**Name: Hamad Nasir**

**Roll No: 120312**

**Section: BSCS-6C**

**Speech and Image Processing - Lab 1**

**Task Description:**

Download/copy a text of around 1,000 words in Urdu, and store it in a UTF-8 file. The file is named corpus.txt

**Part (a)**

Using python, remove punctuation and diacritics from this text, so that it becomes

**Part (b)**

Using python, make a file vocabulary.txt, that contains all the unique words in the corpus\_filtered.txt file and their frequencies. The file should be sorted according to frequencies

**Solution:**

**Code:**

#importing libraries

**import** io

**import** sys

**import** argparse

**import** re

**import** string

**import** collections

**import** operator

#storing all urdu punctuations in a list

urdu\_punctuations **=** '''`÷×۔؛<>\_()\*&^%][ـ،/:"؟.,'{}~¦+|!”…“–ـ'''

#storing all english punctuations in a list

english\_punctuations **=** string**.**punctuation

#adding both lists together

punctuations\_list **=** urdu\_punctuations **+** english\_punctuations

#compiling all urdu diacritics

urdu\_diacritics **=** re**.**compile**(**"""

ّ | # Tashdid

َ | # Fatha

ً | # Tanwin Fath

ُ | # Damma

ٌ | # Tanwin Damm

ِ | # Kasra

ٍ | # Tanwin Kasr

ْ | # Sukun

ـ # Tatwil/Kashida

"""**,** re**.**VERBOSE**)**

#function to remove arabic diacritics

**def** diacritics\_remover**(**text**):**

text **=** re**.**sub**(**urdu\_diacritics**,** ''**,** text**)**

**return** text

#function to remove arabic punctuation

**def** punctuations\_remover**(**text**):**

translator **=** str**.**maketrans**(**''**,** ''**,** punctuations\_list**)**

**return** text**.**translate**(**translator**)**

#making object of argument parser

parser **=** argparse**.**ArgumentParser**(**description**=**'Pre-process arabic text (remove '

'diacritics, punctuations, and repeating '

'characters).'**)**

#adding argument corpus.txt as input file

parser**.**add\_argument**(**'-i'**,** '--corpus'**,** type**=**argparse**.**FileType**(**mode**=**'r'**,** encoding**=**'utf-8'**),**

help**=**'input file.'**,** required**=True)**

#adding argument corpus\_filtered.txt as output file

parser**.**add\_argument**(**'-o'**,** '--corpus\_filtered'**,** type**=**argparse**.**FileType**(**mode**=**'w'**,** encoding**=**'utf-8'**),**

help**=**'out file.'**,** required**=True)**

#function for getting frequency of words and printing them in descending order

**def** counter**(**text**):**

#opening a file for writing

f **=** open**(**"vocabulary.txt"**,** "w+"**,** encoding**=**"utf-8"**)**

#preparing a word list by spliting lowering the words

word\_list **=** **[**word**.**lower**()** **for** word **in** text**.**split**()]**

#making counter object

count **=** collections**.**Counter**()**

#counting the words

**for** word **in** word\_list**:**

count**[**word**]** **+=** 1

**print(**"\nThe frequency of words in descending order is as follows:"**)**

#calculating frequency of words

freq **=** operator**.**itemgetter**(**1**)**

#getting the decreasing order of word, frequency pair

**for** k**,** v **in** sorted**(**count**.**items**(),** reverse**=True,** key**=**freq**):**

**print(**"%3d %s" **%** **(**v**,** k**))**

#writing in file

f**.**write**(**"%3d %s \n" **%** **(**v**,** k**))**

#closing the file

f**.**close**()**

**if** \_\_name\_\_ **==** '\_\_main\_\_'**:**

args **=** parser**.**parse\_args**()**

#reading corpus.txt

text **=** args**.**corpus**.**read**()**

#removing punctuations

text **=** punctuations\_remover**(**text**)**

#removing diacritics

text **=** diacritics\_remover**(**text**)**

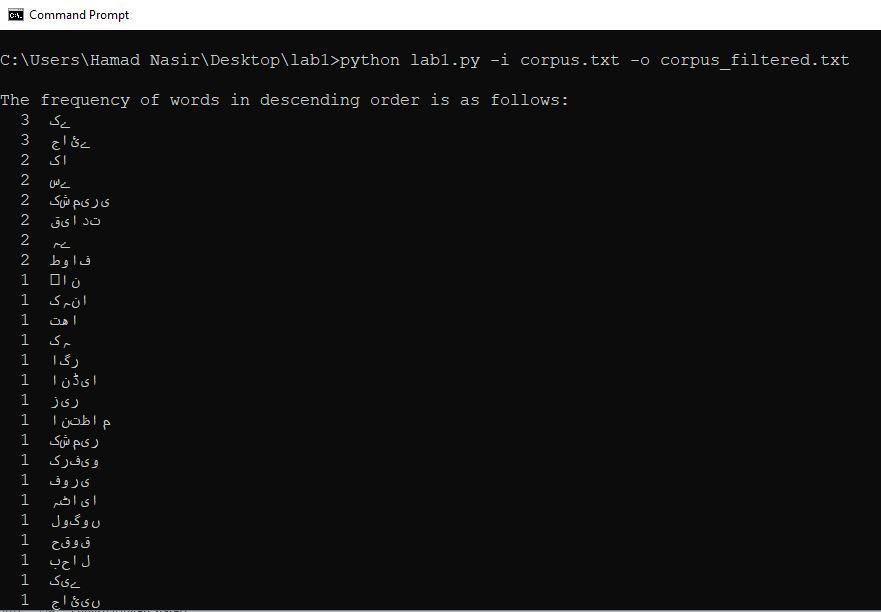
#writng the filtered text in corpus\_filtered.txt

args**.**corpus\_filtered**.**write**(**text**)**

#calling counter function and passing it filtered text

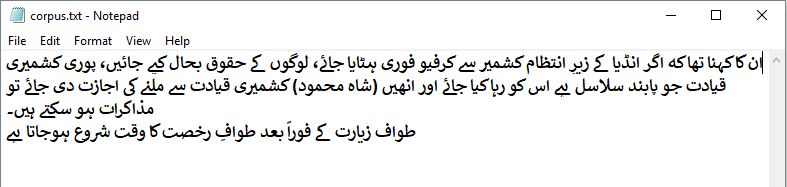
counter**(**text**)**

**Output Screenshots:**

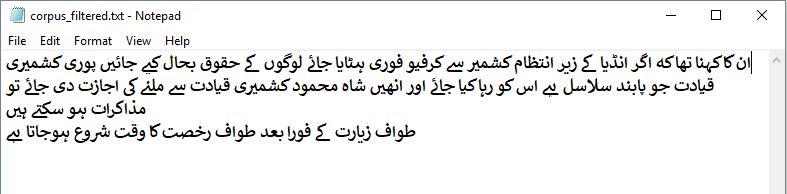


**Screenshots of files:**

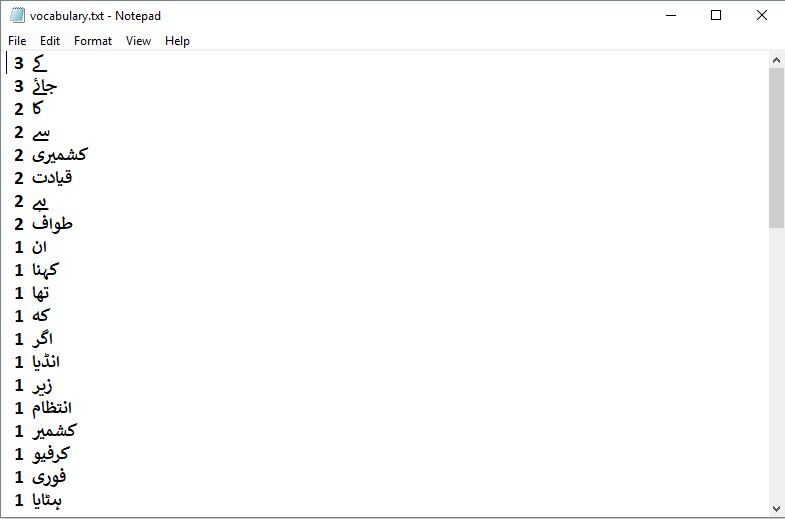
**Corpus.txt:**

****

**Corpus\_filtered.txt:**

****

**Vocabulary.txt:**

****