Assignment 2

Login:

ssh [jl10919@dumbo.hpc.nyu.edu](mailto:jl10919@dumbo.hpc.nyu.edu)

copy all files under directory:

scp -r jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2\_py.zip .

upload file to dumbo (put py code in /src)

scp task1a.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task1/task1a

scp task2a.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2a

edit .bash:

vi ~/.bashrc

add lines to .bash:

export PYSPARK\_PYTHON='/share/apps/python/3.6.5/bin/python'

export PYSPARK\_DRIVER\_PYTHON='/share/apps/python/3.6.5/bin/python'

update bash:

source ~/.bashrc

get data:

hfs -get /user/hc2660/hw2data/. .

start pyspark:

module load python/gnu/3.6.5

module load spark/2.4.0

pyspark

(如果打不开可能是进入账号后，开过2.2.0再开2.4.0出错，可以重进一遍服务器就可以了)

put the input file to hdfs:

hfs -put sherlock.txt

run the .py using spark:

spark-submit wordcount.py sherlock.txt

get output from hdfs:

hfs -getmerge wc.out wc.out

view the first 20 lines of the output:

head -n 20 wc.out

task1

Task 1a/Task 1a-SQL

scp task1a.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task1/task1a

scp task1a-sql.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task1/task1a

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task1/task1a/task1a.py /user/hc2660/hw2data/Trips.csv /user/hc2660/hw2data/Fares.csv

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task1/task1a/task1a-sql.py /user/hc2660/hw2data/Trips.csv /user/hc2660/hw2data/Fares.csv

查看端口（4042）有什么任务：netstat -apn | grep 4042

结束YARN上面跑的出现异常无法关闭的任务 ：yarn application -kill appid

A screenshot of a computer screen

Description automatically generated

关于spark inner join

<https://luminousmen.com/post/introduction-to-pyspark-join-types>

heroes.join(races, on="id", how="inner").show()

debug:

可以进入pyspark

line1 = sc.textFile('/home/jl10919/hw2data/Trips.csv')

line2 = sc.textFile('/home/jl10919/hw2data/Fares.csv')

一句句输入命令 用

for row in output.take(10): print(row)

看RDD是否对

看输出文件是否包含了head:

grep -i "medallion" task1a.out

sql-debug

Trips = spark.read.format('csv').options(header='true', inferschema='true').load('Trips.csv').createOrReplaceTempView("trip")

Fares = spark.read.format('csv').options(header='true', inferschema='true').load('Fares.csv').createOrReplaceTempView("fare")

task1b

scp task1b.py [jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task1/task1b](mailto:jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task1/task1b)

scp task1b-sql.py [jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task1/task1b](mailto:jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task1/task1b)

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task1/task1b/task1b.py /user/hc2660/hw2data/Fares.csv /user/hc2660/hw2data/Licenses.csv

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task1/task1b/task1b-sql.py /user/hc2660/hw2data/Fares.csv /user/hc2660/hw2data/Licenses.csv

debug

输入文件必须是在自己的hdfs下已有文件

line1 = sc.textFile('fares.csv')

line1 = line1.mapPartitions(lambda x: reader(x))

fares\_map = line1.map(lambda x: (x[0], (x[1], x[2], x[3], x[4], x[5], x[6], x[7], x[8], x[9], x[10])))

for row in fares\_map.take(10): print(row)

line2 = sc.textFile('licenses.csv')

line2 = line2.mapPartitions(lambda x: reader(x))

licenses\_map = line2.map(lambda x: (x[0], (x[1], x[2], x[3], x[4], x[5], x[6], x[7], x[8], x[9], x[10], x[11], x[12], x[13], x[14], x[15])))

line2 = line2.mapPartitions(lambda x: reader(x))

for row in licenses\_map.take(10): print(row)

result = fares\_map.join(licenses\_map)

for row in result.take(10): print(row)

result = result.sortBy(lambda x: (x[0], x[1][0][0], x[1][0][2]))

for row in result.take(10): print(row)

output = result.map(lambda x: x[0] + ',' + ','.join(value for value in x[1][0]) + ',' + '\"' + x[1][1][0] + '\"' + ',' + ','.join(value for value in x[1][1][1:]))for row in output.take(10): print(row)

sql

SELECT format\_string("Hello World %d %s", 100, "days");　　Hello World 100 days

Format\_string():https://www.cnblogs.com/feiyumo/p/8763186.html

一些操作：<https://blog.csdn.net/sparkexpert/article/details/51042970>

task2a

scp task2a.py [jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2a](mailto:jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2a)

scp task2a-sql.py [jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2a](mailto:jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2a)

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task2/task2a/task2a.py /user/jl10919/task1a.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task2/task2a/task2a-sql.py /user/jl10919/task1a-sql.out

debug:

TypeError: must be str, not int:

You can only append strings to strings using the "+" operator, not integers.

line 8, in maptorange

'list' object has no attribute 'flatMap'

不可连续用saveastext追加保存到文件中

import sys  
from pyspark import SparkContext  
from csv import reader

line1 = sc.textFile('alltrip')

line1 = line1.mapPartitions(lambda x: reader(x))

range1 = line1.map(lambda x: ('[0,5)', (1 if 0 <= float(x[-6]) < 5 else 0))).reduceByKey(lambda x, y: x + y).map(lambda x: x[0] + "," + str(x[1]))

for row in range1.take(10): print(row)

range1 = range1.sortBy(lambda x: int(re.search('\d+',str(x[0])).group()))

out = sc.textFile('task2a.out')

out = out.mapPartitions(lambda x: reader(x))

for row in out.take(10): print(row)

out = out.sortBy(lambda x: int(x[0][1:]))

one = out.take(1)

debug

import sys  
from pyspark.sql import \*  
from pyspark.sql.functions import \*

all\_trips = spark.read.format('csv').options(header='false',inferschema='true').load('task1a-sql.out')

all\_trips.createOrReplaceTempView("alltrip")

spark.sql ("SELECT t1.p1 AS num1,t2.p2 AS num2 FROM

(SELECT COUNT(\_c15) AS p1 FROM alltrip WHERE \_c15>=0 and \_c15 <=5") t1

INNER JOIN (SELECT COUNT(\_c15) AS p2 FROM alltrip where \_c15>5 and \_c15 <15) t2)

Sql 统计区间

<https://blog.csdn.net/qq_39964694/article/details/79758330>

https://blog.csdn.net/kai\_wei/article/details/51789812

task2b

scp task2b.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2b

scp task2b-sql.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2b

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task2/task2b/task2b.py /user/jl10919/task1a.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task2/task2b/task2b-sql.py /user/jl10919/task1a.out

task2c

scp task2c.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2c

scp task2c-sql.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2c

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task2/task2c/task2c.py /user/jl10919/task1a.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task2/task2c/task2c-sql.py /user/jl10919/task1a-sql.out

task2d

scp task2d.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2d

scp task2d-sql.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task2/task2d

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task2/task2d/task2d.py /user/jl10919/task1a.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task2/task2d/task2d-sql.py /user/jl10919/task1a-sql.out

import sys  
from pyspark import SparkContext  
from csv import reader

from operator import add

line1 = sc.textFile('alltrip')

line1 = line1.mapPartitions(lambda x: reader(x))

c1 = line1.map(lambda x: ((x[0],x[3][:10]),1)).reduceByKey(add).sortBy(lambda x: (x[0][0],x[0][1]))

for row in c1.take(10): print(row)

...

(('00005007A9F30E289E760362F69E4EAD', '2013-08-01'), 48)

(('00005007A9F30E289E760362F69E4EAD', '2013-08-02'), 41)

(('00005007A9F30E289E760362F69E4EAD', '2013-08-03'), 44)

(('00005007A9F30E289E760362F69E4EAD', '2013-08-04'), 47)

(('00005007A9F30E289E760362F69E4EAD', '2013-08-05'), 46)

(('00005007A9F30E289E760362F69E4EAD', '2013-08-06'), 60)

(('00005007A9F30E289E760362F69E4EAD', '2013-08-07'), 55)

(('000318C2E3E6381580E5C99910A60668', '2013-08-01'), 17)

(('000318C2E3E6381580E5C99910A60668', '2013-08-05'), 15)

(('000318C2E3E6381580E5C99910A60668', '2013-08-06'), 28)

>>> c2 = c1.map(lambda x: (x[0][0],(1,x[1]))).reduceByKey(lambda x,y: (x[0]+y[0],x[1]+y[1])).sortByKey()

>>> for row in c2.take(10): print(row)

...

('00005007A9F30E289E760362F69E4EAD', (7, 341))

('000318C2E3E6381580E5C99910A60668', (4, 111))

('000351EDC735C079246435340A54C7C1', (7, 330))

('0009986BDBAB2F9A125FEF49D0BFCCDD', (7, 366))

('00115F46520039845A5F719C979BEA45', (7, 206))

('00153E36140C5B2A84EA308F355A7925', (4, 127))

('001C8EC421C9BE57D08576617465401A', (7, 262))

('001D3B86C2ACDEE4D1B98AFE52969F3D', (7, 152))

('001DFAC01BC0A32F48C3769DD1414778', (7, 257))

('00244196AAA321571762E0CCC55EEAD9', (5, 92))

>>> output = c2.map(lambda x: x[0] + "," + str(x[1][1]) + "," + str(x[1][0]) + "," + "{0:.2f}".format(float(x[1][1])/float(x[1][0])))

>>>

>>> for row in output.take(10): print(row)

...

00005007A9F30E289E760362F69E4EAD,341,7,48.71

000318C2E3E6381580E5C99910A60668,111,4,27.75

000351EDC735C079246435340A54C7C1,330,7,47.14

0009986BDBAB2F9A125FEF49D0BFCCDD,366,7,52.29

00115F46520039845A5F719C979BEA45,206,7,29.43

00153E36140C5B2A84EA308F355A7925,127,4,31.75

001C8EC421C9BE57D08576617465401A,262,7,37.43

001D3B86C2ACDEE4D1B98AFE52969F3D,152,7,21.71

001DFAC01BC0A32F48C3769DD1414778,257,7,36.71

00244196AAA321571762E0CCC55EEAD9,92,5,18.40

TypeError: 'PipelinedRDD' object is not iterable

用map处理上一个map的结果即可

Map的结果最好写成line1.map(lambda x: ((x[0],x[3]),1)) ,(key,value)，括号包住。

这里的key是(x[0],x[3])， value是1

注意，key是第一个元素。比如map的结果是(a,(b,c))，则key为x[0]=a，而x[1]=(b,c),x[1][0]=b,x[1][1]=c

当reducebykey的时候，函数里面的参数x相当于只剩下了value，相当于只有(b,c)

所以lambda x: x[0],x[1] 则是 x=(b,c),x[0]=b,x[1]=c

当map用+相连的时候，对于int x等数字，不能用‘+x’ 而是用‘,x’表示放入这个元素，否则就’+ str(x) +’

Task3a

scp task3a.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task3/task3a

scp task3a-sql.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task3/task3a

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task3/task3a/task3a.py /user/jl10919/task1a.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task3/task3a/task3a-sql.py /user/jl10919/task1a-sql.out

task3b

scp task3b.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task3/task3b

scp task3b-sql.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task3/task3b

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task3/task3b/task3b.py /user/jl10919/task1a.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task3/task3b/task3b-sql.py /user/jl10919/task1a-sql.out

task3c

scp task3c.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task3/task3c

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task3/task3c/task3c.py /user/jl10919/task1a.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task3/task3c/task3c-sql.py /user/jl10919/task1a.out

task3d

scp task3d.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task3/task3d

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task3/task3d/task3d.py /user/jl10919/task1a.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task3/task3d/task3d-sql.py /user/jl10919/task1a-sql.out

task4a

scp task4a.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task4/task4a

scp task4a-sql.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task4/task4a

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task4/task4a/task4a.py /user/jl10919/task1b.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task4/task4a/task4a-sql.py /user/jl10919/task1b-sql.out

debug

import sys  
from pyspark import SparkContext  
from csv import reader

from operator import add

line1 = sc.textFile('task1b.out')

line1 = line1.mapPartitions(lambda x: reader(x))

for row in line1.take(10): print(row)

sql null:

<https://www.jianshu.com/p/4dad54c37d0d>

task 4b

scp task4b.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task4/task4b

scp task4b-sql.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task4/task4b

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task4/task4b/task4b.py /user/jl10919/task1b.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task4/task4b/task4b.py /user/jl10919/task1b-sql.out

task4c

scp task4c.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task4/task4c

scp task4c-sql.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task4/task4c

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task4/task4c/task4c.py /user/jl10919/task1b.out

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task4/task4c/task4c-sql.py /user/jl10919/task1b-sql.out

sortby():

**第一个参数**是一个函数，该函数的也有一个带T泛型的参数，返回类型和RDD中元素的类型是一致的；  
　　**第二个参数**是ascending，从字面的意思大家应该可以猜到，是的，这参数决定排序后RDD中的元素是升序还是降序，默认是true，也就是升序；  
　　**第三个参数**是numPartitions，该参数决定排序后的RDD的分区个数，默认排序后的分区个数和排序之前的个数相等，即为this.partitions.size。

scp task4c.py jl10919@dumbo.hpc.nyu.edu:~jl10919/hw2/task4/task4c

spark-submit --conf spark.pyspark.python=/share/apps/python/3.6.5/bin/python /home/jl10919/hw2/task4/task4c/task4c.py /user/jl10919/task1b.out

关于sql

debug

import sys  
from pyspark.sql import \*  
from pyspark.sql.functions import \*

all\_trips = spark.read.format('csv').options(header='false',inferschema='true').load('task1a-sql.out')

all\_trips = spark.read.format('csv').options(header='false',inferschema='true').load('/user/kc4152/data/test')

all\_trips.createOrReplaceTempView("alltrip")

spark.sql ("select \_c2 from alltrip").show()

trips\_lic = spark.read.format('csv').options(header='false',inferschema='true').load('task1b-sql.out')

trips\_lic.createOrReplaceTempView("triplic")

A screenshot of a cell phone

Description automatically generated