LAB5

November 14, 2024

[53]:
```
inputpath = "/data/students/bigdata-01QYD/Lab2/"
inputRDD = sc.textFile(inputpath).cache()
```

0.1 TASK 1

[30]:
```
wordHORDD = inputRDD.filter(lambda word : word.startswith("ho")).cache()
```

[31]:
```
wordHORDD.take(10)
```

[31]:
```
['ho\t50',
 "ho'\t7",
 "ho'd\t1",
 "ho'ding\t1",
 'ho)\t1',
 'ho-ho\t4',
 "ho-ho's\t3",
 'ho-hos\t10',
 'ho-hum\t47',
 'ho-hum/nothing\t2']
```

[32]:
```
wordHORDD.count()
```

[32]:
```
1913
```

[33]:
```
freqWordRDD = wordHORDD.map(lambda freq : float(freq.split("\t")[1]))
```

[34]:
```
freqWordRDD.take(5)
```

[34]:
```
[50.0, 7.0, 1.0, 1.0, 1.0]
```

[35]:
```
maxfreqRDD = freqWordRDD.reduce(lambda val1,val2 : max (val1,val2))
```

[36]:
```
print (maxfreqRDD)
```

```
39399.0
```

0.2 TASK 2

[37]:
```
Ttrshold = 0.8*maxfreqRDD
```

[41]:
```
Word80freqRDD = wordHORDD.filter(lambda LineFW : float(LineFW.
↪split("\t")[1])>Ttrshold)
```

[43]:
```
Word80freqRDD.count()
```

[43]: 3

[44]:
```
MostFWRDD = Word80freqRDD.map(lambda word : word.split("\t")[0])
outputpathe = "Result_LAB5_2024_Task2"
```

[45]:
```
MostFWRDD.saveAsTextFile(outputpathe)
```

[47]:
```
MostFWRDD.take(5)
```

[47]: ['hot', 'how', 'however']

0.3 TASK 3

[77]:
```
Group0RDD = inputRDD.filter(lambda line0 : int(line0.split("\t")[1])>=0 and␣
↪int(line0.split("\t")[1])<100)
```

[78]:
```
Group1RDD = inputRDD.filter(lambda line1 : int(line1.split("\t")[1])>=100 and␣
↪int(line1.split("\t")[1])<200)
```

[79]:
```
Group2RDD = inputRDD.filter(lambda line2 : int(line2.split("\t")[1])>=200 and␣
↪int(line2.split("\t")[1])<300)
```

[80]:
```
Group3RDD = inputRDD.filter(lambda line3 : int(line3.split("\t")[1])>=300 and␣
↪int(line3.split("\t")[1])<400)
```

[93]:
```
Group4RDD = inputRDD.filter(lambda line4 : int(line4.split("\t")[1])>=400 and␣
↪int(line4.split("\t")[1])<500)
```

[82]:
```
Group5RDD = inputRDD.filter(lambda line5 : int(line5.split("\t")[1])>=500)
```

[83]:
```
Group0WordCountRDD = Group0RDD.map(lambda x : x.split("\t")[0]).count()
```

[84]:
```
print ("GROUP 0:--->",Group0WordCountRDD)
```

```
GROUP 0:---> 275062
```

[85]:
```
Group1WordCountRDD = Group1RDD.map(lambda x : x.split("\t")[0]).count()
```

[86]:
```
print ("GROUP 1:--->",Group1WordCountRDD)
```

```
GROUP 1:---> 3578
```

[87]:
```
Group2WordCountRDD = Group2RDD.map(lambda x : x.split("\t")[0]).count()
```

[88]:
```
print ("GROUP 2:--->",Group2WordCountRDD)
```

```
GROUP 2:---> 1527
```

[89]:
```
Group3WordCountRDD = Group3RDD.map(lambda x : x.split("\t")[0]).count()
```

[90]:
```
print ("GROUP 3:--->",Group3WordCountRDD)
```

```
GROUP 3:---> 936
```

[95]:
```
Group4WordCountRDD = Group4RDD.map(lambda x : x.split("\t")[0]).count()
```

[96]:
```
print ("GROUP 4:--->",Group4WordCountRDD)
```

```
GROUP 4:---> 636
```

[97]:
```
Group5WordCountRDD = Group5RDD.map(lambda x : x.split("\t")[0]).count()
```

[98]:
```
print ("GROUP 5:--->",Group5WordCountRDD)
```

```
GROUP 5:---> 4434
```