Solution12

November 16, 2018

0.0.1 Assignment 12

Perform the following actions: #### 1. Use the following codes to load the assignment12.csv which contains file names. How many file names in it? (10 points) file = open("Assignment_12.txt" , ’r’)
text1 = file.read()
file.close()

```
In [1]: import os
        import pandas as pd
        import re
        import nltk
        from nltk.book import *
```

```
*** Introductory Examples for the NLTK Book ***
Loading text1, ..., text9 and sent1, ..., sent9
Type the name of the text or sentence to view it.
Type: 'texts()' or 'sents()' to list the materials.
text1: Moby Dick by Herman Melville 1851
text2: Sense and Sensibility by Jane Austen 1811
text3: The Book of Genesis
text4: Inaugural Address Corpus
text5: Chat Corpus
text6: Monty Python and the Holy Grail
text7: Wall Street Journal
text8: Personals Corpus
text9: The Man Who Was Thursday by G . K . Chesterton 1908
```

```
In [2]: os.chdir(r'E:\GoogleDriveNew\PSU\DAAN862\Course contents\Lesson 12')

        file = open("Assignment_12.txt" , 'r')

        filenames = file.read()

        file.close()
```

```
In [3]: filenames
```

```
Out[3]: 'arxiv_annotate10_7_1.txt   arxiv_annotate10_7_2.txt   arxiv_annotate10_7_3.txt   arxi
```

```
In [4]: filenames_list = re.split( '\s+', filenames)

In [5]: len(filenames_list)

Out[5]: 90
```

2. Identify the pattern of the file names and find out how many file names match the pattern. (20 points)

```
In [6]: pattern = '[a-z]+_[a-z0-9]+_[0-9]+_[0-9]{1}.[a-z]{3}'
        re_pattern = re.compile(pattern)
        results = re_pattern.findall(filenames )
        results[:5]

Out[6]: ['arxiv_annotate10_7_1.txt',
         'arxiv_annotate10_7_2.txt',
         'arxiv_annotate10_7_3.txt',
         'arxiv_annotate1_13_1.txt',
         'arxiv_annotate1_13_2.txt']

In [7]: len(results)

Out[7]: 84
```

3. Find out file names who doesn't match with the pattern you designed. (20 points)

```
In [8]: filenames_notmatch = []
        for name in filenames_list:
            if not re_pattern.match(name):
                filenames_notmatch.append(name)

In [9]: filenames_notmatch

Out[9]: ['jdm_ann^otate3_120_1.txt',
         'jdm_anno&tate6_32_2.txt',
         'jdm_annotat#e8_177_2.txt',
         'plos_annotat*e1_6_2.txt',
         'plos_anno%tate5_1375_3.txt',
         'plos_annot@ate7_1233_2.txt']
```

4. Use following codes to read the text from "arxiv_annotate1_13_1.txt" in file = open("arxiv_annotate1_13_1.txt" , 'r')
text = file.read()
file.close()
Identify the words and normalizate it.

```
In [10]: file = open('arxiv_annotate1_13_1.txt', 'r')

         text = file.read()

         file.close()
```

```
In [11]: words = nltk.word_tokenize(text)
         words_clean = []
         for w in words:
             if w.isalnum():
                 words_clean.append(w)
```

```
In [12]: porter = nltk.PorterStemmer()
         words_stem = [porter.stem(w) for w in words_clean]
```

```
In [13]: words_stem [:10]
```

```
Out[13]: ['abstract',
          'misc',
          'although',
          'the',
          'internet',
          'as',
          'level',
          'topolog',
          'ha',
          'been']
```

```
In [14]: words_count = FreqDist(words_stem)
         words_count
```

```
Out[14]: FreqDist({'the': 44, 'of': 34, 'as': 28, 'and': 24, 'misc': 20, 'we': 20, 'a': 19, 'i
```

```
In [15]: len(words_count)
```

```
Out[15]: 294
```