Exam Preparation Online

Assignment report for internship selection process

Assignment 4:(Role : Data Science)

The text source given to us was in the pdf format and so the steps to that are required to use the pdf file format in python are-

1. Use pachage- PyPDF2
2. Create a pdf file object and pass the required file name and mode as parameters for the open function.

pdfFileObj = open('JavaBasics-notes.pdf', 'rb')

1. Creating a pdf reader object as follows-

pdfReader = PyPDF2.PdfFileReader(pdfFileObj)

1. Printing number of pages in pdf file as follows-

print(pdfReader.numPages)

The output would be 23 as there are total 23 pages.

1. For the 23 pages I distributed their contents into 3 parts of strings named docs1, docs2 and docs 3, through the following process-

docs1=" " //Creating 3 empty string objects.

docs2=" "

docs3=" "

for i in range (1,9):

pageObj = pdfReader.getPage(i)

docs1=docs1+(pageObj.extractText())

for i in range (9,19):

pageObj = pdfReader.getPage(i)

docs2=docs2+(pageObj.extractText())

for i in range (19,23):

pageObj = pdfReader.getPage(i)

docs3=docs3+(pageObj.extractText())

pdfFileObj.close() //Closing the file’s opened terminal

1. Then the required packages for word extraction are imported as-

import math

from textblob import TextBlob as tb

1. Then the required functions were defined in the following way-

def tf(word, blob):

return blob.words.count(word) / len(blob.words)

def n\_containing(word, bloblist):

return sum(1 for blob in bloblist if word in blob.words)

def idf(word, bloblist):

return math.log(len(bloblist) / (1 + n\_containing(word, bloblist)))

def tfidf(word, blob, bloblist):

return tf(word, blob) \* idf(word, bloblist)

1. Then documents were created for the three docs created before using the tb() function.

And then creating a bloblist that has the three documents as it’s elements.

document1=tb(docs1)

document2=tb(docs2)

document3= tb(docs3)

bloblist = [document1,document2,document3]

1. Then the user will input the number of keywords the user wants to extract from the file.

no\_of\_keywords=input("Enter the number of keywords you want to extract-")

1. Then we iterate through the bloblist and calculate score for each of the keyword and on that basis, we print the keywords based on the tf-idf score value and that too in the descending order.

for i, blob in enumerate(bloblist):

scores = {word: tfidf(word, blob, bloblist) for word in blob.words}

sorted\_words = sorted(scores.items(), key=lambda x: x[1], reverse=True)

for word, score in sorted\_words[:int(no\_of\_keywords)]:

print("\tWord: {}, TF-IDF: {}".format(word, round(score, 10)))

THANK YOU