task1_31901611

January 28, 2021

1 FIT5196 Task 1 in Assessment 1

Student Name: Prashasti Garg

Student ID: 31901611 Date: 23/01/2021

Version: 1.0

Environment: Python 3.7.9 and Jupyter notebook

Libraries used: please include the main libraries you used in your assignment here, e.g.,: * os (for fetching the directory path to read all the files located in a folder) * re (for regular expression to search a pattern) * langid (for classifying the text language)

1.0.1 Importing Libraries

- In this task, we are provided 140 text files,which includes covid-19 related tweets.
- The text files includes the id, text and date of each tweets.
- id is of 19 characters.
- text is the tweets related to covid-19.
- date is the Created_at which consists of date and time of tweets.

[1]:
```
import os
import re
import langid
```

1.0.2 Extracted the data from the text files

[2]:
```
# an empty list is created to append all the files
store = []
# the file is listed using os.listdir() from its destination
for file in os.listdir(r'D:\Jupyter Notebook\Wrangling\Dataset_txt'):
    # the files ending with '.txt' are collected
    if file.endswith('.txt'):
        # each file in the folder is open with a read comand
        with open(os.path.join(r'D:\Jupyter Notebook\Wrangling\Dataset_txt',␣
 ↪file), 'rt',  encoding = "utf8") as fin:
            text = fin.read()
            store.append(text)
```

1.0.3 The id, text and date in each line is stored in a list using Regex

- Text is collected using re.findall(), where regex is used to search for the specific pattern.

[3]:
```
# an empty list is created where the collected text is appended
tweet_extract = []
for i in (store):
    tweet_extract.append(re.findall(r'\'id\': \'[a-zA-Z0-9]{19}\', \'text\': \'.
 ↪*?\', \'Created_at\': \'[0-9]{4}\-[0-9]{2}\-[0-9]{2}', i))
```

1.0.4 Segregated id, text and date using Regex, from each line, is stored in a dictionary

- r''id': '[a-zA-Z0-9]{19}' : Used to find all the ids which has 19 characters in the extracted_tweet list
- r''text': '.*?" : Used to find all the text in the extracted_tweet list
- r''Created_at': '[0-9]{4}-[0-9]{2}-[0-9]{2}' : Used to find all the dates in the extracted_tweet list

[30]:
```
tweets_dict = {}
extracted_id_re= r'\'id\': \'[a-zA-Z0-9]{19}'
extracted_text_re= r'\'text\': \'.*?\''
extracted_date_re= r'\'Created_at\': \'[0-9]{4}\-[0-9]{2}\-[0-9]{2}'
for i in tweet_extract:
    for j in i:
        tweet_id = re.sub('\'id\': \'', '', re.findall(extracted_id_re, j)[0]).
 ↪strip("'")
        text = re.sub('\'text\': \'', '', re.findall(extracted_text_re, j)[0]).
 ↪strip("'").replace("\\n", " ")
        date = re.sub('\'Created_at\': \'', '', re.findall(extracted_date_re,␣
 ↪j)[0])
        if not date in tweets_dict:
                tweets_dict[date] = {}
        tweets_dict[date][tweet_id] = text
```

1.0.5 English text is collected via langid

- All the tweets which are in english language only are assembled using langid library which classifies the text according to its language.

[44]:
```
# an empty dictionary is created to assemble all the text with their respective␣
 ↪dates
en_dict = {}
for date,text in tweets_dict.items():
    for k, v in text.items():
        # the text is checked for english language
        if langid.classify(v)[0] == 'en':
            if not date in en_dict:
```

```
                en_dict[date] = {}
            en_dict[date][k] = v
```

1.0.6 XML is created from dictionary

[46]:
```
def dicttoxml(data):
    result = "<data>"
    for date in data.keys():
        #string formatting is done accordig to the required format in xml
        result += "<tweets date=\"{}\">".format(date)
        for tid in data[date].keys():
            result += "<tweet id=\"{}\">{}</tweet>".format(tid,␣
↪data[date][tid])
        result += "</tweets>"
    result += "</data>"
    return result
```

1.0.7 Data passed to the function dictoxml

- The function dicttoxml created above, is sent a data which is en_dict to load the XML.

[47]:
```
xml_data = dicttoxml(en_dict)
```

1.0.8 Created XML file and xml data is writtedn to a file

[48]:
```
# a file is created to store the xml file with the required file name format
fout = open("./31901611.xml", "w", encoding="utf-8")
# the data is written in fout variable using .write()
fout.write(xml_data)
# the variable is closed
fout.close()
```