درس : Big Data

استاد : خانم دکتر صفری

دانشجو : رقیه یزدانی

تمرین : MapReduce

MapReduce

انجام wordCount با استفاده از MapReduce

برای انجام این مثال، روی متنی از کتاب شکسپیر که برای تمرین hadoop دانلود شد، کار کرده و سعی می‌شود که دفعات تکرار کلمات در این فایل شمارش شود و از زبان پایتون برای نوشتن برنامه mapper و reducer استفاده شده است. فایل wordcount-mapper.py بصورت زیر، به ازای هر خط ورودی، کلمات آن را جدا کرده و لیستی از (key,value) که key هر کلمه و value مقدار 1 است را برمی‌گرداند. در ادامه مثالی از اجرای آن هم آورده شده است:

```
negar@negar-VirtualBox:~$ cat HW/wordcount-mapper.py
#!/usr/bin/env python
import sys

#--- get all lines from stdin ---
for line in sys.stdin:
    #--- remove leading and trailing whitespace---
    line = line.strip()

    #--- split the line into words ---
    words = line.split()

    #--- output tuples [word, 1] in tab-delimited format---
    for word in words:
        print ('%s\t%s' % (word, "1"))
negar@negar-VirtualBox:~$
```

```
negar@negar-VirtualBox:~$ cat test.txt
This is a sample text for map-reduce job.
This is an example for big-data course.
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$ cat test.txt | python3 HW/wordcount-mapper.py
This    1
is      1
a       1
sample  1
text    1
for     1
map-reduce      1
job.    1
This    1
is      1
an      1
example 1
for     1
big-data        1
course. 1
negar@negar-VirtualBox:~$
```

فایل wordcount-reducer.py هم لیستی از (key,value) ها را می‌گیرد و یک map می‌سازد که هر بار مقدار value هر pair را به count معادل آن کلمه اضافه می‌کند.

```
negar@negar-VirtualBox:~$ cat HW/wordcount-reducer.py
#!/usr/bin/env python
import sys

# maps words to their counts
word2count = {}

# input comes from STDIN
for line in sys.stdin:
    # remove leading and trailing whitespace
    line = line.strip()

    # parse the input we got from mapper.py
    word, count = line.split('\t', 1)
    # convert count (currently a string) to int
    try:
        count = int(count)
    except ValueError:
        continue

    try:
        word2count[word] = word2count[word]+count
    except:
        word2count[word] = count

# write the tuples to stdout
# Note: they are unsorted
for word in word2count.keys():
    print ('%s\t%s'% ( word, word2count[word] ))
negar@negar-VirtualBox:~$
```

در ادامه برای اطمینان از صحت عملکرد برنامه، برای فایل test.txt ابتدا برنامه mapper اجرا شده و سپس key ها مرتب شده و به reducer داده می‌شود. این عملیات بصورت local انجام می‌شود و خارج از hdfs بوده و تنها برای تست برنامه است :

```
negar@negar-VirtualBox:~$ cat test.txt | python3 HW/wordcount-mapper.py | sort | python3 HW/wordcount-reducer.py
a       1
an      1
big-data        1
course. 1
example 1
for     2
is      2
job.    1
map-reduce      1
sample  1
text    1
This    2
negar@negar-VirtualBox:~$
```

حال که از درستی عملکرد برنامه mapper و reducer اطمینان حاصل شد، باید برای فایل شکسپیر و روی hdfs این کار انجام شود. فایل متن با نام t8.shakespeare.txt در مسیر /homeworks/shakespeare روی hdfs قرار داده شده است و برنامه های mapper و reducer روی ماشین local و در مسیر HW/ قرار دارند. با دستور hadoop jar بصورتی

که در شکل زیر نشان داده شده، با mapper- برنامه mapper و با reducer- برنامه reducer و با input- مسیر فایل ورودی روی hdfs و با output- مسیر خروجی را تعیین می‌کنیم.

```
negar@negar-VirtualBox:~$ hdfs dfs -ls /homeworks/shakespeare
Found 1 items
-rw-r--r--   1 negar supergroup    5458199 2020-05-29 12:20 /homeworks/shakespeare/t8.shakespeare.txt
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$ hadoop jar hadoop-3.2.1/share/hadoop/tools/lib/hadoop-streaming-3.2.1.jar -mapper
'python3 HW/wordcount-mapper.py' -reducer 'python3 HW/wordcount-reducer.py' -input /homeworks/shakespeare -o
utput /homeworks/output-shakespeare
```

پس از اجرای دستور بالا؛ همانطور که در شکل زیر مشخص شده، فایلی به نام part-00000 در مسیر تعیین شده برای خروجی، ساخته می‌شود که محتوای چند سطر اول آن در زیر؛ نشان داده شده است.

```
negar@negar-VirtualBox:~$ hdfs dfs -ls /homeworks/
Found 2 items
drwxr-xr-x   - negar supergroup          0 2020-05-29 14:10 /homeworks/output-shakespeare
drwxr-xr-x   - negar supergroup          0 2020-05-29 14:08 /homeworks/shakespeare
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$ hdfs dfs -ls /homeworks/output-shakespeare
Found 2 items
-rw-r--r--   1 negar supergroup          0 2020-05-29 14:10 /homeworks/output-shakespeare/_SUCCESS
-rw-r--r--   1 negar supergroup     717768 2020-05-29 14:10 /homeworks/output-shakespeare/part-00000
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$ hdfs dfs -head /homeworks/output-shakespeare/part-00000
2020-05-29 14:19:49,246 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = fa
lse, remoteHostTrusted = false
"       241
"'Tis   1
"A      4
"AS-IS".        1
"Air,"  1
"Alas,  1
"Amen"  2
"Amen"? 1
"Amen," 1
"And    1
"Aroint 1
"B      1
"Black  1
"Break  1
"Brutus"        1
"Brutus,        2
```

-numReduceTasks

هنگام استفاده از دستور hadoop jar می‌توان از numReduceTasks- نیز استفاده کرد، مقدار پیش‌فرض آن 1 است و به معنای تعداد reducer ها می‌باشد. اگر به آن مقدار 0 بدهیم، در واقع مرحله reducer را حذف کرده‌ایم و اجرا نخواهد شد. تمرین بالا را با این حالت نیز اجرا می کنیم :

```
negar@negar-VirtualBox:~$ hadoop jar hadoop-3.2.1/share/hadoop/tools/lib/hadoop-streaming-3.2.1.jar -mapper
'python3 HW/wordcount-mapper.py' -reducer 'python3 HW/wordcount-reducer.py' -input /homeworks/shakespeare -o
utput /homeworks/output-shakespeare -numReduceTasks 0
```

خروجی زیر نشان می دهد؛ که تنها مرحله mapper اجرا شده است:

```
negar@negar-VirtualBox:~$ hdfs dfs -head /homeworks/output-shakespeare-0/part-00000
2020-05-29 14:29:00,795 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = fa
lse, remoteHostTrusted = false
This    1
is      1
the     1
100th   1
Etext   1
file    1
presented       1
by      1
Project 1
Gutenberg,      1
and     1
```

یکبار دیگر این تمرین با مقدار 2 برای پارامتر numReduceTasks- اجرا می‌کنیم :

```
negar@negar-VirtualBox:~$ hadoop jar hadoop-3.2.1/share/hadoop/tools/lib/hadoop-streaming-3.2.1.jar -mapper
'python3 HW/wordcount-mapper.py' -reducer 'python3 HW/wordcount-reducer.py' -input /homeworks/shakespeare -o
utput /homeworks/output-shakespeare-2 -numReduceTasks 2
```

این بار دو فایل خروجی می سازد، فایلهای part-00000 و part-00001 که هر کدام خروجی یک reducer است:

```
negar@negar-VirtualBox:~$ hdfs dfs -ls /homeworks/output-shakespeare-2
Found 3 items
-rw-r--r--   1 negar supergroup          0 2020-05-29 14:32 /homeworks/output-shakespeare-2/_SUCCESS
-rw-r--r--   1 negar supergroup     359057 2020-05-29 14:32 /homeworks/output-shakespeare-2/part-00000
-rw-r--r--   1 negar supergroup     358711 2020-05-29 14:32 /homeworks/output-shakespeare-2/part-00001
```

بررسی فایلها نشان می‌دهد، کلیدها در هر کدام از این فایلها، متفاوت هستند و در واقع ترکیب این دو فایل معادل خروجی زمانی است که برنامه با یک reducer اجرا شد. به عبارت دیگر برنامه چه با یک reducer و چه با دو reducer اجرا شود، تعداد دفعات تکرار هر کلمه در کل متن شمرده می‌شود و نتیجه برای هر کلمه در هر دو حالت یکسان است.

انجام join با استفاده از MapReduce

این بخش شامل دو تمرین جدا است که به نحوی تکمیل کننده هم هستند.

تمرین اول :

دو فایل داریم به نامهای join1_FileA.txt و join1_FileB.txt که در ادامه محتوای هر کدام آورده شده است. هر سطر از فایل اول، شامل یک کلمه به عنوان key و یک عدد به عنوان value است که با , از هم جدا شده‌اند و هر سطر از فایل دوم، شامل یک تاریخ و یک کلمه به عنوان key و یک عدد بعنوان value که با , از هم جدا شده‌اند.

Join1_FileA.txt

```
able,991
about,11
burger,15
actor,22
```

join1_FileB.txt

```
Jan-01 able,5
Feb-02 about,3
Mar-03 about,8
Apr-04 able,13
Feb-22 actor,3
Feb-23 burger,5
Mar-08 burger,2
Dec-15 able,100
```

هدف این است که این دو فایل روی "کلمه" با هم join شوند و خروجی ترکیبی بسازند. برای انجام آن با استفاده از MapReduce نیاز به یک برنامه mapper و یک برنامه reducer داریم. کد پایتون برنامه mapper در زیر آورده شده است :

```
  GNU nano 2.9.3                                  join1_mapper.py

#!/usr/bin/env python3
import sys

for line in sys.stdin:
    line       = line.strip()   #strip out carriage return
    key_value  = line.split(",")   #split line, into key and value, returns a list
    key_in     = key_value[0].split(" ")   #key is first item in list
    value_in   = key_value[1]   #value is 2nd item

    if len(key_in)>=2:           #if this entry has <date word> in key
        date = key_in[0]      #now get date from key field
        word = key_in[1]
        value_out = date+" "+value_in     #concatenate date, blank, and value_in
        print( '%s\t%s' % (word, value_out) )  #print a string, tab, and string
    else:   #key is only <word> so just pass it through
        print( '%s\t%s' % (key_in[0], value_in) )  #print a string tab and string
```

نتیجه اجرای برنامه join1_mapper.py روی فایلهای ورودی بصورت زیر است. همانطور که مشخص است با فایل اول که کلید، یک کلمه است، کاری ندارد اما برای فایل دوم، تاریخ؛ از بخش key جدا شده و به بخش value اضافه شده است :

```
negar@negar-VirtualBox:~/HW$ cat join1_*.txt | python3 join1_mapper.py
able    991
about   11
burger  15
actor   22
able    Jan-01 5
about   Feb-02 3
about   Mar-03 8
able    Apr-04 13
actor   Feb-22 3
burger  Feb-23 5
burger  Mar-08 2
able    Dec-15 100
negar@negar-VirtualBox:~/HW$
```

سپس نیاز به یک برنامه reducer داریم که در شکل زیر کد پایتون آن آورده شده است :

```
 GNU nano 2.9.3                    join1_reducer.py                    Modified

#!/usr/bin/env python3
import sys

prev_word          = "  "               #initialize previous word  to blank string
months             = ['Jan','Feb','Mar','Apr','Jun','Jul','Aug','Sep','Nov','Dec']

dates_to_output = []
day_cnts_to_output = [] #an empty list of day counts for a given word

line_cnt = 0  #count input lines

for line in sys.stdin:
    line = line.strip()       #strip out carriage return
    key_value  = line.split('\t')   #split line, into key and value, returns a list
    line_cnt   = line_cnt+1

    curr_word  = key_value[0]         #key is first item in list, indexed by 0
    value_in   = key_value[1]         #value is 2nd item
    if curr_word != prev_word:
        if line_cnt>1:
            for i in range(len(dates_to_output)):  #loop thru dates, indexes start at 0
                print('{0} {1} {2} {3}'.format(dates_to_output[i], prev_word, day_cnts_to_output[i], curr_word_total_cnt))
                #now reset lists
            dates_to_output   =[]
            day_cnts_to_output=[]
        prev_word = curr_word  #set up previous word for the next set of input lines

    if (value_in[0:3] in months):
        date_day =value_in.split() #split the value field into a date and day-cnt

        dates_to_output.append(date_day[0])
        day_cnts_to_output.append(date_day[1])
    else:
        curr_word_total_cnt = value_in

for i in range(len(dates_to_output)):  # loop thru dates, indexes start at 0
    print('{0} {1} {2} {3}'.format(dates_to_output[i],prev_word,day_cnts_to_output[i],curr_word_total_cnt))
```

در شکل زیر ابتدا مرحله sort بعد از اجرای mapper انجام شده و خروجی آن نشان داده است و سپس مرحله reducer با اجرای برنامه بالا انجام شده و نتیجه کار که حاصل join اطلاعات می باشد، آورده شده است.

```
negar@negar-VirtualBox:~/HW$ cat join1_*.txt | python3 join1_mapper.py | sort
able    991
able    Apr-04 13
able    Dec-15 100
able    Jan-01 5
about   11
about   Feb-02 3
about   Mar-03 8
actor   22
actor   Feb-22 3
burger  15
burger  Feb-23 5
burger  Mar-08 2
negar@negar-VirtualBox:~/HW$ cat join1_*.txt | python3 join1_mapper.py | sort | python3 join1_reducer.py
Apr-04 able 13 991
Dec-15 able 100 991
Jan-01 able 5 991
Feb-02 about 3 11
Mar-03 about 8 11
Feb-22 actor 3 22
Feb-23 burger 5 15
Mar-08 burger 2 15
negar@negar-VirtualBox:~/HW$
```

همین کار را مشابه قبل باید روی hdfs انجام دهیم، که لازم است ابتدا فایلهای داده، به hdfs منتقل شوند و سپس برنامه MapReduce اجرا شود. در شکلهای زیر، مراحل این کار نشان داده است و سپس خروجی کار نمایش داده شده است.

```
negar@negar-VirtualBox:~$ hdfs dfs -mkdir /homeworks/join1-data
negar@negar-VirtualBox:~$ hdfs dfs -copyFromLocal HW/join1_*.txt /homeworks/join1-data
2020-05-30 20:34:22,913 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = fa
lse, remoteHostTrusted = false
2020-05-30 20:34:23,063 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = fa
lse, remoteHostTrusted = false
negar@negar-VirtualBox:~$ hdfs dfs -ls /homeworks/join1-data
Found 2 items
-rw-r--r--   1 negar supergroup         37 2020-05-30 20:34 /homeworks/join1-data/join1_FileA.txt
-rw-r--r--   1 negar supergroup        122 2020-05-30 20:34 /homeworks/join1-data/join1_FileB.txt
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$ hadoop jar hadoop-3.2.1/share/hadoop/tools/lib/hadoop-streaming-3.2.1.jar -mapper
'python3 HW/join1_mapper.py' -reducer 'python3 HW/join1_reducer.py' -input /homeworks/join1-data -output /ho
meworks/join1-output
2020-05-30 20:35:20,180 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2020-05-30 20:35:20,258 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2020-05-30 20:35:20,258 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2020-05-30 20:35:20,283 WARN impl.MetricsSystemImpl: JobTracker metrics system already initialized!
2020-05-30 20:35:20,610 INFO mapred.FileInputFormat: Total input files to process : 2
2020-05-30 20:35:20,631 INFO mapreduce.JobSubmitter: number of splits:2
```

```
negar@negar-VirtualBox:~$ hdfs dfs -ls /homeworks/join1-output
Found 2 items
-rw-r--r--   1 negar supergroup          0 2020-05-30 20:35 /homeworks/join1-output/_SUCCESS
-rw-r--r--   1 negar supergroup        157 2020-05-30 20:35 /homeworks/join1-output/part-00000
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$ hdfs dfs -cat /homeworks/join1-output/part-00000
2020-05-30 20:36:34,672 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = fa
lse, remoteHostTrusted = false
Dec-15 able 100 991
Apr-04 able 13 991
Jan-01 able 5 991
Mar-03 about 8 11
Feb-02 about 3 11
Feb-22 actor 3 22
Mar-08 burger 2 15
Feb-23 burger 5 15
negar@negar-VirtualBox:~$
```

تمرین دوم :

برای این بخش، برنامه پایتون به نام make_join2data.py در اختیار قرار داده شده تا بکمک آن بتوان فایلهای داده را ایجاد کرد. کد این برنامه در ادامه آمده است :

make_join2data.py

```
#!/usr/bin/env python
import sys
# ------------------------------------------------------------------------
#  (make_join2data.py) Generate a random combination of titles and viewer
    counts, or channels
# this is a simple version of a congruential generator,
#   not a great random generator but enough
# ------------------------------------------------------------------------
chans   = ['ABC','DEF','CNO','NOX','YES','CAB','BAT','MAN','ZOO','XYZ','BOB']
sh1 =['Hot','Almost','Hourly','PostModern','Baked','Dumb','Cold','Surreal'
    ,'Loud']
sh2 =['News','Show','Cooking','Sports','Games','Talking','Talking']
vwr =range(17,1053)
chvnm=sys.argv[1]  #get number argument, if its n, do numbers not channels,
lch=len(chans)
lsh1=len(sh1)
lsh2=len(sh2)
lvwr=len(vwr)
ci=1
s1=2
s2=3
vwi=4
ri=int(sys.argv[3])
for i in range(0,int(sys.argv[2])):  #arg 2 is the number of lines to output
    if chvnm=='n':  #no numuber
        print('{0}_{1},{2}'.format(sh1[s1],sh2[s2],chans[ci]))
    else:
        print('{0}_{1},{2}'.format(sh1[s1],sh2[s2],vwr[vwi]))
    ci=(5*ci+ri) % lch
    s1=(4*s1+ri) % lsh1
    s2=(3*s1+ri+i) % lsh2
    vwi=(2*vwi+ri+i) % lvwr

    if (vwi==4): vwi=5
```

این برنامه داده هایی تصادفی ایجاد می کند که با شش بار اجرای آن بصورت زیر، شش فایل تولید خواهد شد :

```
python3 make_join2data.py y 1000 13 > join2_gennumA.txt
python3 make_join2data.py y 2000 17 > join2_gennumB.txt
python3 make_join2data.py y 3000 19 > join2_gennumC.txt
python3 make_join2data.py n 100  23 > join2_genchanA.txt
python3 make_join2data.py n 200  19 > join2_genchanB.txt
python3 make_join2data.py n 300  37 > join2_genchanC.txt
```

در ادامه نمونه ای از چند خط ابتدایی داده های فایل join2_gennumA.txt آورده شده است که شامل نام یک برنامه تلویزیونی و تعداد دفعات تماشای آن می‌باشد.

```
negar@negar-VirtualBox:~/HW$ head join2_gennumA.txt
Hourly_Sports,21
PostModern_Show,38
Surreal_News,73
Dumb_Cooking,144
Cold_Talking,287
Almost_Talking,574
Loud_News,113
Hot_Talking,228
Baked_Games,459
Hourly_Talking,922
negar@negar-VirtualBox:~/HW$
```

در تصویر زیر، چند خط ابتدایی اطلاعات فایل join2_genchanA.txt آمده است که شامل نام برنامه تلویزیونی و نام channel پخش آن برنامه است :

```
negar@negar-VirtualBox:~/HW$ head join2_genchanA.txt
Hourly_Sports,DEF
Baked_News,BAT
PostModern_Talking,XYZ
Loud_News,CNO
Almost_Show,ABC
Hot_Talking,DEF
Dumb_Show,BAT
Surreal_Show,XYZ
Cold_Talking,CNO
Hourly_Cooking,ABC
negar@negar-VirtualBox:~/HW$
```

می خواهیم با استفاده از MapReduce بدست آوریم که شبکه ABC، چند تماشاگر داشته است. برای اینکار لازم است که این دو نوع فایل را روی نام برنامه تلویزیونی، join کنیم و سپس تعداد تماشاگرهای برنامه‌های مختلف این channel را با هم جمع کنیم. برنامه mapper بصورت زیر نوشته می‌شود:

```
 GNU nano 2.9.3                                    join2_mapper.py

#!/usr/bin/env python
import sys

for line in sys.stdin:
    line = line.strip()
    key_value = line.split(",")
    key_in = key_value[0]
    value_in = key_value[1]
    testNum = [int(s) for s in value_in.split() if s.isdigit()]

    if len(testNum)>0:
        print( '%s\t%s' % (key_in, value_in))
    else:
        if value_in == 'ABC':
            print( '%s\t%s' % ( value_in, key_in))
```

برنامه reducer نیز بصورت زیر نوشته می‌شود :

```
  GNU nano 2.9.3                                          join2_reducer.py

#!/usr/bin/env python
import sys

ABC_dict={}
kvs=[]

for line in sys.stdin:
    line = line.strip()
    key_value = line.split('\t')
    kvs.append(key_value)

    if key_value[0]=="ABC":
        if key_value[1] not in ABC_dict:
            ABC_dict.update({key_value[1]:0})

for key_value in kvs:
    if key_value[0] in ABC_dict:
        ABC_dict[key_value[0]]+=int(key_value[1])

for key, value in ABC_dict.items() :
    print( '%s %s' % (key, value))
```

پس از تست برنامه های mapper و reducer روی یک نمونه فایل کوچک و اطمینان از صحت عملکرد آن، حال برای اجرا روی hdfs، ابتدا باید فایلهای داده تولید شده را به hdfs منتقل کنیم که این کار را با روشهایی که در تمرین hadoop آموختیم، انجام می دهیم. مراحل اینکار در شکل زیر نشان داده شده است :

```
negar@negar-VirtualBox:~/HW$ hdfs dfs -mkdir /homeworks/join-data
negar@negar-VirtualBox:~/HW$ hdfs dfs -copyFromLocal join2_gennum*.txt /homeworks/join-data
2020-05-30 19:55:13,165 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted =
 false, remoteHostTrusted = false
2020-05-30 19:55:13,830 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted =
 false, remoteHostTrusted = false
2020-05-30 19:55:13,896 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted =
 false, remoteHostTrusted = false
2020-05-30 19:55:13,955 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted =
 false, remoteHostTrusted = false
negar@negar-VirtualBox:~/HW$ hdfs dfs -copyFromLocal join2_gench*.txt /homeworks/join-data
2020-05-30 19:55:28,365 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted =
 false, remoteHostTrusted = false
2020-05-30 19:55:28,528 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted =
 false, remoteHostTrusted = false
2020-05-30 19:55:28,592 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted =
 false, remoteHostTrusted = false
2020-05-30 19:55:28,655 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted =
 false, remoteHostTrusted = false
negar@negar-VirtualBox:~/HW$ hdfs dfs -ls /homeworks/join-data
Found 8 items
-rw-r--r--   1 negar supergroup       1714 2020-05-30 19:55 /homeworks/join-data/join2_genchanA.txt
-rw-r--r--   1 negar supergroup       3430 2020-05-30 19:55 /homeworks/join-data/join2_genchanB.txt
-rw-r--r--   1 negar supergroup       5152 2020-05-30 19:55 /homeworks/join-data/join2_genchanC.txt
-rw-r--r--   1 negar supergroup        179 2020-05-30 19:55 /homeworks/join-data/join2_genchanT.txt
-rw-r--r--   1 negar supergroup      17114 2020-05-30 19:55 /homeworks/join-data/join2_gennumA.txt
-rw-r--r--   1 negar supergroup      34245 2020-05-30 19:55 /homeworks/join-data/join2_gennumB.txt
-rw-r--r--   1 negar supergroup      51400 2020-05-30 19:55 /homeworks/join-data/join2_gennumC.txt
-rw-r--r--   1 negar supergroup        177 2020-05-30 19:55 /homeworks/join-data/join2_gennumT.txt
negar@negar-VirtualBox:~/HW$
```

سپس بصورت زیر برنامه MapReduce را با معرفی برنامه mapper، برنامه reducer، فایلهای ورودی و محل ذخیره خروجی اجرا می کنیم :

```
negar@negar-VirtualBox:~$ hadoop jar hadoop-3.2.1/share/hadoop/tools/lib/hadoop-streaming-3.2.1.jar -mapper
'python3 HW/join2_mapper.py' -reducer 'python3 HW/join2_reducer.py' -input /homeworks/join-data -output /hom
eworks/join-output
2020-05-30 20:17:24,086 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2020-05-30 20:17:24,166 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2020-05-30 20:17:24,166 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2020-05-30 20:17:24,190 WARN impl.MetricsSystemImpl: JobTracker metrics system already initialized!
2020-05-30 20:17:24,547 INFO mapred.FileInputFormat: Total input files to process : 6
2020-05-30 20:17:24,568 INFO mapreduce.JobSubmitter: number of splits:6
```

پس از پایان کار، محتوای فایل خروجی تولید شده را مشاهده می‌کنیم، که در واقع، لیست برنامه های تلویزیونی ای که در شبکه ABC پخش شده‌اند را به همراه آمار تماشاگر آنها نشان می‌دهد.

```
negar@negar-VirtualBox:~$ hdfs dfs -ls /homeworks/join-output
Found 2 items
-rw-r--r--   1 negar supergroup          0 2020-05-30 20:17 /homeworks/join-output/_SUCCESS
-rw-r--r--   1 negar supergroup        390 2020-05-30 20:17 /homeworks/join-output/part-00000
negar@negar-VirtualBox:~$
negar@negar-VirtualBox:~$ hdfs dfs -cat /homeworks/join-output/part-00000
2020-05-30 20:19:23,025 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = fa
lse, remoteHostTrusted = false
Hourly_Show 48283
Almost_News 46592
Cold_Sports 52005
Loud_Games 49482
Surreal_Sports 46834
PostModern_News 50021
Dumb_Show 53824
Baked_News 47211
Hot_Games 50228
Hourly_Talking 108163
Almost_Games 49237
Cold_News 47924
Loud_Show 50820
Surreal_News 50420
PostModern_Games 50644
Dumb_Talking 103894
Baked_Games 51604
Hot_Show 54378
Hourly_Cooking 54208
Almost_Show 50202
negar@negar-VirtualBox:~$
```