CSDL7013 Natural Language Processing Lab 4

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

Regular Expressions

1. **Title:** n-gram language model
2. **Objective/Aim:** To generate n-gram language model
3. **Due Date:** Friday August 19, 2022

Name : Harshvardhan Singh

Roll Number : 1019161

Q. 1. Read the file **survey.txt** given, generate the trigram model. Consider each sentence separately.

[*02 Marks*]

```
#Solution to Q. 1.
from nltk import ngrams
with open("survey.txt",'r') as sr:
  sentences = sr.read()
  sr.close()
sentences = sentences.split("\n")
for sentence in sentences:
  # print("trigram model for \"",sentence,"\":")
  trigrams = ngrams(sentence.split(),n=3)
  for gram in trigrams:
    print(gram)
```

```
('I', 'like', 'biryani')
('I', 'like', 'cake')
('I', 'like', 'chocolate')
('I', 'like', 'biryani')
('I', 'get', 'water')
('I', 'get', 'lassi')
('I', 'get', 'eggs')
('I', 'get', 'dosa')
('I', 'get', 'idli')
('I', 'like', 'apple')
('I', 'like', 'mango')
('I', 'like', 'biryani')
('I', 'like', 'cake')
('I', 'like', 'chocolate')
```

✓ 0s completed at 3:10 PM

```
('I', 'want', 'wadapav')
('I', 'want', 'water')
('I', 'want', 'lassi')
('I', 'want', 'eggs')
('I', 'like', 'apple')
('I', 'like', 'banana')
('I', 'like', 'milk')
('I', 'like', 'coffee')
('I', 'want', 'orange')
('I', 'want', 'wadapav')
('I', 'want', 'water')
('I', 'want', 'lassi')
('I', 'want', 'eggs')
('I', 'get', 'water')
('I', 'get', 'lassi')
('I', 'get', 'eggs')
('I', 'get', 'dosa')
('I', 'get', 'idli')
('I', 'like', 'biryani')
('I', 'like', 'cake')
('I', 'want', 'orange')
('I', 'want', 'wadapav')
('I', 'want', 'water')
('I', 'want', 'lassi')
('I', 'want', 'eggs')
('I', 'like', 'chocolate')
('I', 'like', 'biryani')
('I', 'like', 'apple')
('I', 'like', 'pulav')
('I', 'like', 'dhokla')
('I', 'want', 'orange')
('I', 'want', 'wadapav')
('I', 'get', 'apple')
('I', 'get', 'biryani')
('I', 'get', 'water')
('I', 'get', 'lassi')
('I', 'get', 'water')
('I', 'get', 'lassi')
('I', 'get', 'eggs')
('I', 'get', 'dosa')
('I', 'get', 'idli')
('I', 'get', 'eggs')
```

```
file = open('survey.txt', 'r')
read_data = file.read()
count_occur = read_data.count("I want")
occurrences = read_data.count("I want water")
print('Number of I want  :', count_occur)
print('Number of I want wanter :', occurrences)
prob = occurrences/count_occur
print(prob)
```

```
Number of I want  : 55
Number of I want wanter : 10
0.18181818181818182
```

Q. 2. Read the given text files, remove all punctuation symbols, change all words to small case, generate trigram model for each one of them.

[*08 Marks*]

```
#Solution to Q. 2.
filename = input("Enter filename: ")
def remove_punc(string):
    punc = '''!()-[]{};:'"\, <>./?@#$%^&*_~'''
    for ele in string:
        if ele in punc:
            string = string.replace(ele, "")
    return string
try:
    with open(filename,'r',encoding="utf-8") as f:
        data = f.read()
    with open(filename,"w+",encoding="utf-8") as f:
        f.write(remove_punc(data))
    print("Removed punctuations from the file", filename)
except FileNotFoundError:
    print("File not found")
```

```
Enter filename: madteaparty.txt
Removed punctuations from the file madteaparty.txt
```

```
file = open('madteaparty.txt', 'rt').read().lower()
file
```

```
'youshouldlearnnottomakepersonalremarksalicesaidwithsomeseverityitsveryrud
e\nthehatteropenedhiseyesverywideonhearingthisbutallhesaidwaswhyisaravenli
keawritingdesk\ncomeweshallhavesomefunnowthoughtaliceimgladtheyvebegunaski
ngriddlesibelieveicanguessthatsheaddedaloud\ndoyoumeanthatyouthinkyoucanfi
ndouttheanswertoitsaidthemarchhare\nexactlysosaidalice\nthenyoushouldsaywh
atyoumeanthemarchharewenton\nidoalicehastilyrepliedatleastatleastimeanwhat
isaythatsthesamethingyouknow\nnotthesamethingabitsaidthehatteryoumightjust
aswellsaythatiseewhatieatisthesamethingasieatwhatisee\nyoumightjustaswells
ayaddedthemarchharethatilikewhatigetisthesamethingasigetwhatilike\nyoumigh
```