

Project 1 Report

Team Number:9

UMKC School of Computing and Engineering

Done by:

- Nageswara Rao Nandigam(nrnxxh9)
- Revanth Chakilam(rcww4)
- Syed Moin(slrp3)
- Devender Sarda(dspc8)

Project Tasks:

- **Main Requirements:**

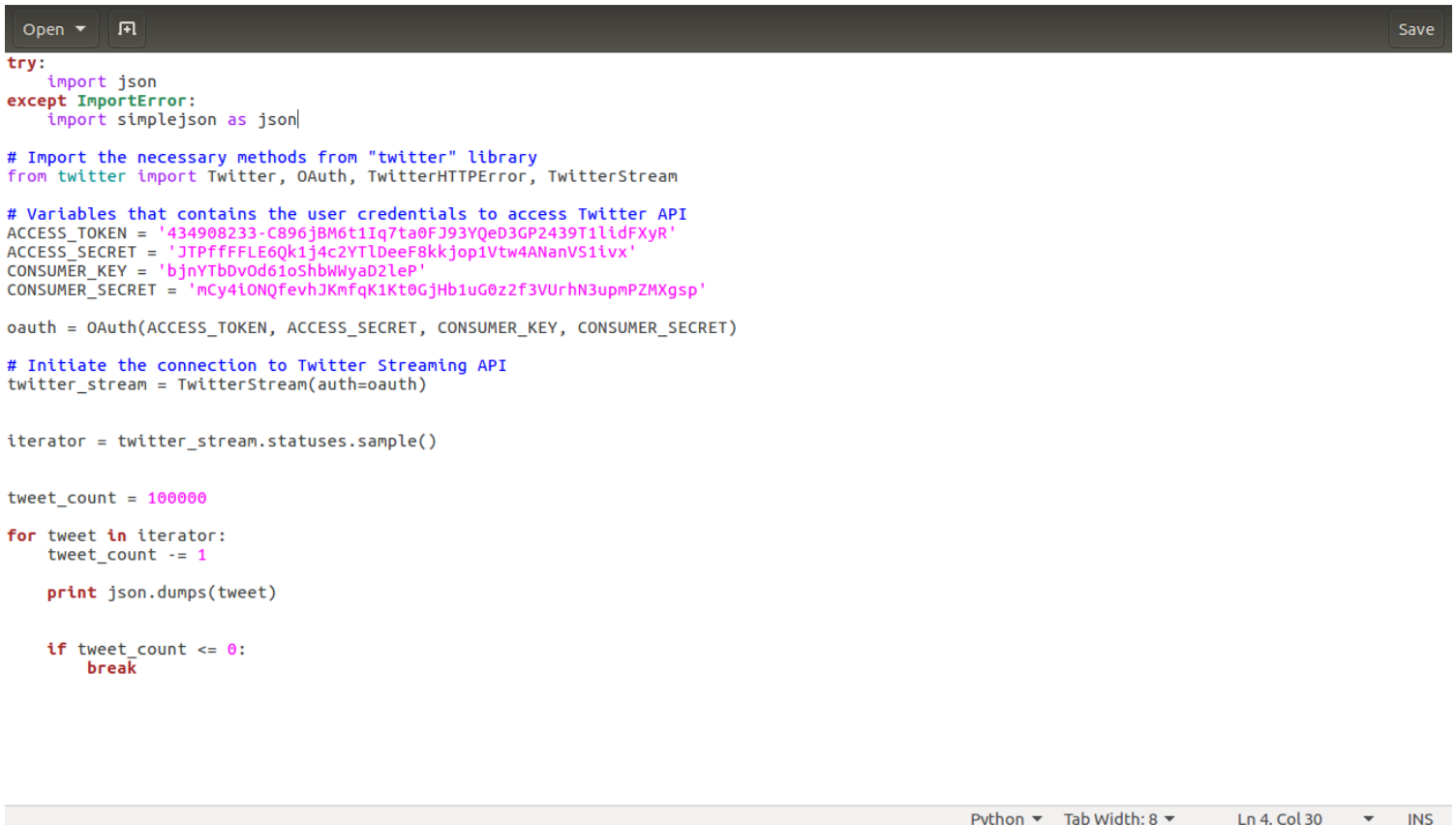
- Collect tweets in JavaScript Object Notation (JSON) format (at least 100K record).
 - Find the list of top ten used hashtags in your collection.
- Create a directory in HDFS for each hashtag from the top ten hashtag list.
 - Create additional two directories: "Others" and "None"
- Store the tweets on files in HDFS
 - If a tweet contains a hashtag from the top ten list, store the tweet in that hashtag's directory.
 - If a tweet contains one or more hashtags, but none of the hashtags are in the top ten list, store the tweet in the "Others" directory.
 - If a tweet does not contain a hashtag, store it in the "None" directory.

01/26/2017

Collect tweets in JavaScript Object Notation (JSON) format (at least 100K record)

We created a twitter development account to get consumer key and access keys and using this tokens we developed python script to collect tweets in JSON format. We programed a easy way to get the tweets as may we want by changing single parameter. So we are able to get 1 lakh tweets which stored in local disk file.

Screen shot of Pyhton program to collect tweets



```
try:
    import json
except ImportError:
    import simplejson as json

# Import the necessary methods from "twitter" library
from twitter import Twitter, OAuth, TwitterHTTPError, TwitterStream

# Variables that contains the user credentials to access Twitter API
ACCESS_TOKEN = '434908233-C896jBM6t1Iq7ta0FJ93YQeD3GP2439T1lidFXyR'
ACCESS_SECRET = 'JTPffFFLE6Qk1j4c2YTLDeeF8kkjop1Vtw4ANanVS1lvx'
CONSUMER_KEY = 'bjnYTbdvOd6ioShbWwyad2leP'
CONSUMER_SECRET = 'mCy4iONQfevhJKnfqK1Kt0GjHb1uG0z2f3VUrhN3upmPZMXgsp'

oauth = OAuth(ACCESS_TOKEN, ACCESS_SECRET, CONSUMER_KEY, CONSUMER_SECRET)

# Initiate the connection to Twitter Streaming API
twitter_stream = TwitterStream(auth=oauth)

iterator = twitter_stream.statuses.sample()

tweet_count = 100000

for tweet in iterator:
    tweet_count -= 1

    print json.dumps(tweet)

    if tweet_count <= 0:
        break
```

Python ▾ Tab Width: 8 ▾ Ln 4, Col 30 ▾ INS

01/26/2017

Output:

[illegible]

01/26/2017

Find the list of top ten used hashtags in your collection

we developed JAVA program to find out top ten hashtags. We followed coupled of steps to get top 10 hashtags

1) Read the input twitter collections file through bufferedreader reader line by line because each line is related to one tweet and store it in temporary String each time to parse like a JSONObject.

```
46  
47 // Reading the twitter collections file  
48 try  
49 {  
50     br=new BufferedReader(new FileReader("/home/nag/Desktop/twitter_1L.json"))  
51     str=new StringBuilder();  
52     String line=br.readLine();  
53     while(line!=null)  
54     {  
55         str.append(line);  
56         line=br.readLine();  
57     }  
58     result=str.toString();  
59     str.delete(0, str.length());  
60 }
```

2) We pass the above temp string to JSONObject as a parameter to read like JSON format data and iteratively we read JSON objects until we find hashtag array object ,after that we are retrieving hashtag text.

```
50  
51 //Retrieving the JSONObject from the file and getting hashtag texts  
52  
53 JSONObject object=new JSONObject(result);  
54 if(object.has("retweeted_status"))  
55 {  
56     JSONObject re=object.getJSONObject("retweeted_status");  
57     JSONObject e=re.getJSONObject("entities");  
58     JSONArray hash=e.getJSONArray("hashtags");  
59     if(hash.length()!=0){  
60         JSONObject t=hash.getJSONObject(0);  
61  
62         for(int i=0;i<1;i++)  
63         {  
64             String s=t.getString("text");  
65         }  
66     }
```

01/26/2017

3) After getting hashtag text, those values stored in hash map as a key-value pair. So, that if same hashtag appears next time it will increment its value by one. So, that we will get to know total repetitions of each hashtag.

```
76 |
77 | //Storing those hashtags in hashmap like key-value pair
78 |
79 |     if(map.containsKey(s.toLowerCase())){
80 |         int j=map.get(s.toLowerCase());
81 |         map.put(s.toLowerCase(),j+1);
82 |     }
83 |
84 |     else
85 |     {
86 | map.put(s.toLowerCase(),1);
87 |     }
88 |
89 |
90 |     }}}}
91 |
```

4) To find out top ten hashtags from hash map, we took another tree map which has function of sorting the elements by key, so we reverse the key as integer (total count) and hash tag as value. So finally, will get the sorted list of small count to highest count hash tags.

```
12 |
13 | Set<Entry<String, Integer>> set = map.entrySet();
14 | Iterator<Entry<String, Integer>> iterator = set.iterator();
15 | while(iterator.hasNext()) {
16 |     Map.Entry<String,Integer> mentry = (Map.Entry<String,Integer>)iterator.next()
17 |     if(mentry.getValue()>=10)
18 |     {
19 |         tr.put(mentry.getValue(),mentry.getKey());
20 |     }
21 | }
```

01/26/2017

5) As we mentioned above last entries had most frequency hashtag count, so we are iterating from last entry to till 10 entries above to get top list.

```
//Finding out top ten hashtags from the hashmap and displaying in console

tr.lastEntry();
int i=1;
ArrayList<String> a=new ArrayList<String>();
while(i<=10) {
    Map.Entry<Integer,String> mentry = (Map.Entry<Integer,String>)tr.lastEntry(

    System.out.print("key is: " + mentry.getKey() + " & Value is: ");
    System.out.println(mentry.getValue());
    a.add(mentry.getValue());
    tr.remove(mentry.getKey());
    i++;
}
```

Output:

<terminated> Hashtag [Java Application] /usr/lib/jvm/java-8-openjdk-amd64/bin/java (Feb 27, 2017, 12:36:39 AM)

```
key is: 202 & Value is: الوكروفلورز نما بين
key is: 117 & Value is: pillowtalk
key is: 104 & Value is: workfromhome
key is: 93 & Value is: harmonizers
key is: 69 & Value is: here
key is: 62 & Value is: 사설토토추천사이트
key is: 60 & Value is: videolove
key is: 56 & Value is: exsandohs
key is: 54 & Value is: kiskaniyorum
key is: 48 & Value is: iheartawards
```

Note: As you can see we found some hash tags are in other language too

01/26/2017

- Create a directory in HDFS for each hashtag from the top ten hashtag list.
- Create additional two directories: “Others” and “None”.

```
nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ bin/hadoop fs -ls /user/project
Found 13 items
-rw-r--r-- 3 nag supergroup 1072811 2017-02-27 00:40 /user/project/[الصيطن زربول فو ركول, pillowtalk, workfromhome, harmonizers, here, 사
설토투추천사이트, videolove, exsandohs, kiskaniyorum, iheartawards]
drwxr-xr-x - nag supergroup 0 2017-02-26 21:39 /user/project/ExsAndOhs
drwxr-xr-x - nag supergroup 0 2017-02-26 21:39 /user/project/Harmonizers
drwxr-xr-x - nag supergroup 0 2017-02-26 21:39 /user/project/Here
drwxr-xr-x - nag supergroup 0 2017-02-26 21:39 /user/project/Kiskaniyorum
drwxr-xr-x - nag supergroup 0 2017-02-27 00:56 /user/project/Others
drwxr-xr-x - nag supergroup 0 2017-02-26 21:39 /user/project/Pillowtalk
drwxr-xr-x - nag supergroup 0 2017-02-26 21:39 /user/project/VideoLove
drwxr-xr-x - nag supergroup 0 2017-02-26 21:39 /user/project/WorkFromHome
drwxr-xr-x - nag supergroup 0 2017-02-26 21:39 /user/project/iHeartAwards
drwxr-xr-x - nag supergroup 0 2017-02-27 00:56 /user/project/none
drwxr-xr-x - nag supergroup 0 2017-02-27 00:56 /user/project/الصيطن زربول فو ركول
```

- Store the tweets on files in HDFS
 - If a tweet contains a hashtag from the top ten list, store the tweet in that hashtag's directory.
 - If a tweet contains one or more hashtags, but none of the hashtags are in the top ten list, store the tweet in the “Others” directory.
 - If a tweet does not contain a hashtag, store it in the “None” directory.

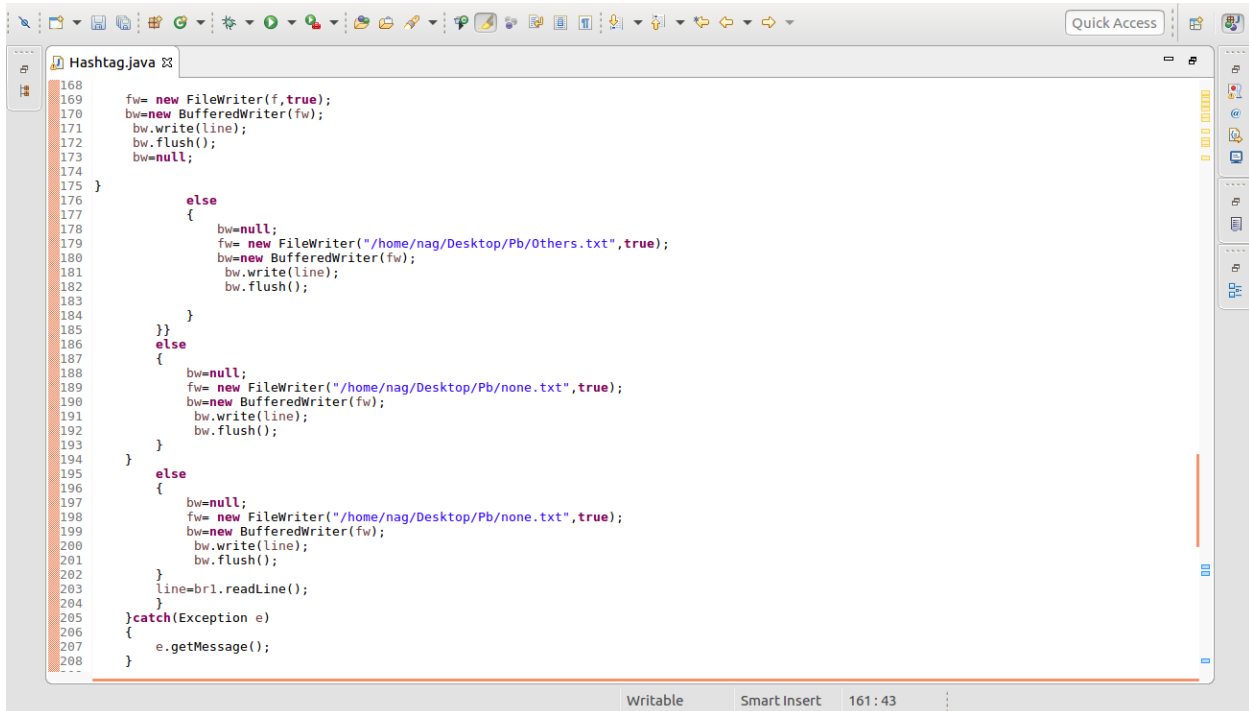
We developed a program to store each hashtag tweet information in respective hdfs hash tag directory. To integrate such java functionality to call hdfs filesystem we required to have lot more JAR dependency to be able to copy files in Hadoop. To make it work that way we added all those dependencies. Finally, we able to pass the contents to Huffs file system.

01/26/2017

```

12 //writing tweets to output files based on hashtag and putting it in respective hashtag local file
13
14 void fileout(ArrayList<String> a) throws IOException, URISyntaxException
15 {
16     BufferedWriter bw = null;
17     FileWriter fw;
18
19
20     try
21     {
22         @SuppressWarnings("resource")
23         BufferedReader brl=new BufferedReader(new FileReader("/home/nag/Desktop/twitter_1L.json"));
24         String line=brl.readLine();
25         while(line!=null)
26         {
27             JSONObject object=new JSONObject(line.toString());
28             if(object.has("retweeted_status"))
29             {
30                 JSONObject re=object.getJSONObject("retweeted_status");
31                 JSONObject e=re.getJSONObject("entities");
32                 JSONArray hash=e.getJSONArray("hashtags");
33                 if(hash.length()!=0){
34                     JSONObject t=hash.getJSONObject(0);
35
36                     for(int j=0;j<1;j++)
37                     {
38                         String s=t.getString("text");
39                         if(a.contains(s.toLowerCase()))
40                         {
41                             String name="/home/nag/Desktop/Pb/"+s+".txt";
42                             File f=new File(name);
43                             if (!f.exists()) {
44                                 f.createNewFile();
45                             }
46                         }
47                     }
48                 }
49             }
50         }
51     }
52 }

```



```

168 fw= new FileWriter(f,true);
169 bw=new BufferedWriter(fw);
170 bw.write(line);
171 bw.flush();
172 bw=null;
173
174 }
175
176 else
177 {
178     bw=null;
179     fw= new FileWriter("/home/nag/Desktop/Pb/Others.txt",true);
180     bw=new BufferedWriter(fw);
181     bw.write(line);
182     bw.flush();
183 }
184
185 }}
186
187 else
188 {
189     bw=null;
190     fw= new FileWriter("/home/nag/Desktop/Pb/none.txt",true);
191     bw=new BufferedWriter(fw);
192     bw.write(line);
193     bw.flush();
194 }
195
196 else
197 {
198     bw=null;
199     fw= new FileWriter("/home/nag/Desktop/Pb/none.txt",true);
200     bw=new BufferedWriter(fw);
201     bw.write(line);
202     bw.flush();
203 }
204 line=brl.readLine();
205 }
206 }catch(Exception e)
207 {
208     e.getMessage();
209 }

```


01/26/2017

```

    }
    //dumping files to hadoop system in each respective hashtag directory
    void hdfs(ArrayList<String> s) throws IOException, URISyntaxException
    {
        URI url=new URI("hdfs://localhost:9000"); //-----> (url where hdfs located)
        Configuration conf = new Configuration();

        FileSystem file1= FileSystem.get(url, conf);
        Path a = new Path("/home/nag/Desktop/Pb/none.txt");
        Path b = new Path("/user/project/none");
        file1.copyFromLocalFile(a,b);
        Path ab = new Path("/home/nag/Desktop/Pb/Others.txt");
        Path bc = new Path("/user/project/Others");
        file1.copyFromLocalFile(ab,bc);

        for(int i=0;i<s.size();i++)
        {
            Path x = new Path("/home/nag/Desktop/Pb/"+s.get(i)+".txt");
            Path y = new Path("/user/project/"+s.get(i));
            file1.copyFromLocalFile(x,y);
        }
    }
}

```

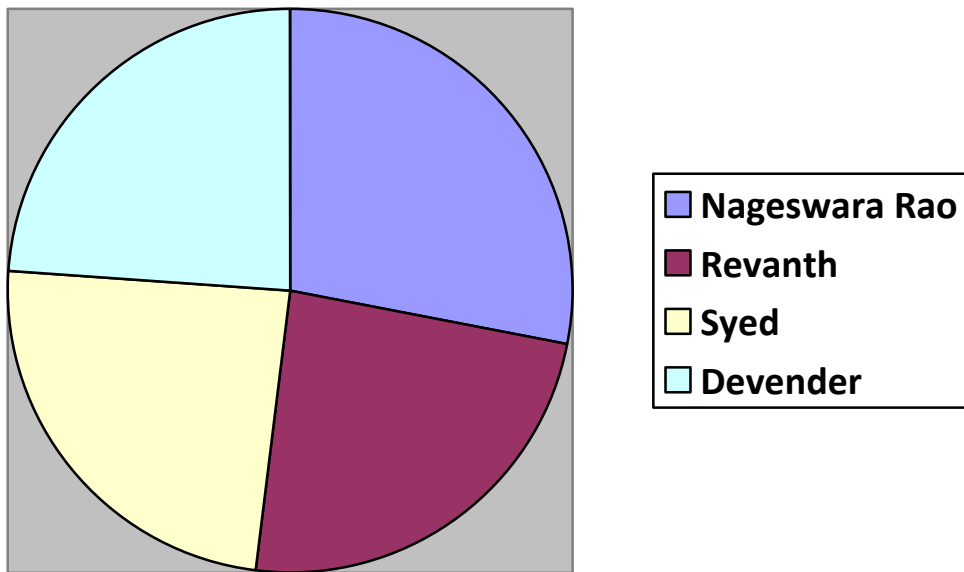
Output:

```

nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ bin/hadoop fs -ls /user/project/النصطن_زرولف_وركول
Found 1 items
-rw-r--r--  3 nag supergroup    1072811 2017-02-27 00:56 /user/project/النصطن_زرولف_وركول/النصطن_زرولف_وركول.txt
nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ bin/hadoop fs -ls /user/project/Others
Found 1 items
-rw-r--r--  3 nag supergroup    32314221 2017-02-27 00:56 /user/project/Others/Others.txt
nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ bin/hadoop fs -ls /user/project/none
Found 1 items
-rw-r--r--  3 nag supergroup    161409867 2017-02-27 00:56 /user/project/none/none.txt
nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ bin/hadoop fs -ls /user/project/WorkFromHome
Found 1 items
-rw-r--r--  3 nag supergroup      820832 2017-02-26 21:39 /user/project/WorkFromHome/WorkFromHome.txt
nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ bin/hadoop fs -ls /user/project/ExsAndohs
ls: '/user/project/ExsAndohs': No such file or directory
nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ bin/hadoop fs -ls /user/project/ExsAndOhs
Found 1 items
-rw-r--r--  3 nag supergroup    432848 2017-02-26 21:39 /user/project/ExsAndOhs/ExsAndOhs.txt
nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ bin/hadoop fs -ls /user/project/VideoLove
Found 1 items
-rw-r--r--  3 nag supergroup    469606 2017-02-26 21:39 /user/project/VideoLove/VideoLove.txt
nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ bin/hadoop fs -ls /user/project/Pillowtalk
Found 1 items
-rw-r--r--  3 nag supergroup    798982 2017-02-26 21:39 /user/project/Pillowtalk/Pillowtalk.txt
nag@nag-Dell-System-XPS-L502X:~/Downloads/hadoop-2.7.3$ █

```

Project Contribution:



References:

- 1) <https://dev.twitter.com/>
- 2) <http://stackoverflow.com/>
- 3) <http://hadoop.apache.org/>