)**

**print(**"\nTask:1"**)**

#calculating levenshtein values for strings

levenshtein**(**tk1**,**tk2**)**

#--------------------------------------------------------------------------

#Task: 2

#calculating levenshtein values for filtered strings

**print(**"\n\nTask:2\nAfter filtering common words:"**)**

filter\_common\_words**(**sent1**,**sent2**)**

**Output:**

