CSDL7013 Natural Language Processing Lab 5

- NOTE: Prepend your Roll Number to the name of this file

Minimum Edit Distance

1. **Title:** Minimum Edit Distance
2. **Objective/Aim:** To implement Minimum Edit Distance Algorithm
3. **Due Date:** Friday September 09, 2022

Name : Harshvardhan Singh

Roll Number : 1019161

Q. 1. Implement the minimum edit distance algorithm in Python, and find out the distances between following pair of words:

```
(apple,mango), (there,their), (laughter,daughter), (rain,reign),
(right,write)
```

[*03 Marks*]

```
import re

def editDist(a, b):
  m = len(a)
  n = len(b)
  dp = [[0] * (n + 1) for _ in range(m + 1)]
  for i in range(m + 1):
    for j in range(n + 1):
      if i == 0:
        dp[i][j] = j
      elif j == 0:
        dp[i][j] = i
      else:
        k = 1
        if a[i - 1] == b[j - 1]:
          k = 0
        dp[i][j] = min(dp[i - 1][j - 1] + k, dp[i - 1][j] + 1, dp[i][j
- 1] + 1)
  return dp[m][n]

for a, b in [("apple", "mango"), ("there", "their"), ("laughter",
"daughter"), ("rain", "reign"), ("right", "write")]:
  print(f"dist({a}, {b}) = {editDist(a, b)}")
```

```
dist(apple, mango) = 5
dist(there, their) = 2
dist(laughter, daughter) = 1
```

```
dist(rain, reign) = 2
dist(right, write) = 4
```

Q. 2. Words with correct spellings are given in the file **bagofwords.txt**, and in the **errordocument.txt**, each line contains a sentence with few words misspelled. Find the correct word from the bag of words and replace it and write the corrected sentence in **correcteddocument.txt**

[*03 Marks*]

```
def getClosest(words, target):
  return min(words, key = lambda w: editDist(w.lower(),
target.lower()))

def fix(content, words):
  corrected = content
  for word in re.findall("\w+", content):
    correction = getClosest(words, word)
    if word.lower() != correction.lower():
      corrected = re.sub(word, correction, corrected)
  return corrected

words = open("Data/bagofwords.txt").read().split()
content = open("Data/errordocument.txt").read()
res = open("Data/correcteddocument.txt", "a")
corrected = fix(content, words)
res.write(corrected)
res.close()
```

Q. 3. From the file **newtonlaws.txt**, ignoring typical stopwords, generate the bag of words. From the files **answers1.txt** to **answers4.txt**, find out the number of words misspelled and the degree of wrong spelling, and assign a score to that file. (Think of some relevant metric.)

```
content = open("Data/newtonlaws.txt").read()
words = re.findall("\w+", content)
for file in ["answers1.txt", "answers2.txt", "answers3.txt",
"answers4.txt"]:
  print(file)
  print("=" * len(file))
  currcontent = open(f"Data/{file}").read()
  currwords = re.findall("\w+", currcontent)
  degree = 0
  ctr = 0
  numwords = len(currwords)
  for w in currwords:
    if w.lower() not in {'a', 'the', 'is'}:
      correct = getClosest(words, w)
      diff = editDist(w.lower(), correct.lower())
      degree += diff
      if diff > 0:
```

```
        ctr += 1
    print(f"Mispelled Words: {ctr}")
    print(f"Degree of wrong spelling: {degree}")
    print(f"Score: {100 * (numwords - ctr) / (numwords):.4g}")
    print()
```

```
answers1.txt
============
Mispelled Words: 5
Degree of wrong spelling: 5
Score: 83.33

answers2.txt
============
Mispelled Words: 11
Degree of wrong spelling: 11
Score: 63.33

answers3.txt
============
Mispelled Words: 9
Degree of wrong spelling: 10
Score: 70

answers4.txt
============
Mispelled Words: 4
Degree of wrong spelling: 4
Score: 86.67
```