Lecture 9 - PageRank

Thursday, February 10, 2022 8:00 AM

Homework 1 stuff:

- template available
  - → simple wiki p1
    - p2p p2
    - amazon p3

p1: Nx bfs_pred./bfs_succ.

don't do what you think

→ code up your own

→ overall, easier than you think

p2: $\alpha = 1 + \frac{1}{n}\left[\sum_{i \in V(G)} \ln\left(\frac{x_i}{x_{min}}\right)\right]$

- n: num verts
- x_i: degree of vert i — sum in + out degree
- x_{min}: min degree

p3: results "not great"

Review centrality

centrality → measure of "importance" for a vertex

degree centrality: degree of a vertex
→ strength of connection within a single hop

closeness centrality: average shortest paths length from a vertex
→ small distance from vertex to rest of the network

betweenness centrality: ratio of shortest paths going through some vertex
→ important for information flow

eigenvector centrality: how close a vertex is to other important vertices

---

Back in the day

- lot of cool stuff on that internet

One problem: I can't find it

One problem: I can't find it

Search engines: Altavista, Yahoo, etc.

↳ used basic keyword matching

Issue: spam or other untrustworthy actors can abuse the system

↳ How would we define trustworthiness?

→ As a company, it's tough

→ However, we can consider "wisdom of the crowd"

↳ we can consider the "crowd" as web pages

→ they'll generally link to trustworthy sites

Obviously → spam still an issue

Obviously → spam still an issue

Google:
- Let's define trustworthiness as above using PageRank
→ include both keyword + trust

PageRank

A spectral centrality measure
→ eigenvectors/values

Lots of applications
→ search, recommender systems

You're important if you're getting linked to by other important vertices

Let's consider 3 computational models

---

1. Random surfer model

→ a person randomly walking through the

walking through the internet
when they hit a 'sink' they randomly jump to any other site

PageRank of v: is probability at some time t that the surfer is at v

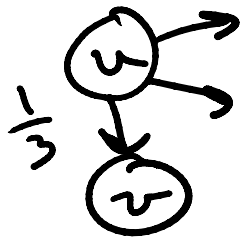
---

2. Vertex-centric model

All $p(v)$ initialized to $\frac{1}{|V(G)|}$

Update $p(v)$

$$p(v) = \sum_{u \in N^-(v)} \frac{p(u)}{d^+(u)}$$



For sinks: we can sum up and distribute PageRanks or we can construct edges from sinks to all other vertices

---

3. Linear Algebraic model

3. Linear Algebraic model

We have adjacency matrix A

Diagonal degree matrix D

→ row sums of A along a diagonal

aka out degree

We can get transition probability matrix $M' = D^{-1}A$

but we'll take the transpose

$$M = M'^T = (D^{-1}A)^T$$

Similar algorithm as before:

$P(v) = \frac{1}{|V(G)|}$ for all $v \in V(G)$

Iterate: $P_{i+1} = MP_i$

until $\| P_{i+1} - P_i \| < \epsilon$

Note: $Ax = \lambda x$

x ↑ eigenvector, λ ← eigenvalue

$M p_\infty = p_\infty$ ← power iteration

$M p_\infty = p_\infty$

aka we're solving for the eigenvector of the largest nonzero eigenvalue → 1

Personalized PageRank

To extend surfer model to "personalize" pagerank for v:

- Consider starting a walk from v
- *Jump back to v with some probability
- Jump back to v from sinks

→ this gives our probability distribution of our random surfer starting from v

Use cases:

Song recommendations: bipartite listening network

Search for some reasons

PageRank for prediction

Could also be used in a competition network to measure "quality"

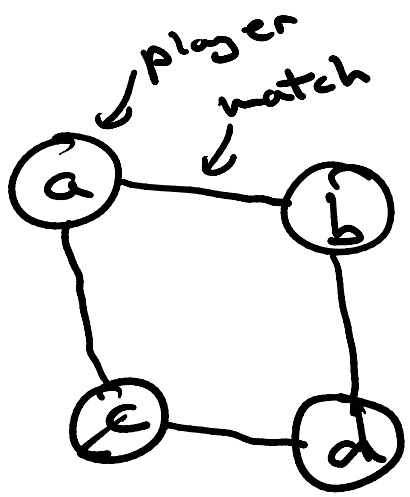
↳ Consider players a, b, c, d

a plays b

b plays d

c plays d

c plays a



We can "orient" our competition network to point to winners

We can compute pagerank to get overall "quality" of each competitor within the competition network