Lucas Summers
lsumme01@calpoly.edu

CSC 466 Lab 7 Report - PageRank

Implementation Overview

For this lab, I implemented the PageRank algorithm following the approach discussed in class.
The main components of my implementation include:

- An adjacency list representation for the graph built using python dictionaries, which allows for efficient storage and traversal
- Automatically detects the dataset type (football, dolphins, etc.) and applies appropriate edge direction rules (ex. in the NCAA football dataset, edges run from losing teams to winning teams) as well as detecting undirected vs directed graphs.
- I used a damping factor of 0.85, which is the standard value used in the original PageRank algorithm.
- The PageRank calculation uses NumPy's efficient vectorized matrix operations
- Dangling nodes (nodes with no outgoing links) are handled by distributing their PageRank scores evenly across all nodes, as described in the PageRank algorithm.
- The algorithm iterates until either the difference between consecutive PageRank vectors falls below a convergence threshold (epsilon = 1e-6) or until a maximum number of iterations (200) is reached.

Results

NCAA-FOOTBALL

Settings: No special settings were used.

Output:

```
Processing file: csv/NCAA_football.csv
Read time: 0.0019 seconds
Number of nodes: 324
Number of edges: 1537
Processing time: 0.0049 seconds
Number of iterations: 31

PageRank Results:
1 Mississippi with pagerank: 0.03468440968146036
```

```
2 Florida with pagerank: 0.02762415375041421
3 Utah with pagerank: 0.018414337577249845
4 Oklahoma with pagerank: 0.01761791088556797
5 Texas Tech with pagerank: 0.017354939767863908
6 Wake Forest with pagerank: 0.016916950760457485
7 Alabama with pagerank: 0.015563481747835705
8 Oregon State with pagerank: 0.015118715234169
9 Virginia Tech with pagerank: 0.014957676381056715
10 Texas with pagerank: 0.014438732341241985
11 Vanderbilt with pagerank: 0.01361621656051863
12 Boston College with pagerank: 0.012730219509029052
13 Georgia Tech with pagerank: 0.012342673985648157
14 Richmond with pagerank: 0.01226188507546662
15 USC with pagerank: 0.011860773986038377
16 South Carolina with pagerank: 0.011643144472374313
17 Virginia with pagerank: 0.011586167999977618
18 James Madison with pagerank: 0.011514137293860749
19 Montana with pagerank: 0.011341290077729547
20 North Carolina with pagerank: 0.010751889913855585
```

Observations:

The PageRank algorithm effectively identified the most prestigious college football teams from the 2009 season. Mississippi, Florida, Utah, and Oklahoma rank at the top, which aligns with their strong performance that season. Teams with high PageRank scores typically defeated many other teams that also had good records, creating a network of quality wins. This demonstrates how PageRank captures not just the number of wins, but also the quality of those wins (ie. beating good teams contributes more to a team's ranking than beating teams with poor records). Overall, the algorithm successfully models the intuitive notion that beating strong opponents should count more toward a team's prestige than beating weaker ones.

DOLPHINS

Settings: No special settings were used.

Output:

```
Processing file: csv/dolphins.csv
Read time: 0.0006 seconds
Number of nodes: 62
Number of edges: 636
Processing time: 0.0003 seconds
```

```
Number of iterations: 24

PageRank Results:
1 Jet with pagerank: 0.03142358004820106
2 Trigger with pagerank: 0.03141556607951017
3 Grin with pagerank: 0.02948094026437463
4 Web with pagerank: 0.029122339276147456
5 SN4 with pagerank: 0.027529945875128515
6 Topless with pagerank: 0.027209265003857744
7 Scabs with pagerank: 0.027077319948614658
8 Patchback with pagerank: 0.025748696878114846
9 Gallatin with pagerank: 0.024739435687592563
10 SN63 with pagerank: 0.023569798117767812
11 Beescratch with pagerank: 0.023436736979700154
12 Kringel with pagerank: 0.02306905252787728
13 Feather with pagerank: 0.02249020829642701
14 Stripes with pagerank: 0.02148189831127218
15 SN9 with pagerank: 0.020590548284768716
16 Upbang with pagerank: 0.020458273676014556
17 SN100 with pagerank: 0.020000446207397597
18 DN21 with pagerank: 0.019444970509950626
19 Haecksel with pagerank: 0.01925472697090657
20 Jonah with pagerank: 0.018407798603147448
```

Observations:

The dolphin social network analysis reveals that Jet and Trigger are the most central dolphins in this community, with nearly identical PageRank scores. This suggests they play similarly important roles in the social structure. The gradual decrease in PageRank scores shows a relatively balanced social network without extreme centralization around a single individual. The top-ranked dolphins likely represent those with the most diverse social connections within the group. It's interesting to note how closely the scores of the top two dolphins match, indicating they may share similar patterns of interaction within the network. Overall, the PageRank algorithm effectively identifies the key dolphins in this natural social network, demonstrating its applicability beyond web pages.

LES-MISERABLES

Settings: No special settings were used.

Output:

```
Processing file: csv/lesmis.csv
Read time: 0.0019 seconds
Number of nodes: 77
Number of edges: 1016
Processing time: 0.0004 seconds
Number of iterations: 29

PageRank Results:
1 Valjean with pagerank: 0.07250038047192493
2 Myriel with pagerank: 0.044960331851734286
3 Gavroche with pagerank: 0.03276565672605693
4 Marius with pagerank: 0.028153313887683107
5 Javert with pagerank: 0.02800781193758472
6 Thenardier with pagerank: 0.026130963346907378
7 Fantine with pagerank: 0.025466485748986703
8 Cosette with pagerank: 0.019654412675424813
9 Enjolras with pagerank: 0.01938299469945728
10 MmeThenardier with pagerank: 0.018397784118505796
11 MlleGillenormand with pagerank: 0.017149129213784555
12 Bossuet with pagerank: 0.01702678554438354
13 Mabeuf with pagerank: 0.01675467107328685
14 Courfeyrac with pagerank: 0.016665033737770802
15 Eponine with pagerank: 0.016463091262474323
16 Bahorel with pagerank: 0.015548623520987058
17 Joly with pagerank: 0.015548623520987058
18 Gillenormand with pagerank: 0.015375859907784466
19 Babet with pagerank: 0.015363100523660484
20 Gueulemer with pagerank: 0.015363100523660484
```

Observations:

The PageRank results for the Les Misérables character co-occurrence network align remarkably well with the narrative importance of characters in the novel. Jean Valjean's dominant position with a PageRank of 0.073 (over 60% higher than the second-ranked character) accurately reflects his role as the protagonist and central character connecting various storylines. The significant gap between Valjean and other characters demonstrates how effectively PageRank identifies the most central node in a complex network. Major characters like Myriel, Gavroche, Marius, and Javert occupy high ranks, which matches their importance in the narrative. Overall, the PageRank algorithm successfully extracts the hierarchical importance of characters from the novel's structure without any knowledge of the actual story.

Overall Summary

The PageRank algorithm demonstrated effective performance across all three datasets, successfully identifying the most central or important nodes in each network. In the NCAA football dataset, the algorithm identified successful teams that not only won many games but also defeated other successful teams. For the Dolphins dataset, PageRank revealed the most socially connected individuals in the community. In the Les Mis dataset, the algorithm accurately ranked characters according to their narrative importance.

Overall, The algorithm works best with well-connected networks that have a natural hierarchy of importance. The Les Mis graph showed this the clearest, with there being a large difference between scores of main character and supporting roles. The football dataset also demonstrates PageRank's ability to capture indirect relationships, where teams are ranked not just by their direct wins, but also by the quality of the teams they defeated. One limitation observed is that in networks with tightly-knit communities (like the dolphins), the differences between ranks can be small, making fine distinctions between similarly-ranked nodes less reliable. This suggests PageRank works best for identifying the top tier of important nodes, rather than creating precise rankings among nodes of similar importance.

Performance Evaluation

Dataset	Nodes	Edges	Read Time (s)	Processing Time (s)	Iterations
NCAA	324	1537	0.0019	0.0049	31
Dolphins	62	636	0.0006	0.0003	24
LesMis	77	1016	0.0019	0.0004	29

The use of NumPy vectorized operations contributed significantly to fast runtimes across all the datasets. The number of iterations remained relatively consistent, with the largest dataset only having slightly more iterations. The processing time also appears to grow linearly with the size of the graph, suggesting good scalability for even larger graphs. However, for significantly larger networks, such as web networks with billions of nodes, the implementation would need to be optimized further:

- The dense matrix representation would need to be replaced with sparse matrix operations to handle the larger memory requirements.
- Parallel computing approaches might be necessary for usable runtimes
- Approximation techniques could be employed to reduce computational complexity while maintaining reasonable accuracy

Appendix (README)

# 466 Lab 7
Lucas Summers (lsumme01@calpoly.edu)

## Requirements
```
numpy
```
90
If not already installed, run `pip install -r requirements.txt`.

## Running
python pageRank.py <input_file>

Where <input_file> is the path to a CSV file formatted according to the specifications in the assignment.
NOTE: "football" must be in the filename to correctly read in the NCAA football graph

The program will output:
- Read time (time to parse the input file)
- Number of nodes and edges in the graph
- Processing time (time to compute PageRank)
- Number of iterations until converg90ence
- The ranked list of nodes with their PageRank scores