IERG4300 / ESTR4300 Fall 2020 Homework 1

Release date: Sept 23, 2020

Due date: Oct 16, 2020 (Friday) 11:59pm

The solution will be posted right after the deadline, so no late homework will be accepted!

Every Student MUST include the following statement, together with his/her signature in the submitted homework.

I declare that the assignment submitted on Elearning system is original except for source material explicitly acknowledged, and that the same or related material has not been previously submitted for another course. I also acknowledge that I am aware of University policy and regulations on honesty in academic work, and of the disciplinary guidelines and procedures applicable to breaches of such policy and regulations, as contained in the website http://www.cuhk.edu.hk/policy/academichonesty/.

Signed (Student Tesze) Date: 16-10-2020

Name Chan Kai Yin SID 1155126983

Submission notice:

- Submit your homework via the elearning system

General homework policies:

A student may discuss the problems with others. However, the work a student turns in must be created COMPLETELY by oneself ALONE. A student may not share ANY written work or pictures, nor may one copy answers from any source other than one's own brain.

Each student **MUST LIST** on the homework paper the **name of every person he/she has discussed or worked with**. If the answer includes content from any other source, the student **MUST STATE THE SOURCE**. Failure to do so is cheating and will result in sanctions. Copying answers from someone else is cheating even if one lists their name(s) on the homework.

If there is information you need to solve a problem but the information is not stated in the problem, try to find the data somewhere. If you cannot find it, state what data you need, make a reasonable estimate of its value, and justify any assumptions you make. You will be graded not only on whether your answer is correct, but also on whether you have done an intelligent analysis.

Recommend the person with the maximal number of common followees

For all users:

```
1155124983@dic14: ~
5477 11997592        17031260        0.230303030303
5478 11997822        24959165        0.6
5479 11999782        5539522 0.371428571429
5480 12001262        16721942        0.75
5481 12001292        19741358        0.666666666667
5482 12002172        19674129        0.384615384615
5483 12002802        3382    0.3125
5484 12006882        290983238       1.0
5485 12007192        12725022        0.416666666667
5486 12011422        186228869       0.252252252252
5487 12012472        14148364        0.555555555556
5488 12012642        67173479        0.666666666667
5489 12019742        11415842        0.52
5490 12021302        13261222        0.5
5491 12023102        17948250        0.454545454545
5492 12025412        27049242        0.4
5493 12032112        29392870        0.4
5494 12032332        235178432       0.5
5495 12032502        11229292        0.409090909091
5496 12034532        739223  0.65625
5497 12038842        167429242       0.461538461538
5498 12039052        18104393        1.0
5499 12039382        19149620        1.0
5500 12040102        11211792        0.285714285714
5501 12044602        19394188        0.148
5502 12045722        13658142        1.0
5503 12046252        5748732 0.651162790698
5504 12047992        11718   0.479452054795
5505 12052522        17731751        1.0
5506 12052632        9946842 0.405405405405
5507 12053132        15357608        0.646341463415
5508 12053442        6482222 1.0
5509 12069912        11963132        0.263157894737
5510 12072102        30752163        0.269230769231
5511 12080652        15372333        0.631578947368
5512 12081222        32227347        1.0
5513 12081542        26376975        0.571428571429
5514 12083802        14667884        0.589743589744
5515 12084412        174535389       0.5
5516 12089102        2713951 0.508196721311
5517 12091452        8552602 0.529411764706
5518 12095412        18817203        0.461971830986
5519 12096202        16443393        0.289156626506
5520 12098472        18401667        0.5
5521 12101862        246401452       0.0714285714286
5522 12109622        47955797        0.461538461538
5523 12109952        14362294        0.56976744186
5524 12111812        9002252 0.6
5525 12116282        18070930        0.441860465116
5526 12118882        80260272        0.571428571429
5527 12120272        200969056       0.609756097561
                                                                                  22
                                                                                  1.3K
                                                                                  5502,1        6%
```

For the SID-version (11551*24983*):

```
        File Input Format Counters
                Bytes Read=47199621
        File Output Format Counters
                Bytes Written=34
20/10/13 14:37:28 INFO streaming.StreamJob: Output directory: md_1a_sid_fin/
1155124983@dic14:~$ hadoop fs -cat md_1a_sid_fin/part-* md_1a_sid_fin.txt
245424983       59507002        0.489795918367
cat: `md_1a_sid_fin.txt': No such file or directory
1155124983@dic14:~$ hadoop fs -cat md_1a_sid_fin/part-* > md_1a_sid_fin.txt
1155124983@dic14:~$
```

For the format shown, the first column Is the user. The second one is the person with the maximal number of common followees with the user. I also print the similarity in the third column as reference.

Job Overview

Job Name:	streamjob1937540332724634808.jar
User Name:	1155124983
Queue:	default
State:	SUCCEEDED
Uberized:	false
Submitted:	Mon Oct 12 16:39:16 HKT 2020
Started:	Mon Oct 12 16:39:23 HKT 2020
Finished:	Mon Oct 12 17:12:46 HKT 2020
Elapsed:	33mins, 23sec
Diagnostics:	
Average Map Time	10sec
Average Shuffle Time	9mins, 12sec
Average Merge Time	0sec
Average Reduce Time	5mins, 22sec

It takes about 33 minutes to run all user version.

Job Overview

Job Name:	streamjob6326505745168795817.jar
User Name:	1155124983
Queue:	default
State:	SUCCEEDED
Uberized:	false
Submitted:	Tue Oct 13 14:36:35 HKT 2020
Started:	Tue Oct 13 14:36:43 HKT 2020
Finished:	Tue Oct 13 14:37:26 HKT 2020
Elapsed:	42sec
Diagnostics:	
Average Map Time	8sec
Average Shuffle Time	4sec
Average Merge Time	0sec
Average Reduce Time	16sec

It takes about 5 minutes to run the SID-version.

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""sm_mapper.py"""
import sys
from collections import defaultdict

# input comes from STDIN (standard input)
a = defaultdict(set)
for line in sys.stdin:
    nums = line.strip().split()
    #print(nums[1],nums[0])
    print("%s\t%s"%(nums[1],nums[0]))

~
```

i Mapper

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""md_mapper.py"""
import sys
from collections import defaultdict

# input comes from STDIN (standard input)
b = defaultdict(set)
infile = "twitter_raw.txt"
with open(infile, 'rt') as f_i:
    for line in f_i.readlines():
        for i in line.strip().split()[1:]:
            b[str(line.strip().split(' ')[0])].add(i)
a = defaultdict(set)
for line in sys.stdin:
   # for i in line.split()[1:]:
    a[str(line.strip().split('\t')[0])].add(line.strip().split('\t')[1])

for v,k in a.items():
    max_v = 0
    max_k = 0
    for j,y in b.items():
        if len(k.union(y)) == 0 or v == j:
            pass
        else:
            new_v = float(len(k.intersection(y)))/len(k.union(y))
            if new_v > max_v:
                max_v = new_v
                max_k = j
    print("%s\t%s\t%s" % (v, max_k, max_v))
~
~
~
~
```

ii Reducer for all

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""md_mapper.py"""
import sys
from collections import defaultdict

# input comes from STDIN (standard input)
b = defaultdict(set)
infile = "twitter_raw.txt"
#set_a = {2, 4, 9, 8, 3}
#set_b = set()
with open(infile, 'rt') as f_i:
    for line in f_i.readlines():
        for i in line.strip().split()[1:]:
            b[str(line.strip().split(' ')[0])].add(i)
a = defaultdict(set)
for line in sys.stdin:
   # for i in line.split()[1:]:
    a[str(line.strip().split('\t')[0])].add(line.strip().split('\t')[1])

for v,k in a.items():
    #set_b = set()
    #for j in list(str(v)):
        #set_b.add(int(j))
    #print(set_b)
    if not(str(v).endswith('24983')):
        continue
    max_v = 0
    max_k = 0
    for j,y in b.items():
        if len(k.union(y)) == 0 or v == j:
            pass
        else:
            new_v = float(len(k.intersection(y)))/len(k.union(y))
            if new_v > max_v:
                max_v = new_v
                max_k = j
    print("%s\t%s\t%s" % (v, max_k, max_v))
    #print(set_a , set_b)
~
~
~
```

iii Reducer for SID version

Most similar people of EVERY user for the medium-sized dataset

For the all user version:

```
1155124983@dic14: ~
18451298      ['472390921', '258193217', '110924879'] [0.7142857142857143, 0.5714285714285714, 0.5714285714285714]
2459133       ['60740376', '22763550', '14817347']    [0.65, 0.5789473684210527, 0.5789473684210527]
18379911      ['140515926', '182216171', '34722951']  [0.3333333333333333, 0.3333333333333333, 0.3333333333333333]
43937306      ['17396212', '213804688', '101917758']  [0.5, 0.5, 0.375]
187744415     ['63787760', '35722372', '154309021']   [0.3643216080402010 3, 0.3448275862068966, 0.3360824742268041 5]
16302090      ['330502724', '17873693', '380440302']  [0.4285714285714285 5, 0.3333333333333333, 0.3333333333333333]
85259796      ['24229391', '26861141', '6854942']     [0.2727272727272727, 0.2272727272727272, 0.2222222222222222]
35727177      ['121777998', '25504218', '410430313']  [0.4285714285714285 5, 0.2941176470588235 4, 0.2857142857142857]
156406287     ['93058809', '47849517', '14186288']    [0.4838709677419355, 0.4444444444444444, 0.4166666666666667]
37677313      ['16107188', '19398363', '33140407']    [0.6, 0.5454545454545454, 0.4827586206896552]
87725109      ['98513781', '99571358', '20026604']    [0.6101694915254238, 0.5833333333333334, 0.5641025641025641]
21856863      ['81417061', '208685423', '15479902']   [0.3076923076923077, 0.3, 0.3]
362962333     ['159920794', '134594380', '23845031']  [0.5, 0.5, 0.5]
351017260     ['33015155', '15923563', '385222368']   [0.5, 0.5, 0.5]
43785477      ['18742761', '67384151', '66449444']    [1.0, 1.0, 1.0]
43228529      ['134532684', '5047361', '255810948']   [0.8461538461538461, 0.8076923076923077, 0.7857142857142857]
14801863      ['8525402', '14202246', '879521']       [0.4754098360655737 4, 0.3904761904761905, 0.2616822429906542]
81396783      ['32117870', '360025190', '54135604']   [0.3478260869565217 3, 0.2926829268292683, 0.2698412698412698]
15814534      ['108627997', '10368962', '41290988']   [0.3333333333333333, 0.3333333333333333, 0.3333333333333333]
538245743     ['495557857', '210707057', '260615063'] [0.3333333333333333, 0.3333333333333333, 0.2727272727272727]
135609133     ['112273454', '254613572', '71988016']  [0.1875, 0.1818181818181818 2, 0.1666666666666667]
15407699      ['16352307', '20187535', '43965834']    [0.5, 0.3333333333333333, 0.3]
547116689     ['81180486', '402210700', '543723375']  [0.8888888888888888, 0.8421052631578947, 0.8235294117647058]
31351370      ['36519250', '342770193', '25042112']   [0.2777777777777778, 0.25, 0.2333333333333334]
271579826     ['540262217', '84207073', '70850380']   [1.0, 1.0, 1.0]
92579250      ['22852321', '86852851', '16455322']    [0.3448275862068966, 0.32, 0.3043478260869565 4]
237901382     ['26391580', '37745903', '93699239']    [0.5, 0.5, 0.5]
29851286      ['358840898', '507651111', '164731069']  [0.5, 0.5, 0.5]
138520677     ['34576550', '138165921', '34174981']   [0.5714285714285714, 0.5, 0.5]
73436079      ['119159738', '22520465', '22553032']   [0.4054054054054054 3, 0.3617021276595745, 0.3488372093023256]
186295073     ['169977743', '236610292', '68082053']  [0.2380952380952380 8, 0.1666666666666666, 0.1428571428571428 5]
14112570      ['85534903', '119316062', '54986035']   [0.3333333333333333, 0.3076923076923077, 0.28]
107237609     ['251489444', '326253259', '537868113']  [0.7333333333333333, 0.7333333333333333, 0.5625]
22598623      ['213898324', '14709326', '15999463']   [0.6666666666666666, 0.5, 0.5]
234013833     ['16972622', '34854406', '150056555']   [0.3333333333333333, 0.3333333333333333, 0.2857142857142857]
113222804     ['83530184', '290223560', '398078754']  [0.5, 0.3333333333333333, 0.2857142857142857]
40901889      ['209122703', '44603491', '27938832']   [0.5, 0.5, 0.5]
92030627      ['47230848', '291353700', '47584677']   [0.4146341463414634, 0.3513513513513513 7, 0.34]
116640012     ['19465038', '45427910', '30943685']    [1.0, 1.0, 1.0]
21638018      ['55034153', '123044076', '117248641']  [0.3522727272727273, 0.35, 0.3283582089552239]
207231880     ['215155315', '252915055', '249858273']  [1.0, 0.5, 0.5]
13201312      ['16029780', '14750983', '18995389']    [0.4523809523809524, 0.4347826086956521 6, 0.4303797468354430 6]
36257808      ['41152764', '15088803', '48125843 6']  [1.0, 1.0, 1.0]
162065142     ['132766651', '301762017', '267152969']  [0.4137931034482758 6, 0.3846153846153846 4, 0.3181818181818182]
14336347      ['6151792', '14502476', '16849405']     [0.3, 0.25, 0.25]
30662150      ['35689389', '186284229', '271398203']  [0.4471547154471544, 0.3730158730158073, 0.3381294964028777]
80188000      ['85105609', '52726298', '211063202']   [0.3333333333333333, 0.3333333333333333, 0.3333333333333333]
18374880      ['16194935', '42381667', '32576594']    [0.6666666666666666, 0.5, 0.5]
224004614     ['120627505', '14868699', '29211085']   [0.6666666666666666, 0.5, 0.4]
18964696      ['7115872', '70720488', '25616964']     [0.53125, 0.4918032786885246, 0.4903846153846153 6]
19560673      ['24642657', '248007541', '49253437']   [1.0, 1.0, 1.0]
                                                                                                  1,1          Top
```

Job Overview

Job Name:	streamjob9265927301200960 16.jar
User Name:	1155124983
Queue:	default
State:	SUCCEEDED
Uberized:	false
Submitted:	Mon Oct 12 12:47:11 HKT 2020
Started:	Mon Oct 12 12:47:16 HKT 2020
Finished:	Mon Oct 12 13:22:53 HKT 2020
Elapsed:	35mins, 36sec
Diagnostics:	
Average Map Time	19sec
Average Shuffle Time	5mins, 24sec
Average Merge Time	0sec
Average Reduce Time	19mins, 30sec

It takes about 35 mins.

For the SID-version (1155*124983*) :

```
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=47932987
        File Output Format Counters
                Bytes Written=110
20/10/13 14:44:33 INFO streaming.StreamJob: Output directory: md_1b_sid_fin/
1155124983@dic14:~$ hadoop fs -cat md_1b_sid_fin/part-*
245424983       ['59507002', '37564255', '82963512']    [0.4897959183673469, 0.46099290780141844, 0.443609022
55639095]
1155124983@dic14:~$
```

–

Job Overview

Job Name:	streamjob1343057855736599733.jar
User Name:	1155124983
Queue:	default
State:	SUCCEEDED
Uberized:	false
Submitted:	Tue Oct 13 14:43:54 HKT 2020
Started:	Tue Oct 13 14:43:59 HKT 2020
Finished:	Tue Oct 13 14:44:31 HKT 2020
Elapsed:	32sec
Diagnostics:	
Average Map Time	7sec
Average Shuffle Time	5sec
Average Merge Time	0sec
Average Reduce Time	12sec

It takes about 32 secs.

For the format shown, the first column Is the user. The second one is the persons with the Top3 maximal number of common followees with the user. I also print the similaritys for the each followees in the third column as reference.

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""sm_mapper.py"""
import sys
from collections import defaultdict

# input comes from STDIN (standard input)
a = defaultdict(set)
for line in sys.stdin:
    nums = line.strip().split()
    #print(nums[1],nums[0])
    print("%s\t%s"%(nums[1],nums[0]))

~
```

iv mapper

dic14.ie.cuhk.edu.hk - PuTTY

```python
#!/usr/bin/env python
"""md_mapper.py"""
import sys
from collections import defaultdict
from collections import Counter

# input comes from STDIN (standard input)
b = defaultdict(set)
infile = "twitter_raw.txt"
with open(infile, 'rt') as f_i:
    for line in f_i.readlines():
        for i in line.strip().split()[1:]:
            b[str(line.strip().split(' ')[0])].add(i)
a = defaultdict(set)
#set_b = {"0":0,"1":0,"2":0}
set_b_k = []
for line in sys.stdin:
   # for i in line.split()[1:]:
    a[str(line.strip().split('\t')[0])].add(line.strip().split('\t')[1])

for v,k in a.items():
    max_v = 0
    max_k = 0
    set_b = {"0":0,"1":0,"2":0}
    for j,y in b.items():
        if len(k.union(y)) == 0 or v == j:
            pass
        else:
            new_v = float(len(k.intersection(y)))/len(k.union(y))
            if new_v > min(set_b.values()):
                set_b[str(j)] = new_v
                #c = Counter(set_b)
                #set_b = c.most_common(3)
                #set_b = sorted([(x, i) for (i, x) in enumerate(set_b)], reverse=True)[:3]
                #if new_v in set_b:
                    #set_b_k.append(v)
    c = Counter(set_b)
    mc = c.most_common(3)
    print("%s\t%s\t%s" % (v, [key for key, val in mc], [val for key, val in mc]))
~
~
~
~
~
```

v reducer for all

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""md_mapper.py"""
import sys
from collections import defaultdict
from collections import Counter

# input comes from STDIN (standard input)
b = defaultdict(set)
infile = "twitter_raw.txt"
#set_a = {2, 4, 9, 8, 3}
#set_b = set()
with open(infile, 'rt') as f_i:
    for line in f_i.readlines():
        for i in line.strip().split()[1:]:
            b[str(line.strip().split(' ')[0])].add(i)
a = defaultdict(set)
#set_b = {"0":0,"1":0,"2":0}
set_b_k = []
for line in sys.stdin:
   # for i in line.split()[1:]:
    a[str(line.strip().split('\t')[0])].add(line.strip().split('\t')[1])

for v,k in a.items():

   # set_c = set()
    #for j in list(str(v)):
        #set_c.add(int(j))
    #print(set_b)
    #if len(set_a.union(set_c)) != 5:
        #continue
    if not(str(v).endswith('24983')):
        continue
    max_v = 0
    max_k = 0
    set_b = {"0":0,"1":0,"2":0}
    for j,y in b.items():
        if len(k.union(y)) == 0 or v == j:
            pass
        else:
            new_v = float(len(k.intersection(y)))/len(k.union(y))
            if new_v > min(set_b.values()):
                set_b[str(j)] = new_v
                #c = Counter(set_b)
                #set_b = c.most_common(3)
                #set_b = sorted([(x, i) for (i, x) in enumerate(set_b)], reverse=True)[:3]
                #if new_v in set_b:
                    #set_b_k.append(v)
    c = Counter(set_b)
    mc = c.most_common(3)
    print("%s\t%s\t%s" % (v, [key for key, val in mc], [val for key, val in mc]))
~
"md_reducer_k_sid.py" 50L, 1453C
```

vi reducer for sid

Common followees shared between A and its similar users

All user version:

```
1155124983@dic14: ~
283196244:263440459     ['171112947', '14848513'] check_sun = 185961460
283196244:378050308     ['14848513', '171112947'] check_sun = 185961460
3725701:16134837        ['15859039', '9814222', '14629595', '20077017', '27948253', '17371254', '802093', '21107303', '14148308', '31045542', '19309099', '39505245', '76128019', '14986716',
'15050354', '28152330', '15923118', '5191441', '17137927', '44402429', '19126166', '56845340', '16650656', '15391752', '5485192', '15269276', '14174563', '18191483', '16153984', '14871640'
] check_sun = 640749356
3725701:31045542        ['9814222', '14629595', '20077017', '27948253', '17371254', '16134837', '802093', '21107303', '20682302', '14174563', '76128019', '14986716', '15050354', '28152330',
'89747994', '15923118', '5191441', '17137927', '44402429', '16650656', '15391752', '5485192', '15269276', '16153984', '56845340'] check_sun = 595257967
3725701:19309099        ['9814222', '14629595', '35806766', '20077017', '27948253', '17371254', '16134837', '802093', '21107303', '14148308', '14174563', '14986716', '15050354', '28152330',
'15923118', '5191441', '17137927', '44402429', '56845340', '16650656', '15391752', '5485192', '15269276', '18191483', '14871640'] check_sun = 475563865
10297:10880202  ['18785089', '61903300', '16729099', '46239813', '39764277', '14207863', '17625522', '16648790', '166686329', '16337410', '77057039', '15922433', '69468651', '14595574', '67
105749', '15947602', '14090017', '16203586', '14165865', '21123853', '59122922', '17817210', '17999296', '17366486', '104708526', '17586128', '17006579', '18006553', '16568227', '16729309',
'25318477', '194213355', '17539739', '21341221', '12609292', '23095870', '19994555', '612066610', '31504082', '35928501', '18992997', '27935056', '80681990', '6519522', '14779526', '19814678
', '14289945', '14380175', '16454856', '39299241', '15117711', '33705467', '13451802', '19209639', '44156753', '76702466', '10915612', '86338519', '23800611', '14677117', '11379362', '34015
638', '67828070', '29426700', '68792936', '14263368', '20804572', '2029971', '14467789', '21407083', '15433693', '37263075', '39511166', '18730477'] check_sun = 2298056382
10297:17539739  ['18785089', '61903300', '67105749', '46239813', '39764277', '17625522', '16648790', '166686329', '14467789', '69468651', '14595574', '12609292', '15947602', '14090017', '59
122922', '17817210', '17999296', '17586128', '39299241', '18006553', '76702466', '25318477', '21123853', '21341221', '23095870', '19994555', '84114313', '61206610', '21518427', '35928501',
'18992997', '27935056', '80681990', '6519522', '31504082', '14380175', '8877332', '10880202', '15117711', '33705467', '19209639', '86338519', '13451802', '14677117', '16454856', '11379362',
'34015638', '67828070', '36681590', '21407083', '20804572', '44156753', '2038061', '39511166', '18730477'] check_sun = 1853155226
10297:15947602  ['18785089', '61903300', '16729099', '46239813', '17625522', '16648790', '17539739', '14467789', '69468651', '14595574', '67105749', '76702466', '16203586', '59122922', '178
17210', '17999296', '17586128', '10880202', '18006553', '21123853', '21341221', '12609292', '23095870', '19994555', '84114313', '21095858', '21518427', '18992997', '27935056', '80681990', '
6519522', '16555481', '14380175', '16454856', '39299241', '15117711', '33705467', '19209639', '16834046', '13451802', '29426700', '34015638', '67828070', '36681590', '11379362', '17298881',
' '20804572', '21407083', '39511166', '166686329', '18730477'] check_sun = 1603228718
121625417:18703983      ['91067832'] check_sun = 91067832
121625417:261333504     ['91067832'] check_sun = 91067832
121625417:258844547     ['91067832'] check_sun = 91067832
15252736:15813531       ['107917068', '64962975', '42778165', '341218443', '141332097', '7861312', '15324310', '10450', '45406850', '633483', '16883084', '16069548', '15583454', '833571', '
15435220', '14630319', '16060735', '72952988', '14904191', '17978616', '14804982', '16338290', '100881649', '13538172', '2172', '107927712', '8914942', '15657647', '15914737', '8878882', '13
3315738', '13790282', '15151229', '15470236', '217911721', '1183041', '16174651', '7861622', '10241', '15300838', '16908436', '16670615', '26190261', '18229878', '14428964', '18474300', '17
75861', '45445539', '17611683', '15679119', '15290180', '243879870', '14134908', '18592004', '4893411', '144147761', '72555359', '40732410', '15329577', '50557241'] check_sun = 2292465040
15252736:15435220       ['15362015', '64962975', '272194422', '552961501', '14080429', '179166711', '166301108', '1775861', '16069548', '7861312', '17611683', '10450', '45406850', '15495464',
'32578648', '15470236', '309104501', '87330803', '15583454', '7453822', '16060735', '16883084', '14904191', '14388022', '16034444', '3272851', '148049821', '185762771', '163382901', '41385286',
'100881649', '59295766', '7861622', '16009002', '10792712', '50392516', '15914737', '8878882', '133315738', '22643137', '42778165', '243879870', '8070562', '16118476', '15938936', '1532957
7', '88626415', '97723803', '217911721', '19873252', '15657647', '16150457', '930061', '16714182', '16464203', '15813531', '127099086', '394460725', '8007092', '7081402', '16908436', '16670
615', '15652377', '17000223', '18229878', '14428964', '15324310', '31453974', '360999401', '45445539', '38056617', '47374533', '15679119', '18305153', '28521141', '26190261', '85822297', '1
5853073', '4893411', '72555359', '40732410', '168848970', '58101355', '50557241'] check_sun = 3871565094
15252736:15657647       ['18305153', '107917068', '309104501', '37219422', '153620015', '14080429', '17916671', '166301108', '1775861', '7861312', '17611683', '10450', '45406850', '16069548',
'15470236', '32578648', '87330803', '19873252', '833571', '15435220', '17298365', '16060735', '15290180', '64962975', '10241', '14388022', '16034444', '55296150', '14804982', '18576277', '
16338290', '41385286', '100881649', '16009002', '8914942', '1779201', '15914737', '17000223', '22643137', '42778165', '243879870', '16118476', '15938936', '88626415', '97723803', '217911721
', '16883084', '16150457', '930061', '16714182', '16464203', '7861622', '8007092', '7081402', '15300838', '16908436', '16670615', '15652377', '15813531', '47374533', '18229878', '272300662
', '28521141', '20099991', '45445539', '38056617', '362286999', '127099086', '11211712', '31453974', '14134908', '85822297', '15853073', '4893411', '14414761', '72555359', '40732410', '156140
50', '168848970', '50557241'] check_sun = 3502690486
18886152:26779967       ['21337501'] check_sun = 21337501
18886152:78325232       ['21337501'] check_sun = 21337501
18886152:22969221       ['21337501'] check_sun = 21337501
189246487:27978241      ['52758395', '7429892', '117809922'] check_sun = 177998209
189246487:386915587     ['52768208', '7429892', '117809922'] check_sun = 178008022
189246487:27409596      ['7429892', '117809922'] check_sun = 125239814
119165547:66786506      ['20747741', '21158690'] check_sun = 41906431
119165547:103314759     ['135397331', '17092592'] check_sun = 152489923
119165547:19160075      ['21158690', '17092592'] check_sun = 38251282
8586095:305073376       ['61086747', '28775296'] check_sun = 89862043
                                                                                                                                                                    28,1          0%
```

Job Overview	
Job Name:	streamjob4588403187632686413.jar
User Name:	1155124983
Queue:	default
State:	SUCCEEDED
Uberized:	false
Submitted:	Mon Oct 12 15:54:43 HKT 2020
Started:	Mon Oct 12 15:55:19 HKT 2020
Finished:	Mon Oct 12 16:25:44 HKT 2020
Elapsed:	30mins, 24sec
Diagnostics:	
Average Map Time	9sec
Average Shuffle Time	4mins, 15sec
Average Merge Time	0sec
Average Reduce Time	17mins, 57sec

It takes about 30mins to run.

SID-version (11551*24983*) :

```
1155124983@dic14: ~
                GC time elapsed (ms)=22484
                CPU time spent (ms)=789050
                Physical memory (bytes) snapshot=76550922240
                Virtual memory (bytes) snapshot=439540453376
                Total committed heap usage (bytes)=72009383936
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=46361139
        File Output Format Counters
                Bytes Written=2538
20/10/13 16:08:21 INFO streaming.StreamJob: Output directory: md_lc_output_f/
1155124983@dic14:~$ hadoop fs -cat md_lc_output_f/part-*
245424983:59507002      ['175114748', '185397767', '79011908', '16405372', '2944
7358', '157064188', '16045268', '22798591', '58607988', '19918047', '19253485',
'19164725', '25440543', '7458942', '116324294', '15941338', '234903540', '207331
730', '123565032', '15940915', '59010637', '21575175', '16679681', '282280747',
'134234059', '36750478', '241203719', '59238436', '82963512', '121537720', '1414
88457', '30913636', '13560922', '174248240', '204377704', '47968639', '38500966',
 '33235204', '271410124', '35415477', '15485844', '14642816', '35777372', '66489
082', '26299078', '227032210', '37564255', '133875756', '59166951', '15335662',
'115911638', '29725730', '241371677', '19780462', '19265186', '15989191', '62417
275', '16028750', '95486467', '15594313', '47348884', '277082507', '118530377',
'1403864851', '22060843', '259286437', '14416109', '59110488', '28676476', '16012
222', '567679531', '101569095'] check_sun = 5020276903
245424983:37564255      ['175114748', '207331730', '79011908', '21685295', '1342
34059', '157064188', '35777372', '15335662', '24045071', '19253485', '26546403',
 '19164725', '59110488', '336644583', '7458942', '116324294', '15941338', '23490
3540', '123565032', '15940915', '59010637', '21575175', '16679681', '282280747',
 '29447358', '36750478', '59238436', '82963512', '121537720', '141488457', '3091
3636', '174248240', '204377704', '47968639', '385009661', '271410124', '35415477',
 '15485844', '14642816', '22798591', '113425027', '95486467', '26299078', '133875
756', '119235720', '59166951', '16405372', '115911638', '29725730', '19780462',
'19265186', '15989191', '16028750', '66489082', '15594313', '47348884', '2770825
07', '25440543', '59507002', '22060843', '151186578', '14416109', '28676476', '1
6012222', '56767953'] check_sun = 4806343331
245424983:82963512      ['175114748', '185397767', '79011908', '16405372', '1342
34059', '157064188', '22798591', '15335662', '19918047', '19253485', '26546403',
 '19164725', '59110488', '7458942', '16679681', '15940915', '30913636', '1235650
32', '21575175', '21685295', '282280747', '29447358', '36750478', '241203719', '
121537720', '59166951', '141488457', '15594313', '47968639', '15485844', '385009
66', '33235204', '35415477', '234903540', '35777372', '66489082', '26299078', '1
51186578', '133875756', '19265186', '58607988', '115911638', '29725730', '197804
62', '59238436', '15989191', '62417275', '16028750', '95486467', '25440543', '15
941338', '118530377', '59507002', '271410124', '259286437', '14416109', '2168033
66', '466979336', '101569095'] check_sun = 4956116248
1155124983@dic14:~$
```

Job Overview

Job Name:	streamjob8091975199811643500.jar
User Name:	1155124983
Queue:	default
State:	SUCCEEDED
Uberized:	false
Submitted:	Tue Oct 13 15:53:52 HKT 2020
Started:	Tue Oct 13 15:53:59 HKT 2020
Finished:	Tue Oct 13 16:07:28 HKT 2020
Elapsed:	13mins, 28sec
Diagnostics:	
Average Map Time	31sec
Average Shuffle Time	2mins, 25sec
Average Merge Time	0sec
Average Reduce Time	19sec

It takes about 13 mins.

The A:{b} [C,D…] sum = x.

First column is User, second one is the top k user that are similar to A user. The third column in the common friends that shared. The sum is the number of shared friends.

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""sm_mapper.py"""
import sys
from collections import defaultdict

# input comes from STDIN (standard input)
a = defaultdict(set)
for line in sys.stdin:
    nums = line.strip().split()
    #print(nums[1],nums[0])
    print("%s\t%s"%(nums[1],nums[0]))

~
```

vii Mapper

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""md_mapper.py"""
import sys
from collections import defaultdict
from collections import Counter

# input comes from STDIN (standard input)
b = defaultdict(set)
infile = "twitter_raw.txt"
with open(infile, 'rt') as f_i:
    for line in f_i.readlines():
        for i in line.strip().split()[1:]:
            b[str(line.strip().split(' ')[0])].add(i)
a = defaultdict(set)
#set_b = {"0":0,"1":0,"2":0}
set_b_k = []
for line in sys.stdin:
    # for i in line.split()[1:]:
    a[str(line.strip().split('\t')[0])].add(line.strip().split('\t')[1])

for v,k in a.items():
    max_v = 0
    max_k = 0
    set_b = {"0":0,"1":0,"2":0}
    for j,y in b.items():
        if len(k.union(y)) == 0 or v == j:
            pass
        else:
            new_v = float(len(k.intersection(y)))/len(k.union(y))
            if new_v > min(set_b.values()):
                set_b[str(j)] = new_v
                #c = Counter(set_b)
                #set_b = c.most_common(3)
                #set_b = sorted([(k, i) for (i, k) in enumerate(set_b)], reverse=True)[:3]
                #if new_v in set_b:
                    #set_b_k.append(v)
    c = Counter(set_b)
    mc = c.most_common(3)
    #print("%s:%s\t%s" % (v, [key for key, val in mc], [val for key, val in mc]))
    for key,val in mc:
        set_c = b[str(key)].intersection(k)
        set_sum = sum(map(int,list(set_c)))
#       print("%s:%s\t%s\t check_sum=%d" % (v,key ,list(set_c), set_sum))
        print("{}:{}\t{} check_sun = {}".format(v,key,list(set_c),set_sum))
~
~
~
~
~
~
"md_reducer_c.py" 44L, 1447C                                          1,1          All
```

viii Reducer for all

dic14ie.cuhk.edu.hk - PuTTY

```
import sys
from collections import defaultdict
from collections import Counter

b = defaultdict(set)
infile = "twitter_raw.txt"
with open(infile, 'rt') as f_i:
    for line in f_i.readlines():
        for i in line.strip().split()[1:]:
            b[str(line.strip().split(' ')[0])].add(i)
a = defaultdict(set)
set_b_k = []
for line in sys.stdin:
    a[str(line.strip().split('\t')[0])].add(line.strip().split('\t')[1])

for v,k in a.items():
    if not(str(v).endswith('24993')):
        continue
    max_v = 0
    max_k = 0
    set_b = {"0":0,"1":0,"2":0}
    for j,y in b.items():
        if len(k.union(y)) == 0 or v == j:
            pass
        else:
            new_v = float(len(k.intersection(y)))/len(k.union(y))
            if new_v > min(set_b.values()):
                set_b[str(j)] = new_v
    c = Counter(set_b)
    mc = c.most_common(3)
    for key,val in mc:
        set_c = b[str(key)].intersection(k)
        set_sum = sum(map(int,list(set_c)))
        print("{}:{}\t{} check_sum = {}".format(v,key,list(set_c),set_sum))
```

ix reducer for sid

Time consumption for each MapReduce job and its tasks.

Mapper num	Reducer num	Max mapper time	Min mapper time	Avg mapper time	Max reducer time	Min reducer time	Avg reducer time	Total job
20	10	1mins, 24sec	5sec	13sec	1hrs, 13mins, 34sec	56mins, 53sec	63 mins, 19sec	1hrs, 13mins, 45sec
10	20	13sec	4sec	10sec	36mins, 28sec	24mins, 59sec	31mins, 13sec	36mins, 45sec
30	50	18sec	4sec	8sec	16mins, 0sec	9mins, 59sec	14mins, 01sec	16mins, 12sec
50	30	16sec	2sec	7sec	28mins, 27sec	17mins, 41sec	21mins, 36sec	31mins, 39sec
60	60	1mins, 1sec	3sec	6sec	14mins, 2sec	8mins, 10sec	11mins, 25sec	41mins, 8sec
60	100	15sec	3sec	5sec	13mins, 34sec	4mins, 40sec	9mins, 14sec	40mins, 14sec

From the first 4 cases, the observation is that when # of reduce task > # of M reduce task will perform better and faster than vice versa version. Like 30-50 case only take about 16min, while 50-30 case almost take about twice of the time. Because the mapper output is spread across files in reducer, it is better for have more reducer than mapper.

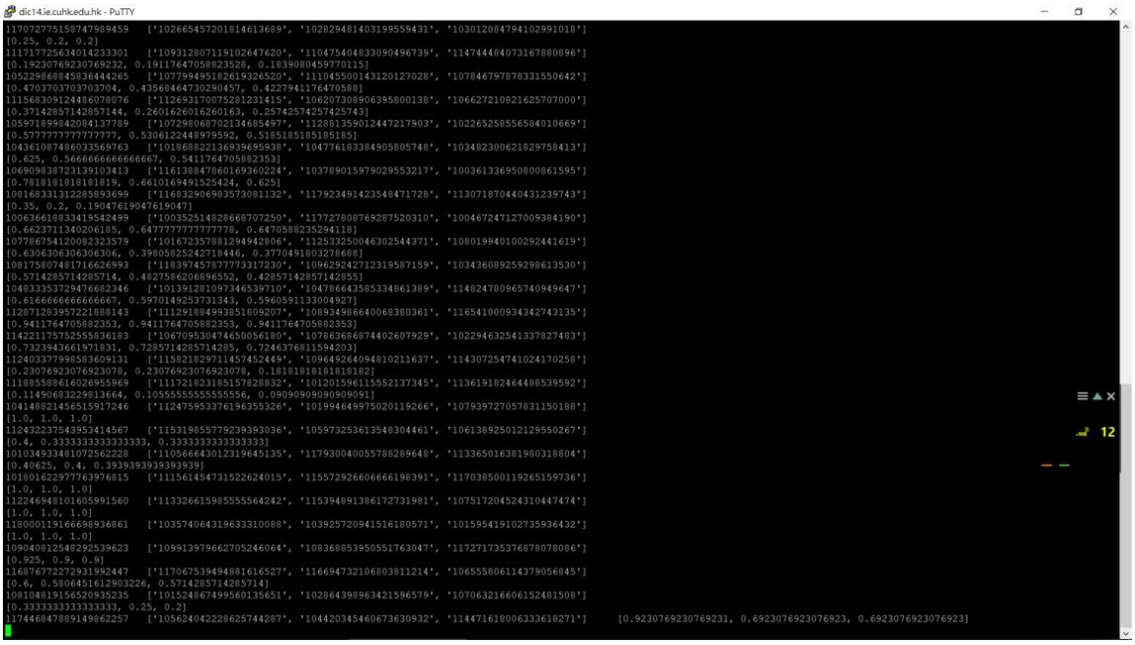
And from the 1-6 cases, observation is the more mappers and reducers will make the avg. time of map and reduce task shorter.

But in the total time needed, more reducer and mapper task doesn't mean

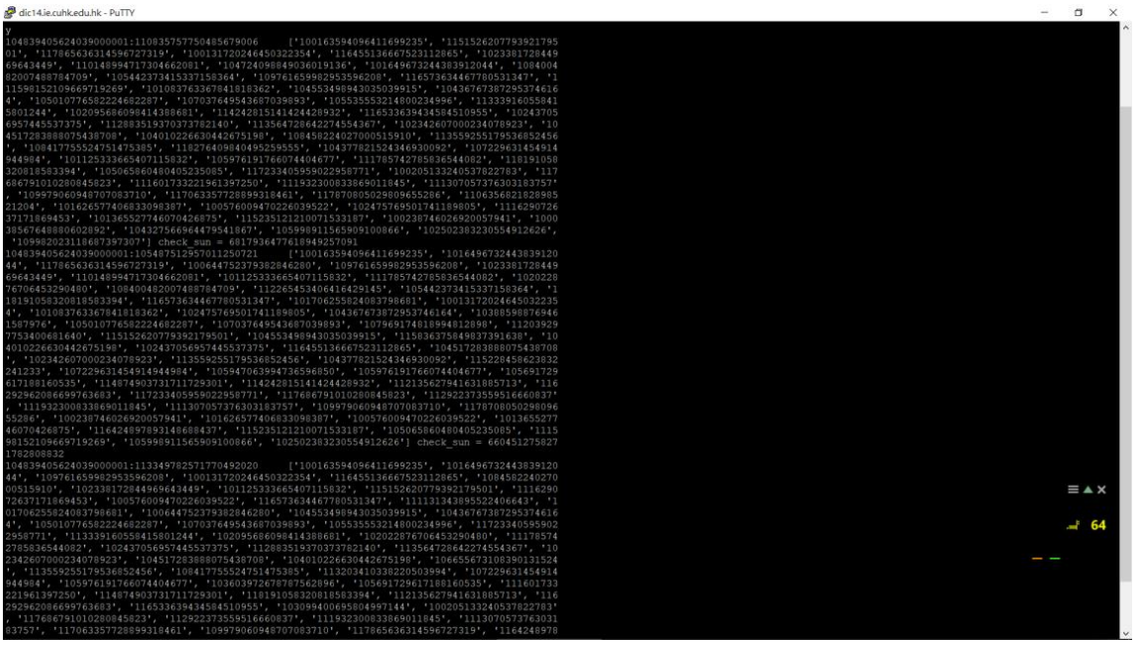
better performance and speed. Like the fastest case in here is 30-50 case which only take about 16 mins, while 60-100 case takes about 100 mins. But perhaps it due to the heavy-workload and congestion of the system.

Find the TOP 3 (=K) most similar people and the list of common followees for each user in the large dataset

For all users:



For SID-version:



Can’t find the one that match my Sid last digit, so I use “00001” as the last 5 digit to match. The format is the same as 1b.

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""sm_mapper.py"""
import sys
from collections import defaultdict

# input comes from STDIN (standard input)
a = defaultdict(set)
for line in sys.stdin:
    nums = line.strip().split()
    #print(nums[1],nums[0])
    print("%s\t%s"%(nums[1],nums[0]))

~
~
~
~
~
~
```

x mapper

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""md_mapper.py"""
import sys
from collections import defaultdict
from collections import Counter

# input comes from STDIN (standard input)
b = defaultdict(set)
infile = "gplus_raw.txt"
with open(infile, 'rt') as f_i:
    for line in f_i.readlines():
        for i in line.strip().split()[1:]:
            b[str(line.strip().split(' ')[0])].add(i)
a = defaultdict(set)
#set_b = {"0":0,"1":0,"2":0}
set_b_k = []
for line in sys.stdin:
    # for i in line.strip()[1:]:
    a[str(line.strip().split('\t')[0])].add(line.strip().split('\t')[1])

for v,k in a.items():
    max_v = 0
    max_k = 0
    set_b = {"0":0,"1":0,"2":0}
    for j,y in b.items():
        if len(k.union(y)) == 0 or v == j:
            pass
        else:
            new_v = float(len(k.intersection(y)))/len(k.union(y))
            if new_v > min(set_b.values()):
                set_b[str(j)] = new_v
                #c = Counter(set_b)
                #set_b = c.most_common(3)
                #set_b = sorted([(x, i) for (i, x) in enumerate(set_b)], reverse=True)[:3]
                #if new_v in set_b:
                    #set_b_k.append(v)
    c = Counter(set_b)
    mc = c.most_common(3)
    print("%s\t%s\t%s" % (v, [key for key, val in mc], [val for key, val in mc]))
~
~
~
~
~
~
~
~
~
~
~
"gp_reducer_k.py" 39L, 1203C                                                    1,1          All
```

58
601.0
2.9K

xi reducer for all

dic14.ie.cuhk.edu.hk - PuTTY

```
#!/usr/bin/env python
"""md_mapper.py"""
import sys
from collections import defaultdict
from collections import Counter

# input comes from STDIN (standard input)
b = defaultdict(set)
infile = "gplus_raw.txt"
set_a = [0, 0, 0, 0, 3]
set_b = set()
with open(infile, 'rt') as f_i:
    for line in f_i.readlines():
        for i in line.strip().split()[1:]:
            b[str(line.strip().split(' ')[0])].add(i)
a = defaultdict(set)
#set_b = {"0":0,"1":0,"2":0}
set_b_k = []
for line in sys.stdin:
    # for i in line.strip()[1:]:
    a[str(line.strip().split('\t')[0])].add(line.strip().split('\t')[1])

for v,k in a.items():

    if not(str(v).endswith('00001')):
        continue
    max_v = 0
    max_k = 0
    set_b = {"0":0,"1":0,"2":0}
    for j,y in b.items():
        if len(k.union(y)) == 0 or v == j:
            pass
        else:
            new_v = float(len(k.intersection(y)))/len(k.union(y))
            if new_v > min(set_b.values()):
                set_b[str(j)] = new_v
                #c = Counter(set_b)
                #set_b = c.most_common(3)
                #set_b = sorted([(x, i) for (i, x) in enumerate(set_b)], reverse=True)[:3]
                #if new_v in set_b:
                    #set_b_k.append(v)
    c = Counter(set_b)
    mc = c.most_common(3)

    for key,val in mc:
        set_c = b[str(key)].intersection(k)
        set_sum = sum(map(int,list(set_c)))
#       print("%s:%s\t%s\t check_sum=%d" % (v,key ,list(set_c), set_sum))
        print("{}:{}\t{} check_sun = {}".format(v,key,list(set_c),set_sum))
~
"gp_reducer_k_sid.py" 49L, 1487C                                                1,1          All
```

53
587.0
2.4K

xii reducer for sid