CSC 466 Lab 7 Report

Authors:
Sophia Chung, spchung@calpoly.edu

Implementation Overview

My implementation of PageRank requires one input: a .csv file that follows the generic data format of <Node1>, <Node1-Value>, <Node2>, <Node2-Value>. Hence, I use the usecols=range(4) option of pd.read_csv() to read in four columns of data regardless of whether the data contains an original fifth column. In parsing the unique node ids, I remove quotation marks and leading and trailing whitespace. Then, my process of building the initial graph consists of four steps. First, I get a list of all unique node ids. Second, I initialize a dataframe with these node ids as the index and starting PageRank values as the first column. Third, I populate a dictionary of inbound links. There is a key for each node, and a corresponding value of a list of nodes that link to the key node. Fourth, I create a dictionary of outbound link counts. There is a key for every node that has outbound links, and a corresponding value of how many outbound links that key node has.

This process was determined by first referring to the NCAA Football dataset, since it is the only dataset in the lab that represents a directed graph. Upon analyzing the data, I realized that nodes in the third column always linked to nodes in the first column. To apply this same logic in building all my graphs, I then used the default files for the Dolphins and Les Miserables datasets. I needed to keep repeated edges to count all links as both inbound and outbound.

My computation of PageRank is contained within a while loop that holds if the number of iterations is less than or equal to 100. In this loop, I first declare my stopping condition by comparing the PageRank values of the last two iterations. If the difference between the values is greater than epsilon, which I have set to 1e-6, then the while loop breaks. In calculating the PageRank values, I have the damping factor set to the typical value 0.85. I normalize values of each iteration by dividing by the sum of values prior to adding the values as a column to my dataframe.

I time how long it takes to read in the data from the input file and build the initial graph structure, as well as the average time of an iteration of computing PageRank. These times and the number of iterations it takes for convergence are printed after the ordered ranks, nodes, and PageRank scores.

Results

NCAA Football

There are 324 NCAA football teams in this dataset, so the output below is shortened to display only the top 100 teams after running my PageRank analysis:

```
1  Mississippi with pagerank: 0.037058862145893934
2  Florida with pagerank: 0.029433987263676385
3  Utah with pagerank: 0.01923606444591856
4  Wake Forest with pagerank: 0.01838056066146475
5  Oklahoma with pagerank: 0.018066973797495053
6  Texas Tech with pagerank: 0.01778523339388684
7  Alabama with pagerank: 0.016569208742077582
8  Virginia Tech with pagerank: 0.016084584483521694
9  Oregon State with pagerank: 0.01578394730787766
10  Texas with pagerank: 0.014743456964213626
11  Vanderbilt with pagerank: 0.01458546848003581
12  James Madison with pagerank: 0.01379738218609474
13  Boston College with pagerank: 0.01377636568151337
14  Georgia Tech with pagerank: 0.013286651851828672
15  Richmond with pagerank: 0.012408383874367702
16  South Carolina with pagerank: 0.012367752058134084
17  Virginia with pagerank: 0.012310011540734522
18  USC with pagerank: 0.012274154631792296
19  Montana with pagerank: 0.012262371910766555
20  North Carolina with pagerank: 0.011563377794093716
21  Florida State with pagerank: 0.011127103500949767
22  Duke with pagerank: 0.01111153786269008
23  Maryland with pagerank: 0.01067240918978553
24  Miami (FL) with pagerank: 0.010618279711005173
25  North Carolina State with pagerank: 0.010543394158478693
26  Clemson with pagerank: 0.009836263063457295
27  Georgia with pagerank: 0.008987511732426085
28  West Virginia with pagerank: 0.008642543126037907
29  Weber State with pagerank: 0.00861990905950392
30  East Carolina with pagerank: 0.008569511090386023
31  TCU with pagerank: 0.008491099348930584
32  Villanova with pagerank: 0.008400106947400982
33  Penn State with pagerank: 0.008072770239517918
34  Pittsburgh with pagerank: 0.008026750956414102
35  LSU with pagerank: 0.00786995273474218
36  Cincinnati with pagerank: 0.007779274602240125
37  Iowa with pagerank: 0.007634105989660923
38  Oregon with pagerank: 0.006614081912660935
39  California with pagerank: 0.006433572674199256
40  Rutgers with pagerank: 0.0060655613045539365
```

41 Tulsa with pagerank: 0.005792688980063491
42 Navy with pagerank: 0.005757669605691721
43 Northwestern with pagerank: 0.005570131235353967
44 Connecticut with pagerank: 0.005541431059199834
45 Boise State with pagerank: 0.005525547631937929
46 Appalachian State with pagerank: 0.005439349754118821
47 Brown with pagerank: 0.004986310064050345
48 Michigan State with pagerank: 0.004978952271997871
49 Missouri with pagerank: 0.004893702740559394
50 New Hampshire with pagerank: 0.004874635826479558
51 Northern Iowa with pagerank: 0.0048494398343069105
52 Stanford with pagerank: 0.004808617593341853
53 Houston with pagerank: 0.004743205027994918
54 Ohio State with pagerank: 0.004700770758320935
55 South Florida with pagerank: 0.004575648749621547
56 Northwestern State with pagerank: 0.004358762728708999
57 Jacksonville with pagerank: 0.004297200693157773
58 Arkansas with pagerank: 0.004211366983173544
59 Central Arkansas with pagerank: 0.0041595649375327495
60 Nebraska with pagerank: 0.004120858936862411
61 Tennessee with pagerank: 0.0038397372016959063
62 Kentucky with pagerank: 0.0038037084273708503
63 Mississippi State with pagerank: 0.003761705430408169
64 Albany with pagerank: 0.003750638074044571
65 Harvard with pagerank: 0.0037402178981057735
66 Grambling State with pagerank: 0.0037385038118449435
67 Oklahoma State with pagerank: 0.0036749081656666173
68 Southern Illinois with pagerank: 0.0036303330325621725
69 South Carolina State with pagerank: 0.0035456939911500253
70 Colorado with pagerank: 0.003522958744443216
71 Brigham Young with pagerank: 0.003514602517587852
72 Dayton with pagerank: 0.003511134217747453
3
73 Arizona with pagerank: 0.0034977940461352478
74 Ball State with pagerank: 0.00347209933165186
75 Massachusetts with pagerank: 0.0034472086568184085
76 Kansas with pagerank: 0.0034445995264185173
77 Holy Cross with pagerank: 0.0034415161280208096
78 Nevada with pagerank: 0.003413282221527165
79 Hawaii with pagerank: 0.0033993889778614673
80 Colgate with pagerank: 0.0033402722599071804
81 Eastern Washington with pagerank: 0.003296632101860067
82 Nicholls State with pagerank: 0.0032136538313672284
83 Liberty with pagerank: 0.0032060255874159964
84 Buffalo with pagerank: 0.0031838019961895466
85 Maine with pagerank: 0.0031812162187547515
86 Southern Miss with pagerank: 0.0031476450013873075
87 Rice with pagerank: 0.003145880720600468
88 Lafayette with pagerank: 0.003033757299900683

```
89  Yale with pagerank: 0.0030305394916929633
90  Wofford with pagerank: 0.002977530725045318
91  Texas State with pagerank: 0.0029257106342275413
92  McNeese State with pagerank: 0.00291342447725324
93  William & Mary with pagerank: 0.002900069762856018
94  Bowling Green with pagerank: 0.0028737031827475684
95  Notre Dame with pagerank: 0.002847608942664432
96  Middle Tennessee State with pagerank: 0.0028438992848191145
97  Illinois with pagerank: 0.0028151187787700373
98  San Diego with pagerank: 0.0027911780697780146
99  Wisconsin with pagerank: 0.0027909732440662234
100  Furman with pagerank: 0.002706887911672733
…
Read time: 0.06284785270690918 s
Average processing time: 0.005137296823354868 s
Total processing time: 0.13356971740722656 s
Number of iterations: 26
```

I believe that my implementation of PageRank did not discover the proper ranking of football teams in this dataset, given that I did not weigh by scores of games. Teams that won by greater amounts should receive higher PageRank scores, and teams that lost by greater amounts should receive lower PageRank scores. My program only takes into account whether a team won or lost.

Dolphins

Below is the full output from running my PageRank analysis on this dataset consisting of 62 dolphins:

```
1  Grin with pagerank: 0.03271007706019736
2  Jet with pagerank: 0.03175637996346916
3  Trigger with pagerank: 0.03144065660138645
4  Web with pagerank: 0.0301269492578724
5  SN4 with pagerank: 0.03010545695998054
6  Topless with pagerank: 0.029782945642991655
7  Scabs with pagerank: 0.028601822121858987
8  Patchback with pagerank: 0.02658669941549343
9  Gallatin with pagerank: 0.026178523041741418
10  Kringel with pagerank: 0.02480561338979213
11  Beescratch with pagerank: 0.02470747697803399
12  SN63 with pagerank: 0.024084107348374257
13  Feather with pagerank: 0.023476654830240282
14  SN9 with pagerank: 0.02212041473753323
15  Stripes with pagerank: 0.021813238049789928
16  Upbang with pagerank: 0.021679394685494454
17  SN100 with pagerank: 0.02077824796474151
18  DN21 with pagerank: 0.020069983199813877
```

19 Haecksel with pagerank: 0.020064610027519938
20 Jonah with pagerank: 0.019507618351089214
21 TR99 with pagerank: 0.01936158302540634
22 SN96 with pagerank: 0.0176854616192506
23 TR77 with pagerank: 0.01741138347432495
24 Number1 with pagerank: 0.017152033153017446
25 Beak with pagerank: 0.017072692117670606
26 MN105 with pagerank: 0.01703014055217033
27 MN83 with pagerank: 0.01702603471025159
28 Hook with pagerank: 0.016746249790970454
29 SN90 with pagerank: 0.016155091344836902
30 Shmuddel with pagerank: 0.016016275996396902
31 DN63 with pagerank: 0.01567957476064571
32 PL with pagerank: 0.015347651313437658
33 Zap with pagerank: 0.01526475506965493
34 Fish with pagerank: 0.015163862809704904
35 Oscar with pagerank: 0.014931873769855337
36 Double with pagerank: 0.014523083884374217
37 DN16 with pagerank: 0.014438049865512928
38 Bumper with pagerank: 0.013379168508043502
39 Ripplefluke with pagerank: 0.013315303517472113
40 Knit with pagerank: 0.012951863809620441
41 Thumper with pagerank: 0.012886690509427375
42 TSN103 with pagerank: 0.01215632223043911
43 Mus with pagerank: 0.011514591500564806
44 Notch with pagerank: 0.011223089748968004
45 Zipfel with pagerank: 0.011076277328159056
46 MN60 with pagerank: 0.009916178482627873
47 TR88 with pagerank: 0.008904242417948273
48 TR120 with pagerank: 0.008852299015703825
49 Wave with pagerank: 0.0083309651373999
50 TSN83 with pagerank: 0.00820627766249678
51 SN89 with pagerank: 0.007787861074833102
52 Vau with pagerank: 0.007528125614555809
53 CCL with pagerank: 0.007331232642144516
54 Zig with pagerank: 0.006192185273502460 5
55 Quasi with pagerank: 0.005418691137713997
56 MN23 with pagerank: 0.005418691137713997
57 TR82 with pagerank: 0.005264797927620771
58 Cross with pagerank: 0.005091757399135877
59 Five with pagerank: 0.005091757399135877
60 Whitetip with pagerank: 0.00497823937498623
61 SMN5 with pagerank: 0.004930265346089241
62 Fork with pagerank: 0.00485045891880098

Read time: 0.011930704116821289 s
Average processing time: 0.0011675059795379639 s
Total processing time: 0.028020143508911133 s

```
Number of iterations: 24
```

I believe that my implementation of PageRank did discover the proper ranking of dolphins in this dataset, given that the dolphin named “Grin” appears most frequently. “Grin” appears 24 times, whereas the second-ranked dolphin named “Jet” appears 18 times. The individual associations between dolphins are not weighed in any way.

Les Miserables

Below is the full output from running my PageRank analysis on this dataset consisting of 77 Les Miserables characters:

```
1  Valjean with pagerank: 0.07543042530837762
2  Myriel with pagerank: 0.04278077266466168
3  Gavroche with pagerank: 0.035767069159211996
4  Marius with pagerank: 0.030894763128564134
5  Javert with pagerank: 0.030302727967609234
6  Thenardier with pagerank: 0.02792646332540599
7  Fantine with pagerank: 0.027022696007954972
8  Enjolras with pagerank: 0.02188185108708204
9  Cosette with pagerank: 0.020611199042427062
10  MmeThenardier with pagerank: 0.0195011125992384
11  Bossuet with pagerank: 0.018959353593539587
12  Courfeyrac with pagerank: 0.01857824785674721
13  Eponine with pagerank: 0.017793835282701956
14  Mabeuf with pagerank: 0.017477861589051402
15  Bahorel with pagerank: 0.017199691515163314
16  Joly with pagerank: 0.01719969151516331
17  Babet with pagerank: 0.01669180470403346
18  Gueulemer with pagerank: 0.01669180470403346
19  Claquesous with pagerank: 0.01656098608100179
20  MlleGillenormand with pagerank: 0.016260193158400056
21  Combeferre with pagerank: 0.01589195510014614
22  Feuilly with pagerank: 0.015891955100146137
23  Tholomyes with pagerank: 0.01564738647879375
24  Bamatabois with pagerank: 0.015576350350707932
25  Montparnasse with pagerank: 0.015170897835358244
26  Gillenormand with pagerank: 0.014957458673204251
27  Grantaire with pagerank: 0.014456509578157223
28  Prouvaire with pagerank: 0.013145737805580752
29  Fameuil with pagerank: 0.012618173360193967
30  Favourite with pagerank: 0.012618173360193967
31  Blacheville with pagerank: 0.012618173360193967
32  Zephine with pagerank: 0.012618173360193967
33  Dahlia with pagerank: 0.012618173360193967
34  Listolier with pagerank: 0.012618173360193967
35  Judge with pagerank: 0.012424742464715026
```

36 Champmathieu with pagerank: 0.012424742464715026
37 Cochepaille with pagerank: 0.012424742464715024
38 Chenildieu with pagerank: 0.012424742464715024
39 Brevet with pagerank: 0.012424742464715024
40 Brujon with pagerank: 0.011866620656394619
41 Fauchelevent with pagerank: 0.011638078620961664
42 MmeHucheloup with pagerank: 0.010689715931435684
43 MmeMagloire with pagerank: 0.010277277886941294
44 MlleBaptistine with pagerank: 0.010277277886941294
45 Simplice with pagerank: 0.009073659379240926
46 LtGillenormand with pagerank: 0.00871357961228561
47 MmeBurgon with pagerank: 0.007805550092551092
48 Pontmercy with pagerank: 0.007368077330779468
49 Woman2 with pagerank: 0.006836872713446389
50 Toussaint with pagerank: 0.006836872713446389
51 Anzelma with pagerank: 0.006313524107898714
52 MotherInnocent with pagerank: 0.006202145594049218
53 MmePontmercy with pagerank: 0.0060101231169387695
54 Child1 with pagerank: 0.005791226313966166
55 Child2 with pagerank: 0.005791226313966166
56 Cravatte with pagerank: 0.005584337717491791
57 Champtercier with pagerank: 0.005584337717491791
58 Count with pagerank: 0.005584337717491791
59 Geborand with pagerank: 0.005584337717491791
60 OldMan with pagerank: 0.005584337717491791
61 Napoleon with pagerank: 0.005584337717491791
62 CountessDeLo with pagerank: 0.005584337717491791
63 Perpetue with pagerank: 0.005407490144423935
64 Magnon with pagerank: 0.005271217001844614
65 Jondrette with pagerank: 0.0052654070278422465
66 Marguerite with pagerank: 0.005260338681025137
67 Woman1 with pagerank: 0.005244189450255616
68 BaronessT with pagerank: 0.005146445420221375
69 Gribier with pagerank: 0.0044211447246837775
70 MlleVaubois with pagerank: 0.00392250287692541
71 MmeDeR with pagerank: 0.0037290528174173895
72 Gervais with pagerank: 0.0037290528174173895
73 Isabeau with pagerank: 0.0037290528174173895
74 Scaufflaire with pagerank: 0.0037290528174173895
75 Labarre with pagerank: 0.0037290528174173895
76 Boulatruelle with pagerank: 0.003431644189611106
77 MotherPlutarch with pagerank: 0.0032986104051919807

Read time: 0.018924951553344727 s
Average processing time: 0.0017047894967568886 s
Total processing time: 0.06307721138000488 s
Number of iterations: 37

I believe that my implementation of PageRank did not discover the proper ranking of characters in this dataset, given that I did not weigh by occurrences. My program fails to take into account the number of chapters that characters co-occur. Characters that occur in more chapters should receive higher PageRank scores, and characters that occur in fewer chapters should receive lower PageRank scores.

Overall Summary

My code ran very fast. Although these three datasets were small, I think my code would still run quickly and efficiently given larger datasets. It produced good rankings for all datasets, but I think it best ranked the Dolphins dataset, which is undirected and unweighted. Individual associations between dolphins were mutual and equally important. I think the rankings for the NCAA Football dataset were the lowest quality, since that data is both directed and weighted. Knowing that there was a winning and losing team, I was able to represent a directed graph. However, I did not make use of the given scores and their differences. The Les Miserables dataset is a combination of the two other datasets, as it is undirected and weighted. Individual associations were mutual, like in the Dolphins dataset, but they are not equally "important," since the number of chapters in which two characters co-occur differs. Better rankings would be produced if my PageRank accounted for weights.

Performance Evaluation

Table 1. Performance Statistics

Dataset	Number of Nodes	Average Processing Time (s)	Number of Iterations
NCAA Football	324	0.005137	26
Dolphins	62	0.001168	24
Les Miserables	77	0.001705	37

Table 1 gives us the average processing time for each PageRank iteration depending on the dataset. The average processing time does not greatly differ by dataset, as the time for the NCAA Football dataset is roughly five times the others since the data has five times the amount of nodes. The number of iterations in calculating PageRank for the Les Miserables dataset is larger than the others presumably because there are many characters with similar importance; it was easier to distinguish importance between football teams, as is also the case with dolphins.

Appendix

README

If you have any questions, please contact:
Sophia Chung spchung@calpoly.edu
.zip archive contains pageRank.py:

- takes as input a .csv file that follows the generic data format assumed by the lab