PageRank

Ben Burns, Dan Magazu, Lucas Chagas,
Thomas Webster, Trung Do

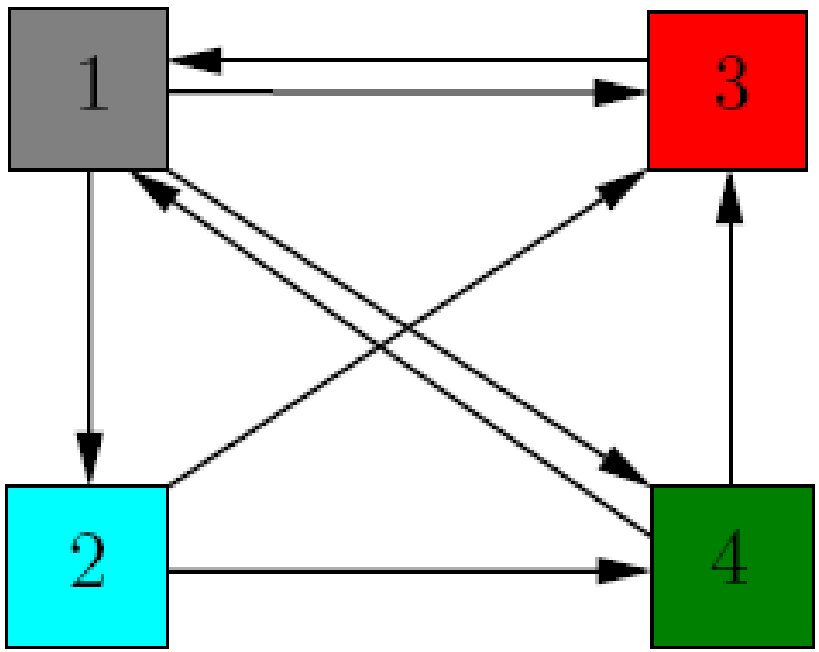
Fall 2021

Table of Contents

PageRank

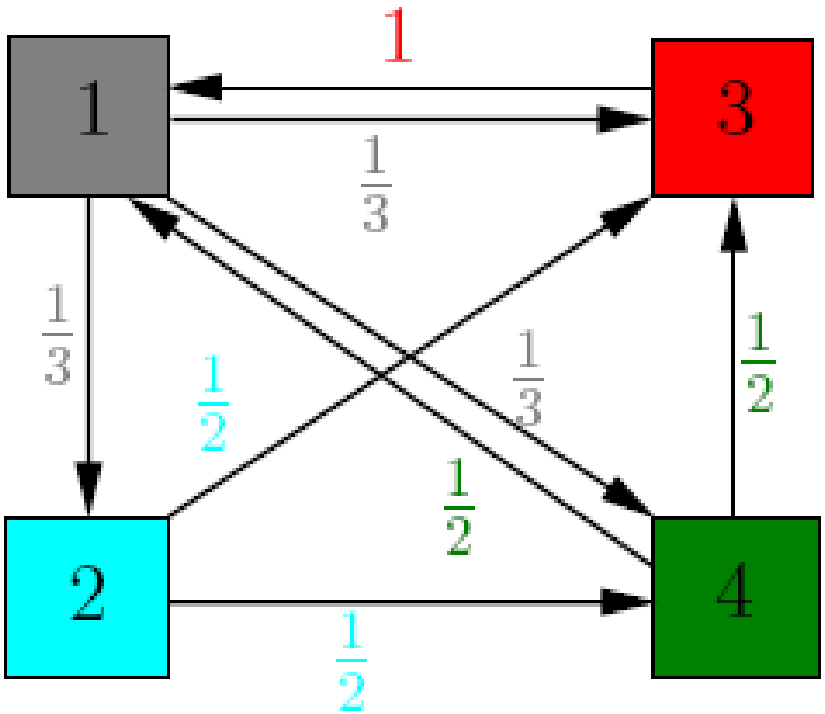
- PageRank is the algorithm to determine the importance of a website relatively to all other websites. The algorithm ranks the importance of website w, i.e., PR(w), based on the number of links points to website w and the *quality* of each pointing link from the other source website.
- **The underlying assumption**: More important website are likely to receive more links from other website. Since the algorithm measures the relative popularity ("ranking") between all websites, websites with higher ranking score are ranked higher. The sum of all ranking score equals 1 (will see why later).

Formalizing the PageRank problem



As a graph: Each website is represented by a node assigned with a PageRank value, denoted by PR(w). If a website w has a link to another website v (meaning there are an outbound link from w and an inbound link to v), then there is a directed edge from node w to node v. Multiple links from w to v is treated as a single edge from node w to v, and all self-links from a website to itself are ignored. Thus, this is a node-weighted, simple, no self-loop directed graph.

Edge Weights



This edge has its weight equal to PR(*w*) divided by the total number of outbound links of w, L(w). Then recipient node v "receives" the PageRank value of w, adding to its own value, i.e., PR(v) += PR(w)/L(w).
In another words, for a given node in the graph:
An outbound link will "give" away the PR value of the source node to the recipient node.
An inbound link will add the PR value from the source node to the recipient node.

Perspectives for Solving

- There are two ways to understand the problem:

1) as an Eigenvector problem

2) as a probability problem

- Both perspectives use linear algebra

Eigenvector Problem

Probability Problem

Power Iteration

Spider Traps and Deadends

Random Teleports + Damping factor

The solution to the Spider traps

- At each time step, the random surfer chooses 1 of 2 options:
- With some probability β, the surfer will follow a random link
- With the remaining probability 1 -β , the surfer will jump to random page
- This means within a few time steps, the surfer will teleport out of the spider trap within a few time steps

The Solution to Dead Ends

- Dead end problem: the pages that have zero scores, which leads to their PageRank score not getting distributed to any other page in the graph, not pointing to any other graph
- After running the power iteration, after several iterations, it will converge towards zero, which leads to the PageRank score of a particular node not being passed down and leaking out of the system
- How to solve the Dead Ends Problem:
  - Solution: Always Teleports
  - If a node has no outgoing links, when we reach that node, we teleport with a probability of 1
  - this means that whenever we reach the dead-end node, we will always jump out and reach somewhere else

Why Do Teleports Solve the problem?

- For any start vector, the power method applied to a Markov transition matrix A will converge to a unique positive stationary vector as long as A is stochastic, irreducible, and aperiodic.
  - Stochastic: Every column in the transition matrix sums to 1
  - Aperiodic: A chain is periodic if there exist k ¿ 1 such that the interval between two visits to some state s is always a multiple of k
  - Irreducible: From any state, there is a non-zero probability of going from any one state to another

The PageRank equation

- Google's solution makes the Markov Transition matrix stochastic, aperiodic, and irreducible.
- PageRank equation:

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1 - \beta)\frac{1}{n}$$

  - The summation is the sum of all of the importance of node *I* that point to it where r_j and r_i is the probability that the random surf is on this node.
  - This is divided by d_i, which is the probability that the random surf traverses the link when iterating towards j. This only happens with probability β, when the surfer decides to follow the link.
  - The $(1 - \beta)/n$ represents when the random walkers decide to jump somewhere else, using the probability $(1 - \beta) \cdot 1/n$, where n is the number of nodes in the entire network.