

1619313040/3301 (Data Science Oriented Programming Language-Python)

SEMESTER 2 (Spring), 2021/2022

LABORATORY WORK THREE

NAME: MD Omar Faruq

STUDENT NO.: 201939060073

SUBMITTED TO:

Assoc. Prof. FANG YU

MARKING SCHEME (TO BE FILLED BY THE LECTURER)

|  |  |
| --- | --- |
| **CRITERIA** | **MARKS** |
| CONTENTS | /6 |
| ORGANIZATION/STRUCTURE | /2 |
| WRITING MECHANICS | /2 |
| **TOTAL** | **/10** |

**DATA ACQUISITION PRACTICE**

In this section, students must display originality of their writings by jotting down their understanding of crawling. Students need to fully describe and discuss in terms of:

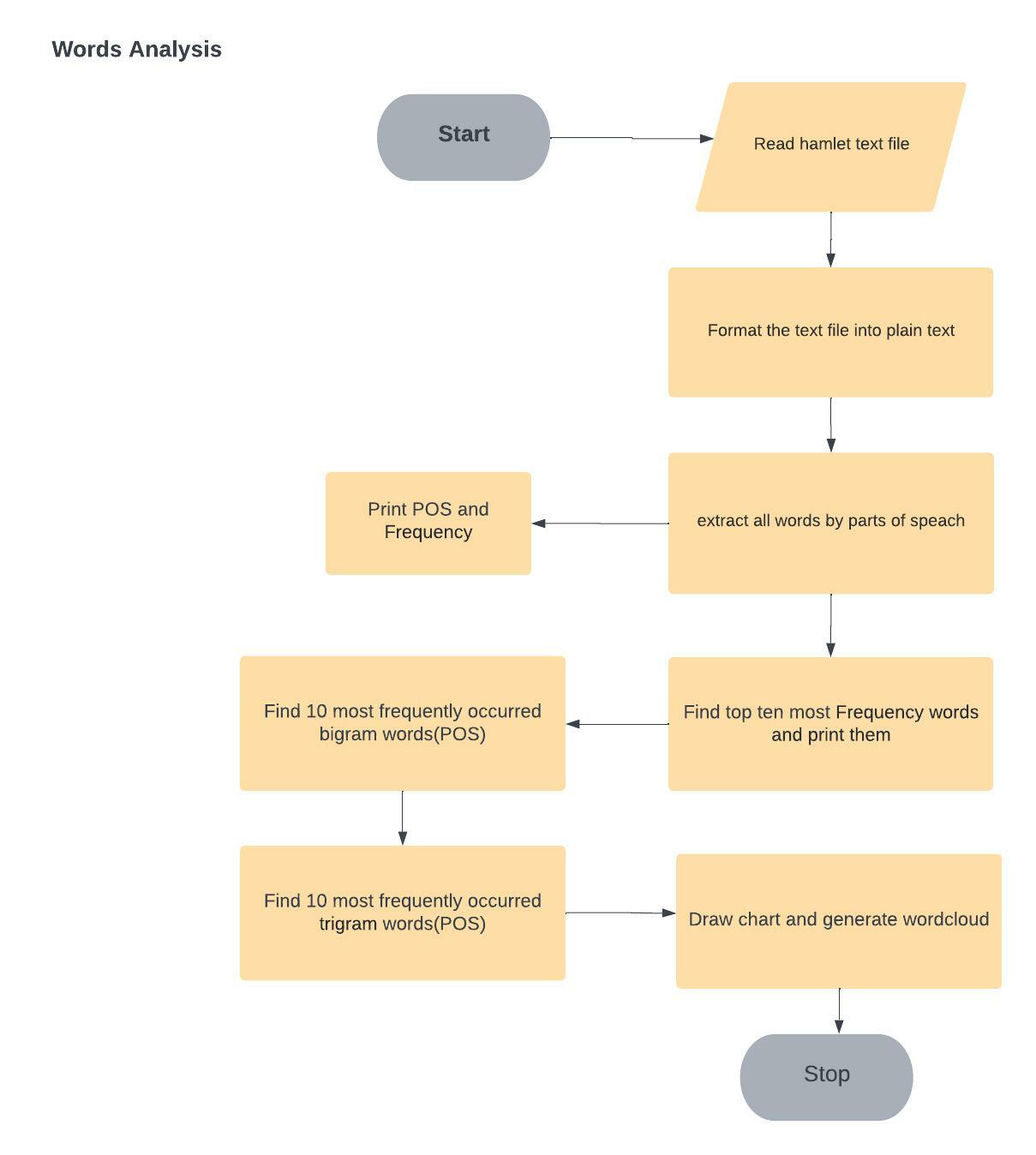
1. **The PROCEDURES of Words Analysis, and flowcharts for your python codes**
2. **Screenshot your running results (better applying video clips), upload your source codes and your lab work report.**

*FOLLOWING IS THE LABWORKS*

**Description about the task:**

1. Analysis the part of speech for Hamlet.txt
2. Display high frequency words with their ‘pos’ and counts.
3. Display a wordcloud of the corpus ‘Hamlet’
4. Make a summary about this task.

**Flowchart**



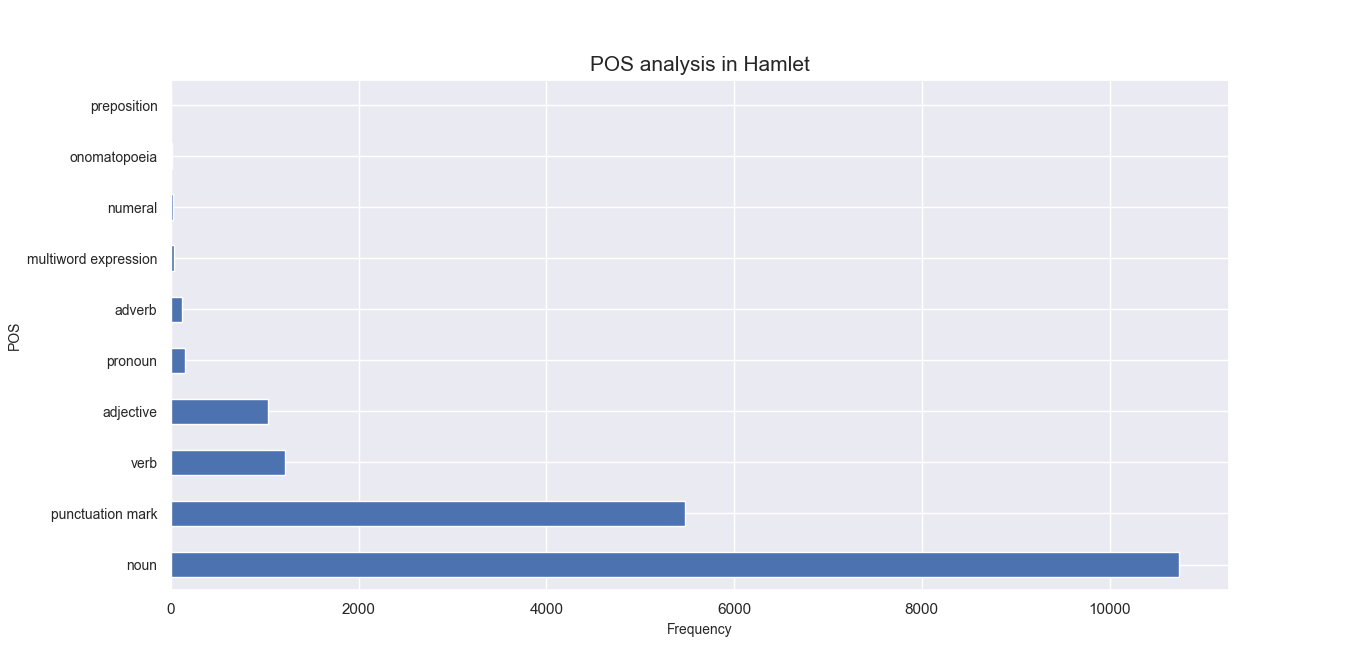
**Source code**

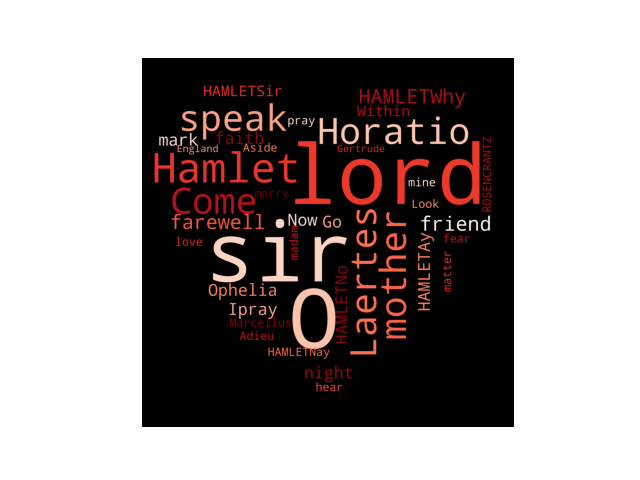
import pynlpir as pynlpir  
import pandas as pd  
import matplotlib.pyplot as plt  
from matplotlib.font\_manager import FontProperties  
import seaborn as sns; sns.set()  
from wordcloud import WordCloud  
from imageio import imread  
from nltk.util import ngrams  
from collections import Counter  
  
font = FontProperties(fname=r'c:\windows\fonts\arial.ttf', size=15)  
text\_file = open('hamlet.txt', 'r').read().replace('\n', '')  
  
pynlpir.open()  
pynlpir.segment(text\_file, pos\_names='parent', pos\_english=True)  
  
word\_list = []  
y\_list = []  
  
y\_list.extend(pynlpir.segment(text\_file, pos\_names='parent', pos\_english=True))  
  
for i in range(len(y\_list)):  
 y\_w = list(y\_list[i])  
 word\_list.append(y\_w)  
  
  
f\_words = pd.DataFrame(word\_list, columns=["word", "pos"])  
f\_words.head(25)  
f\_words.index.size  
  
stopword = open('stopwords\_en.txt', encoding='utf-8').read()  
  
for i in range(f\_words.shape[0]):  
 if f\_words.word[i] in stopword:  
 f\_words.drop(i, inplace=True)  
 else:  
 pass  
  
wordFormat = pd.DataFrame(f\_words["pos"].value\_counts(ascending=False))  
wordFormat.rename(columns={'pos': 'Frequency'}, inplace=True)  
  
# counting POS  
print(f"POS {wordFormat.head()}")  
print("\n--------------------------------\n")  
  
# counting Frequency  
print(f"Word Frequency : {wordFormat['Frequency'].sum()}")  
print("\n--------------------------------\n")  
  
# counting percentage of POS  
wordFormat['percentage'] = wordFormat['Frequency'] / wordFormat['Frequency'].sum()  
  
print(wordFormat['percentage'])  
print("\n--------------------------------\n")  
  
  
# Extracting words from the file by splitting by " "  
wordsCounter = open("hamlet.txt").read().split()  
  
# Print the top 10 most frequently occurred words in the file, and provide the word frequency for each of them.  
word\_freq = Counter(wordsCounter)  
print("\n\n---------High Frequency words--------\n")  
df\_wordCount = pd.DataFrame(word\_freq.most\_common(10))  
df\_wordCount.columns = ["Words", "Frequency"]  
print(df\_wordCount)  
  
  
# Print the top 10 most frequently occurred bigram words(pair of two words) in the file, and provide the word  
# frequency for each of them  
print("\n\n---------10 most frequently occurred bigram words(POS)--------\n")  
n\_gram = 2  
bigram = Counter(ngrams(wordsCounter, n\_gram)).most\_common(10)  
df\_wordCountPOS = pd.DataFrame(bigram)  
df\_wordCountPOS.columns = ["Words", "Frequency"]  
print(df\_wordCountPOS)  
  
# Print the top 10 most frequently occurred trigram words(three-word triplet) in the file, and provide the word  
# frequency for each of them.  
print("\n\n---------10 most frequently occurred trigram words--------\n")  
n\_gram = 3  
trigram = Counter(ngrams(wordsCounter, n\_gram)).most\_common(10)  
df\_wordCountThree = pd.DataFrame(trigram)  
df\_wordCountThree.columns = ["Words", "Frequency"]  
print(df\_wordCountThree)  
  
  
# draw chart  
plt.subplots(figsize=(7, 5))  
wordFormat.iloc[:10]['Frequency'].plot(kind='barh')  
plt.yticks(fontproperties=font, size=10)  
plt.xlabel('Frequency', fontproperties=font, size=10)  
plt.ylabel('POS', fontproperties=font, size=10)  
plt.title('POS analysis in Hamlet', fontproperties=font)  
plt.show()  
  
# draw wordcloud  
myText = ''.join(f\_words.word)  
myText[:20]  
bg\_pic = imread('love.png')  
wc = WordCloud(mask=bg\_pic, max\_words=500, max\_font\_size=50, min\_font\_size=3, background\_color='black', colormap='Reds\_r', scale=15.5, contour\_color='red', repeat=True)  
  
wc.generate(myText)  
plt.imshow(wc)  
plt.axis('off')  
plt.show()  
pynlpir.close()

**Output**

"C:\Program Files\Python310\python.exe" "G:/1. SWPU/6th Semester/Data Science/pythonProject/data\_science/Lab3/WordsAnalysisHamlet.py"

POS Frequency  
noun 10740  
punctuation mark 5478  
verb 1218  
adjective 1033  
pronoun 154  
  
--------------------------------  
  
Word Frequency : 18823  
  
--------------------------------  
  
noun 0.570579  
punctuation mark 0.291027  
verb 0.064708  
adjective 0.054880  
pronoun 0.008181  
adverb 0.006428  
multiword expression 0.002019  
numeral 0.001116  
onomatopoeia 0.000584  
preposition 0.000372  
time word 0.000053  
particle 0.000053  
Name: percentage, dtype: float64  
  
--------------------------------  
  
  
  
---------High Frequency words--------  
  
 Words Frequency  
0 the 988  
1 and 693  
2 of 623  
3 to 604  
4 I 513  
5 a 450  
6 my 441  
7 in 387  
8 HAMLET 378  
9 you 356  
  
  
---------10 most frequently occurred bigram words(POS)--------  
  
 Words Frequency  
0 (KING, CLAUDIUS) 109  
1 (in, the) 86  
2 (LORD, POLONIUS) 86  
3 (QUEEN, GERTRUDE) 73  
4 (to, the) 64  
5 (of, the) 60  
6 (my, lord.) 56  
7 (I, have) 50  
8 (I, am) 41  
9 (my, lord,) 41  
  
  
---------10 most frequently occurred trigram words--------  
  
 Words Frequency  
0 (my, lord., HAMLET) 39  
1 (my, lord?, HAMLET) 19  
2 (in, the, castle.) 12  
3 (the, castle., Enter) 12  
4 (My, lord,, I) 12  
5 (ROSENCRANTZ, and, GUILDENSTERN) 11  
6 (I, do, not) 10  
7 (room, in, the) 9  
8 (I, pray, you,) 9  
9 (Enter, KING, CLAUDIUS,) 8  
  
Process finished with exit code 0





**Summary**

I have completed lab work 3. This is an interesting lab. I have used many python libraries in this lab work. The task was to analysis Hamlet text file and find high frequency words. So, to find parts of speech of the text I have used pynlpir library. By using this library, I am able to separate each word by their parts of speech. Then I have counted all words and print the number of POS, Percentages. After that, I counted top 10 most high frequency words in this text file. High frequency words are those that appear most commonly in everyday usage. Some of them are simple nouns or verbs, such as mother and women or write and speak. Many of them are also pronouns (such as I, that, and your) or forms of the verb 'to be' (such as are or were) that are quite often used in everyday speech. In this text file the most high frequency word is “ The ”, and its frequency is 988.

After that, I draw a chart which is indicating POS and frequency. And I also generate a wordCloud figure for Hamlet.