Hubs and Authorities

Social Networks Analysis and Graph Algorithms

Prof. Carlos Castillo — https://chato.cl/teach

Universitat Pompeu Fabra Barcelona

Sources

- D. Easley and J. Kleinberg (2010): Networks, Crowds, and Markets – Chapter 14
- Fei Li's lecture on PageRank (2011)
- Evimaria Terzi's lecture on link analysis (2013)
- URLs in the footer of specific slides

Motivation: rank search results

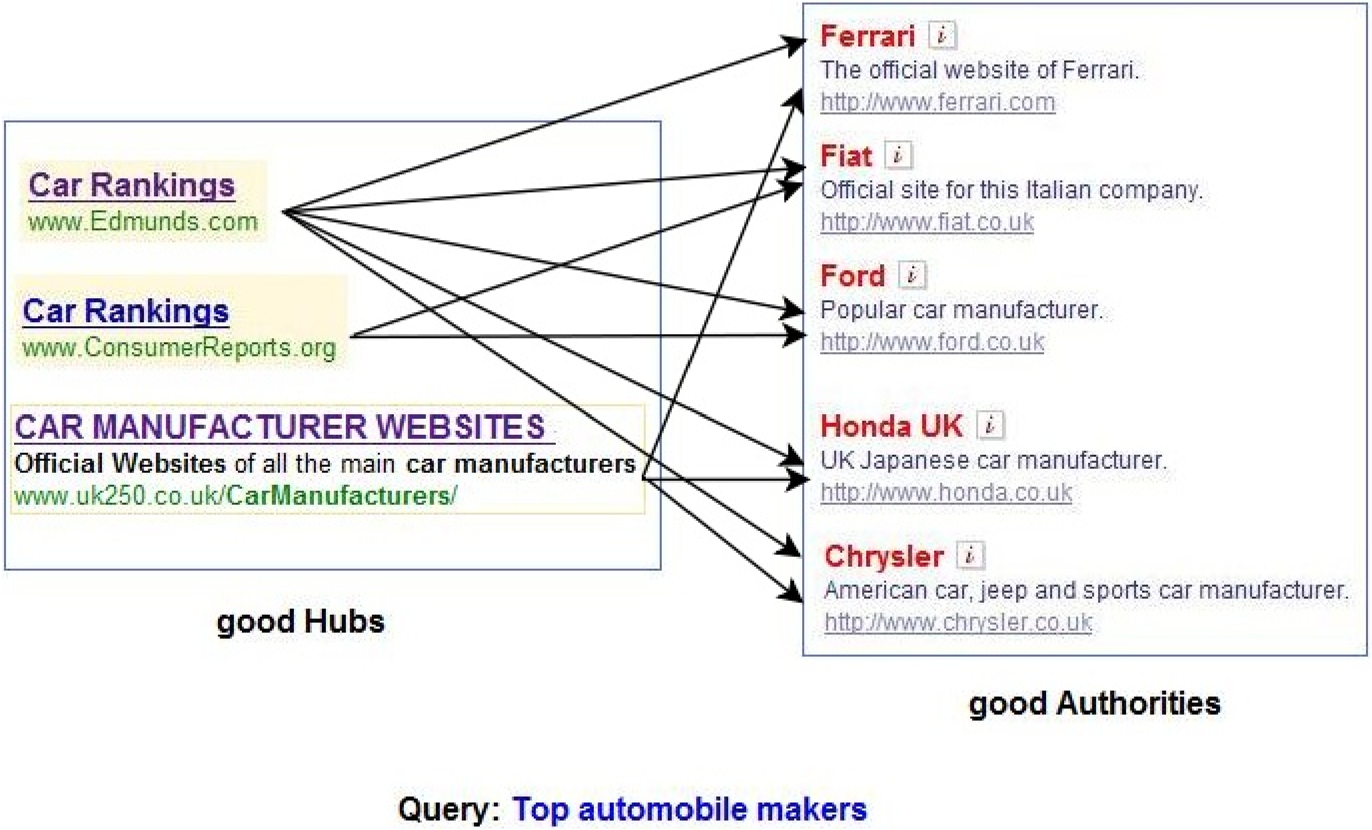
- Demand
  - Information needs are unclear and evolving
- Supply
  - From scarcity to abundance: “filter failure”

Purpose of Link-Based Ranking

- **Static (query-independent)** ranking
- **Dynamic (query-dependent)** ranking
- Applications:
  - Search in social networks
  - Search on the web

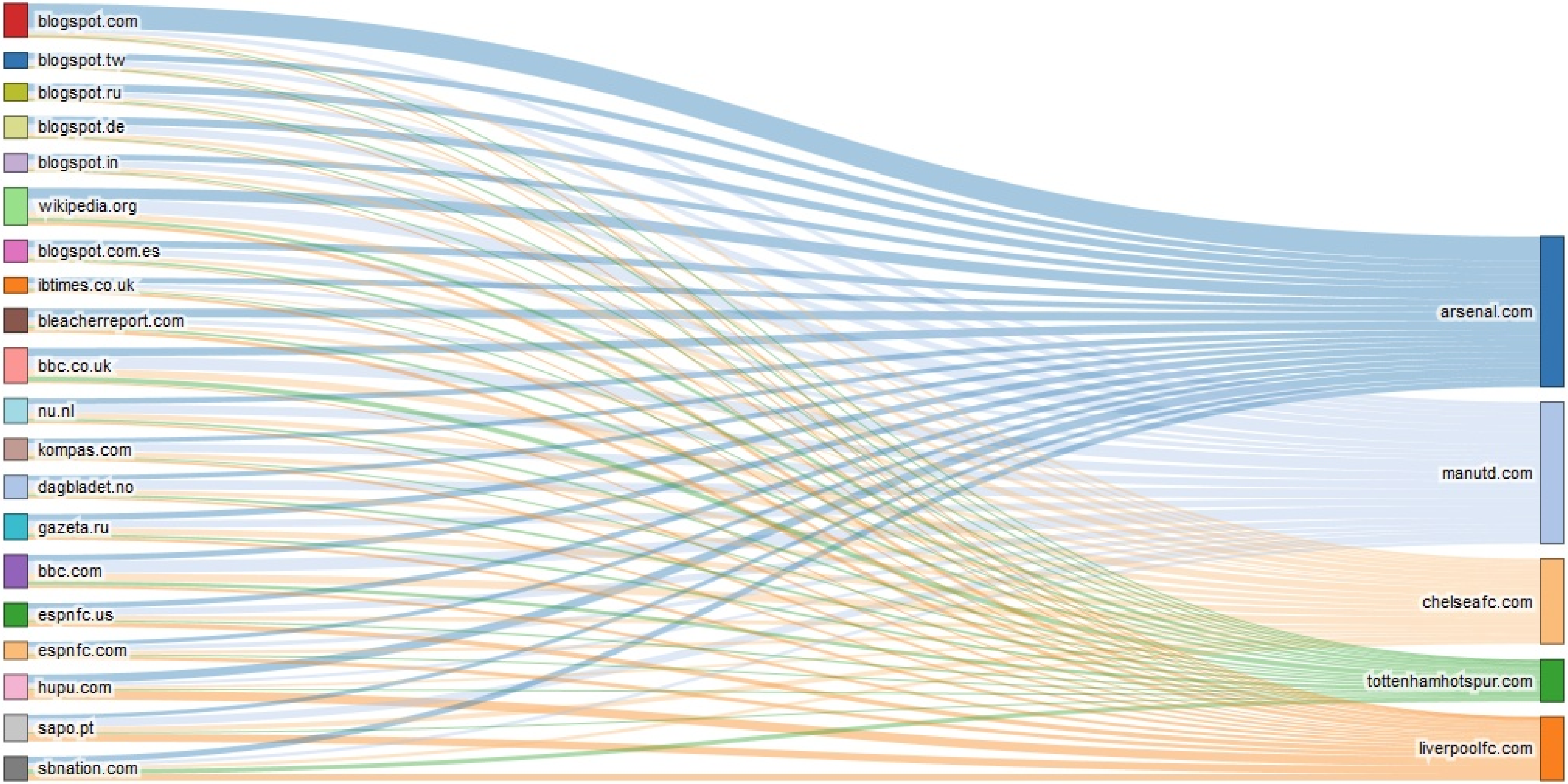
Why computing hubs and authorities?

Example 1: "top automobile makers"



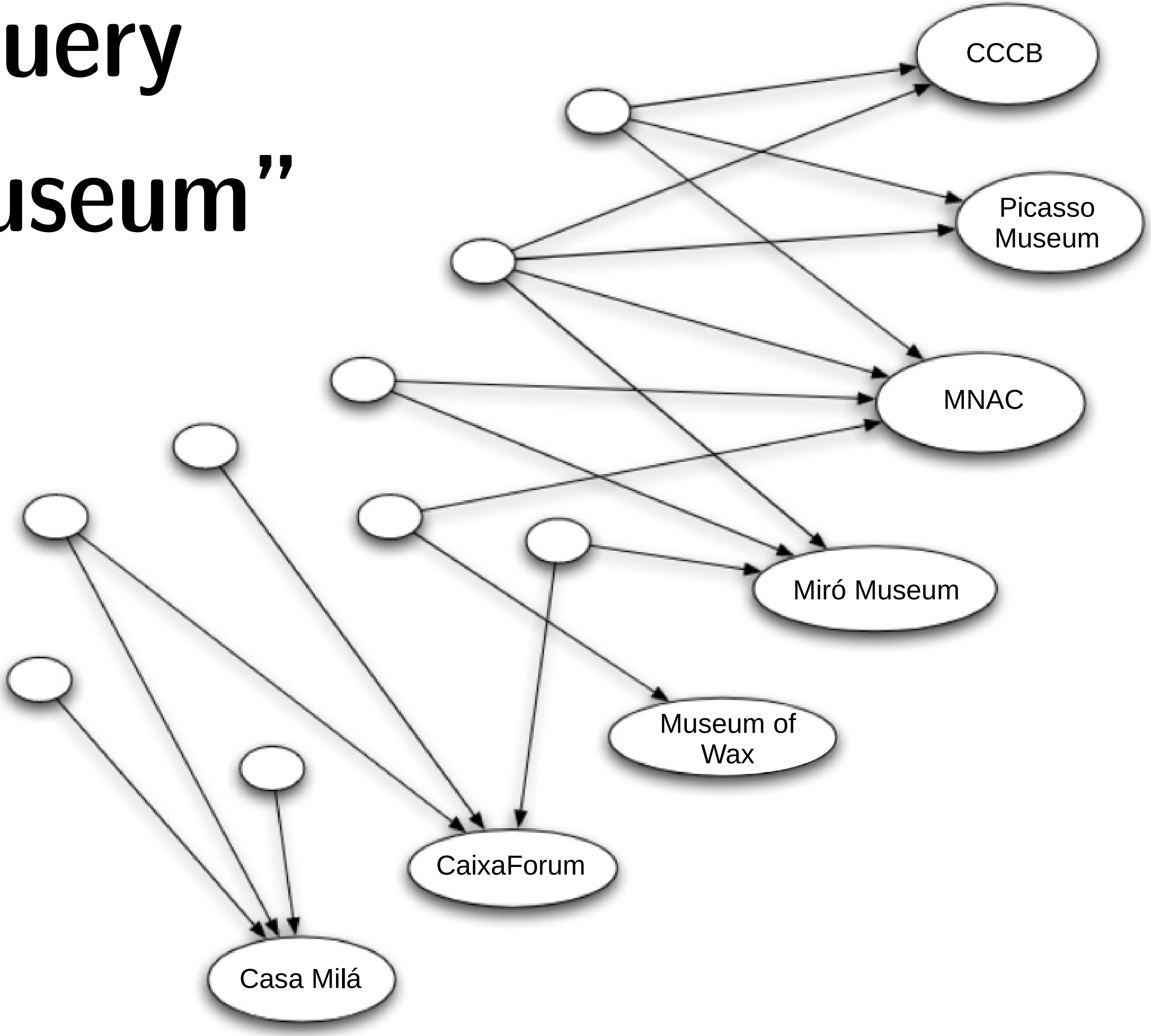
http://pi.math.cornell.edu/~mec/Winter2009/RalucaRemus/Lecture4/lecture4.html (2009)

Example 2: UK football teams on the web



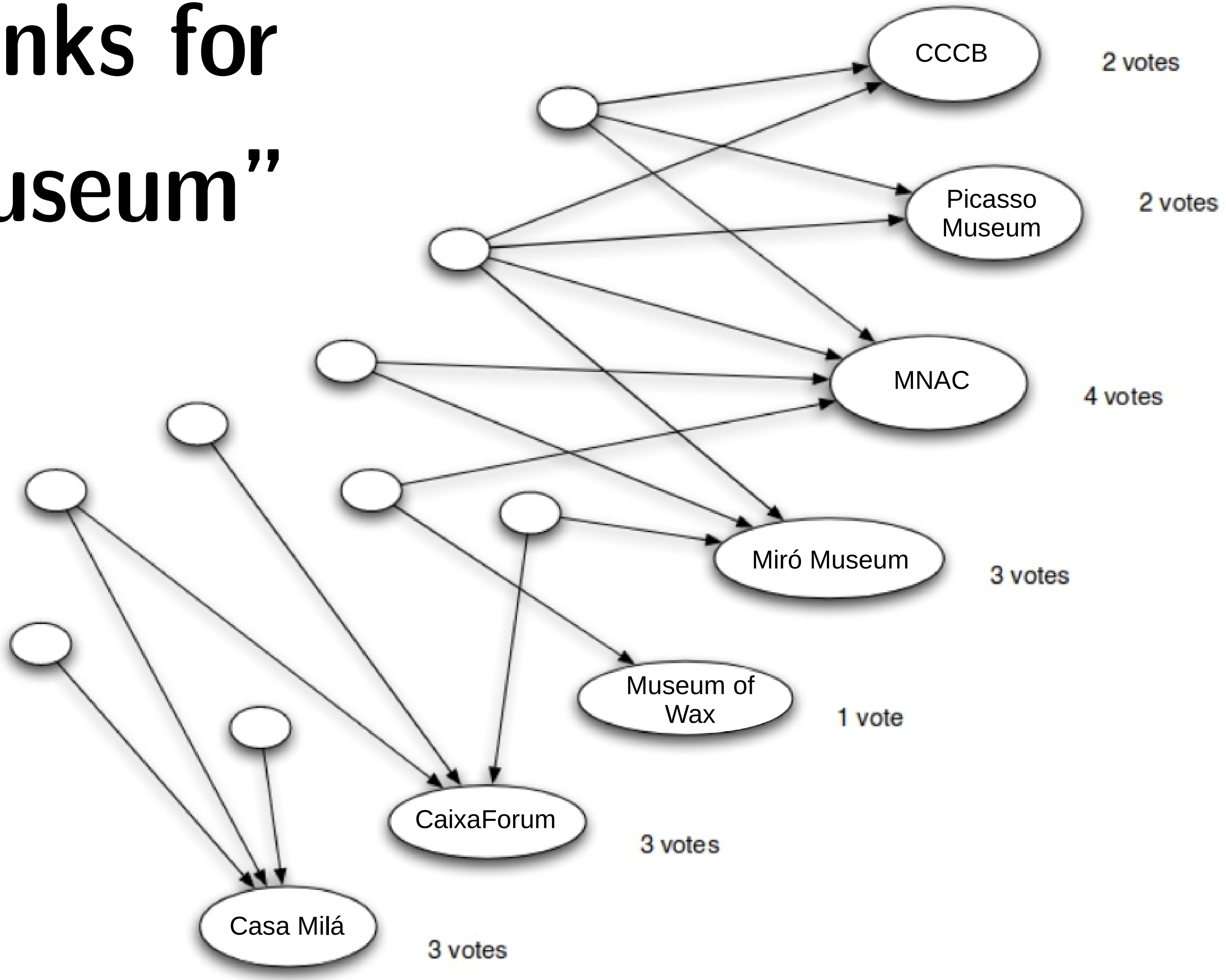
https://blog.majestic.com/development/hubs-of-authority-sankey-chart/ (2015)

Example: query "barcelona museum"

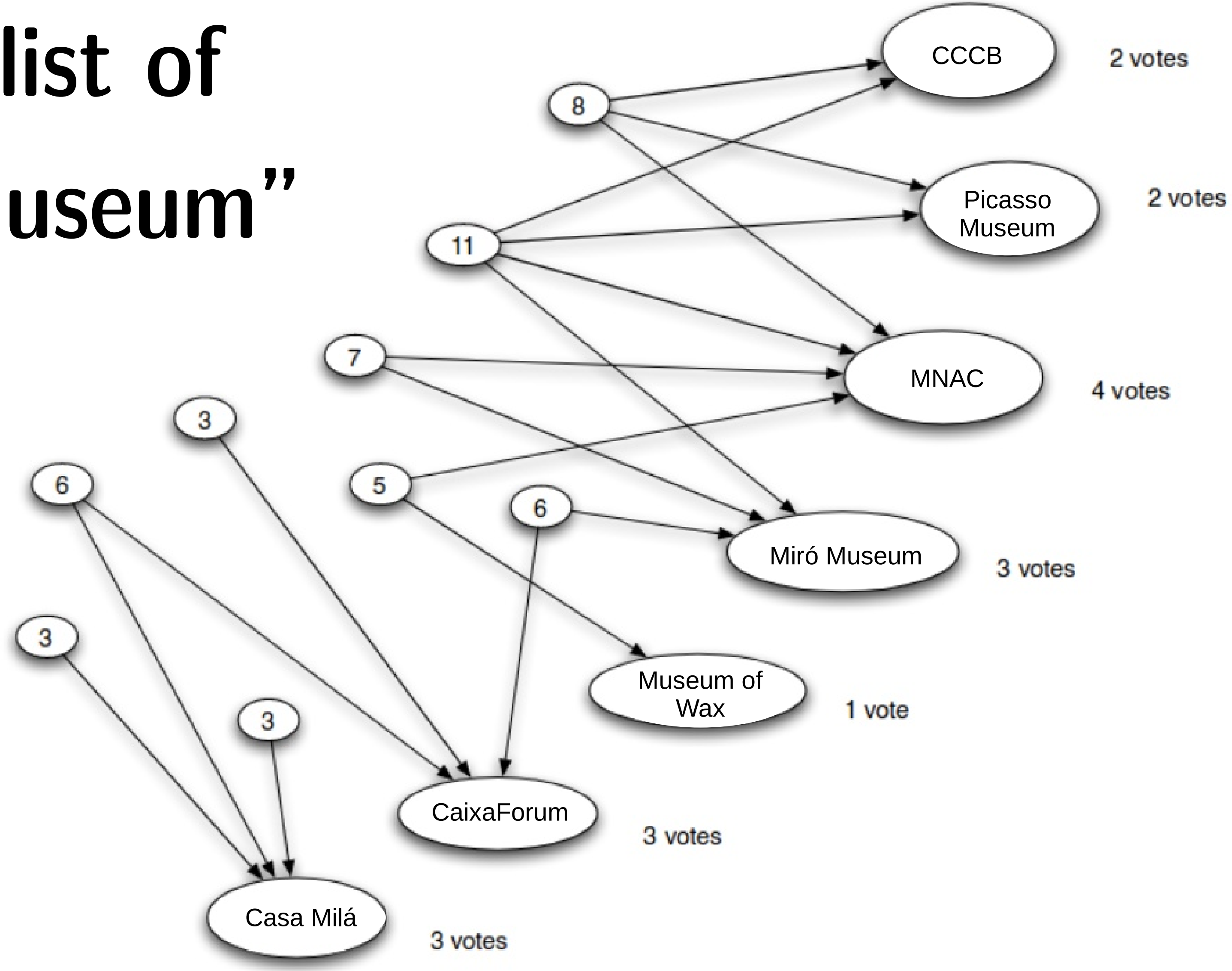


How would you rank these pages?

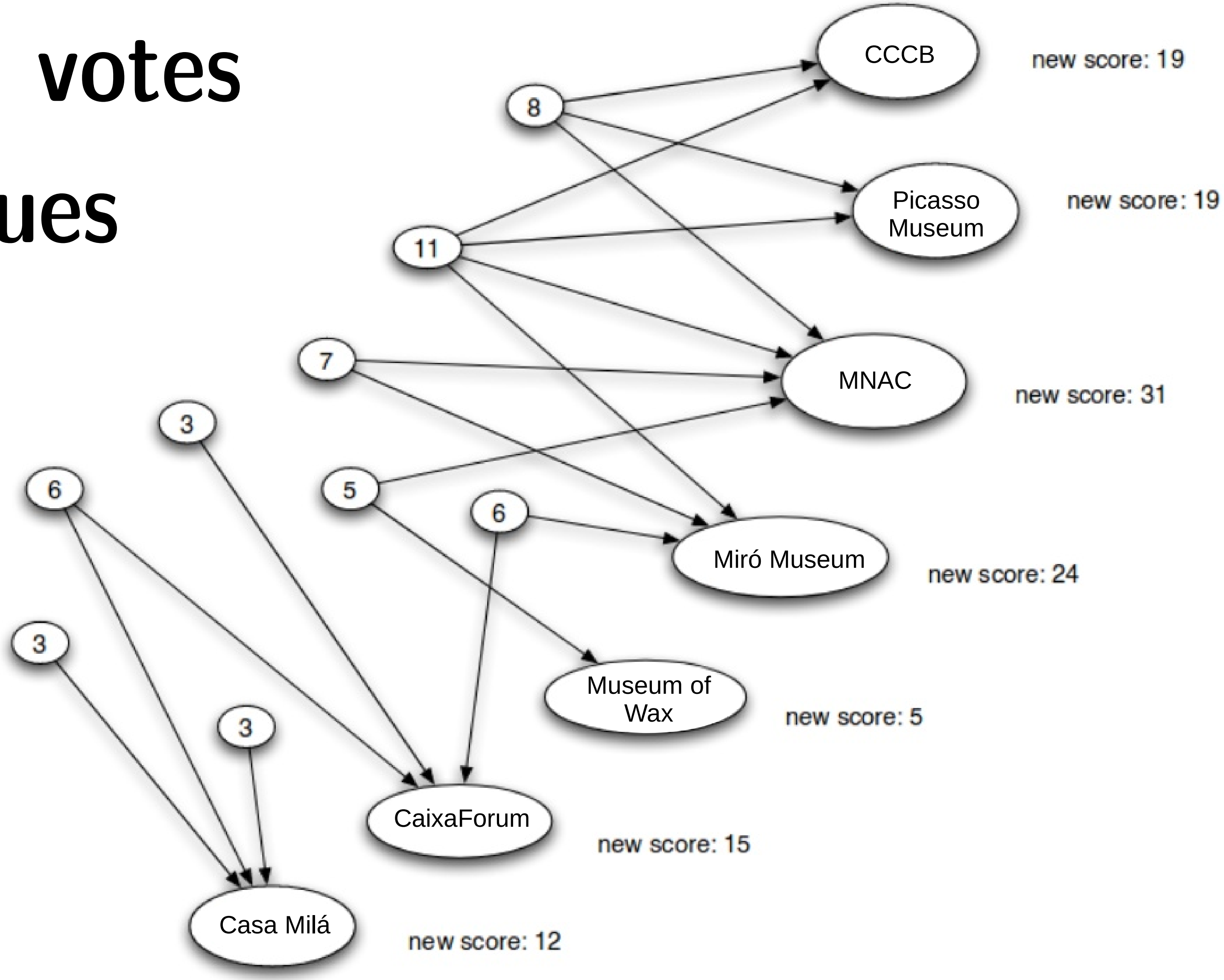
Counting in-links for "barcelona museum"



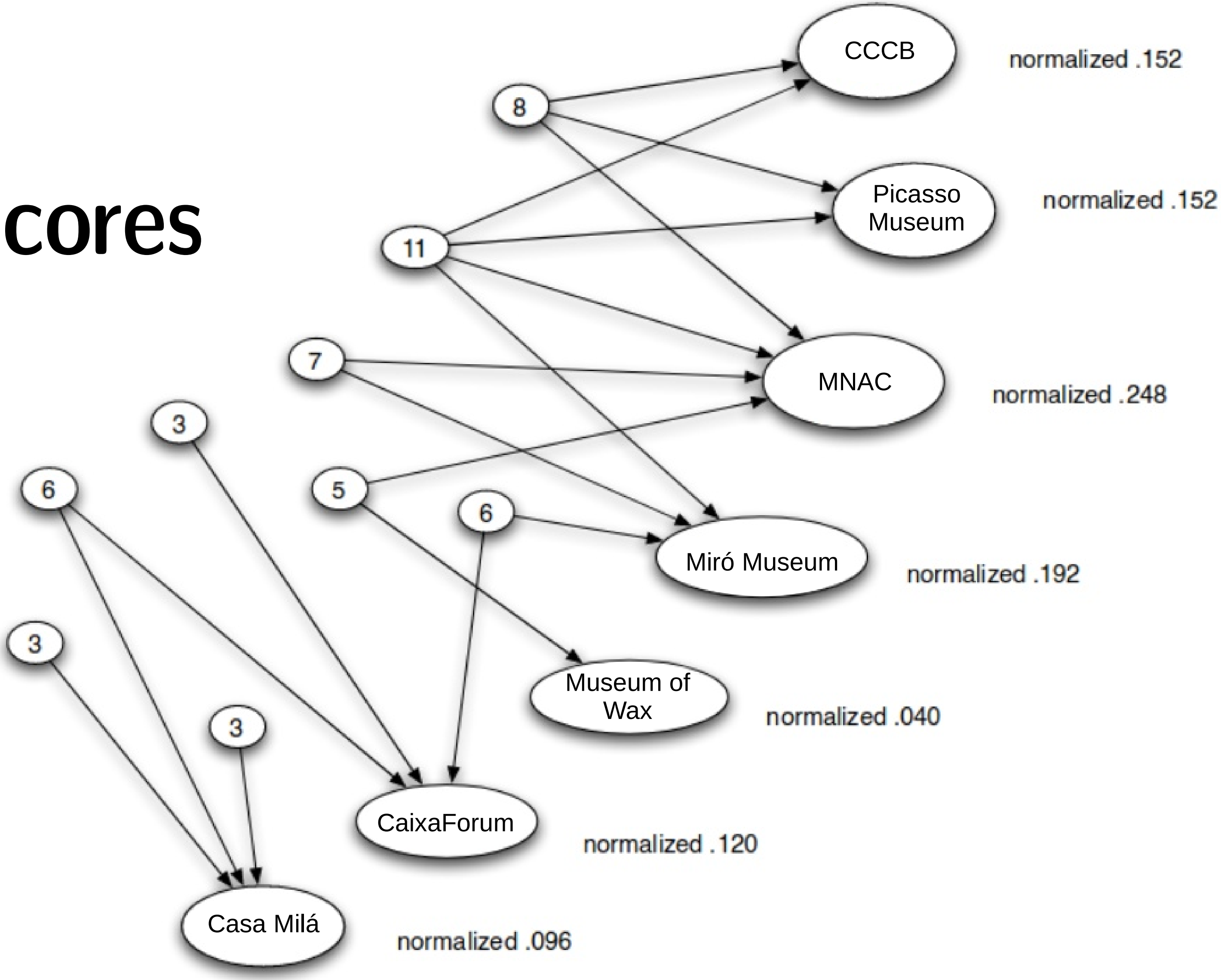
Value of a list of "barcelona museum"



Re-weighting votes by list values



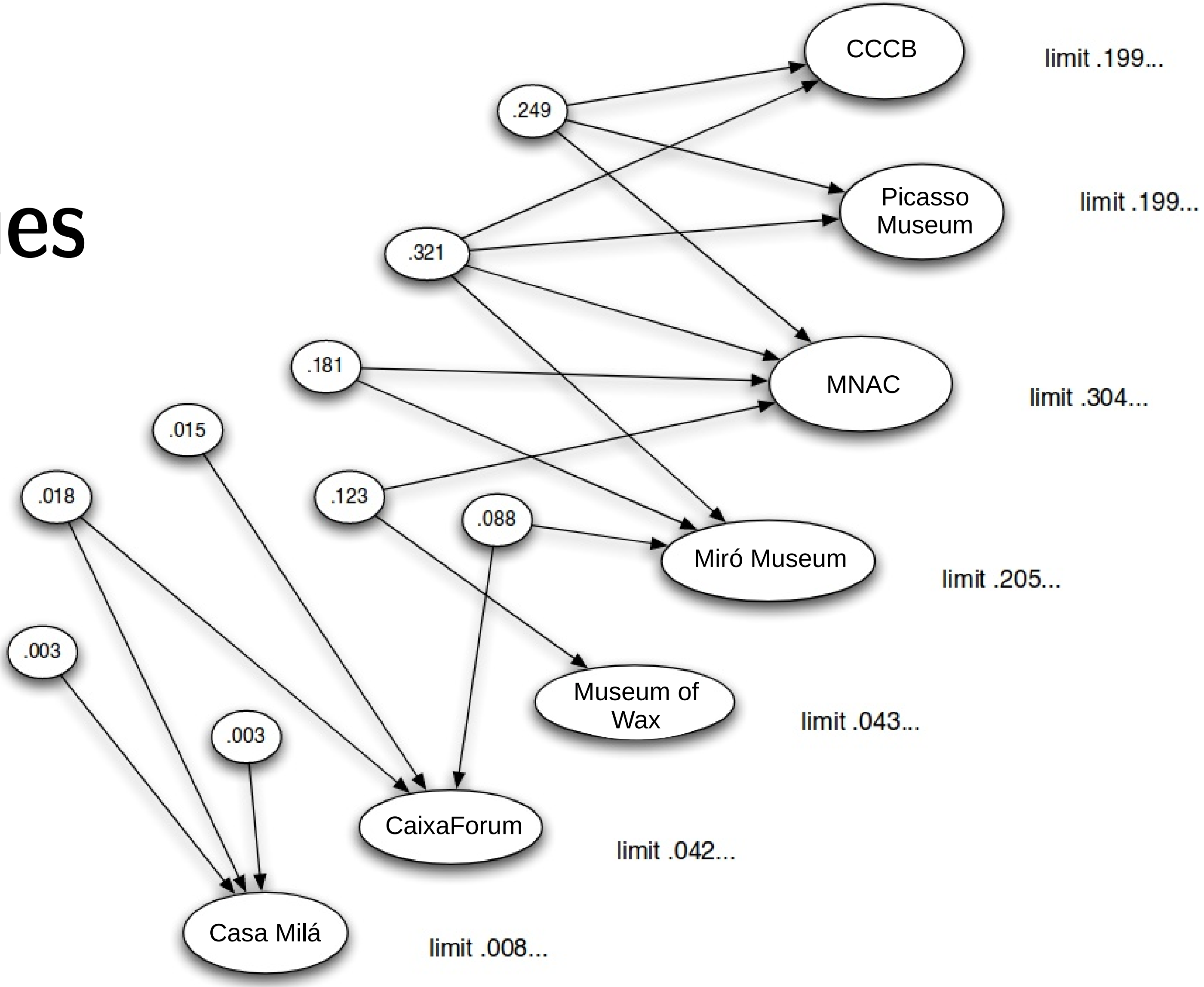
Normalizing scores



The idea behind Hubs and Authorities [Kleinberg 1999]

- Highly-recommended items
  appear in high-value lists
- High-value lists
  contain highly-recommended items
- **Repeated improvement**
  - Re-calculate scores several times

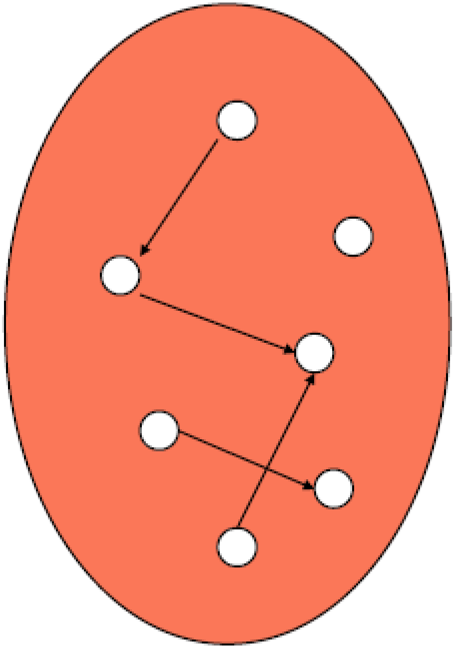
Limit values



This algorithm is called “HITS”

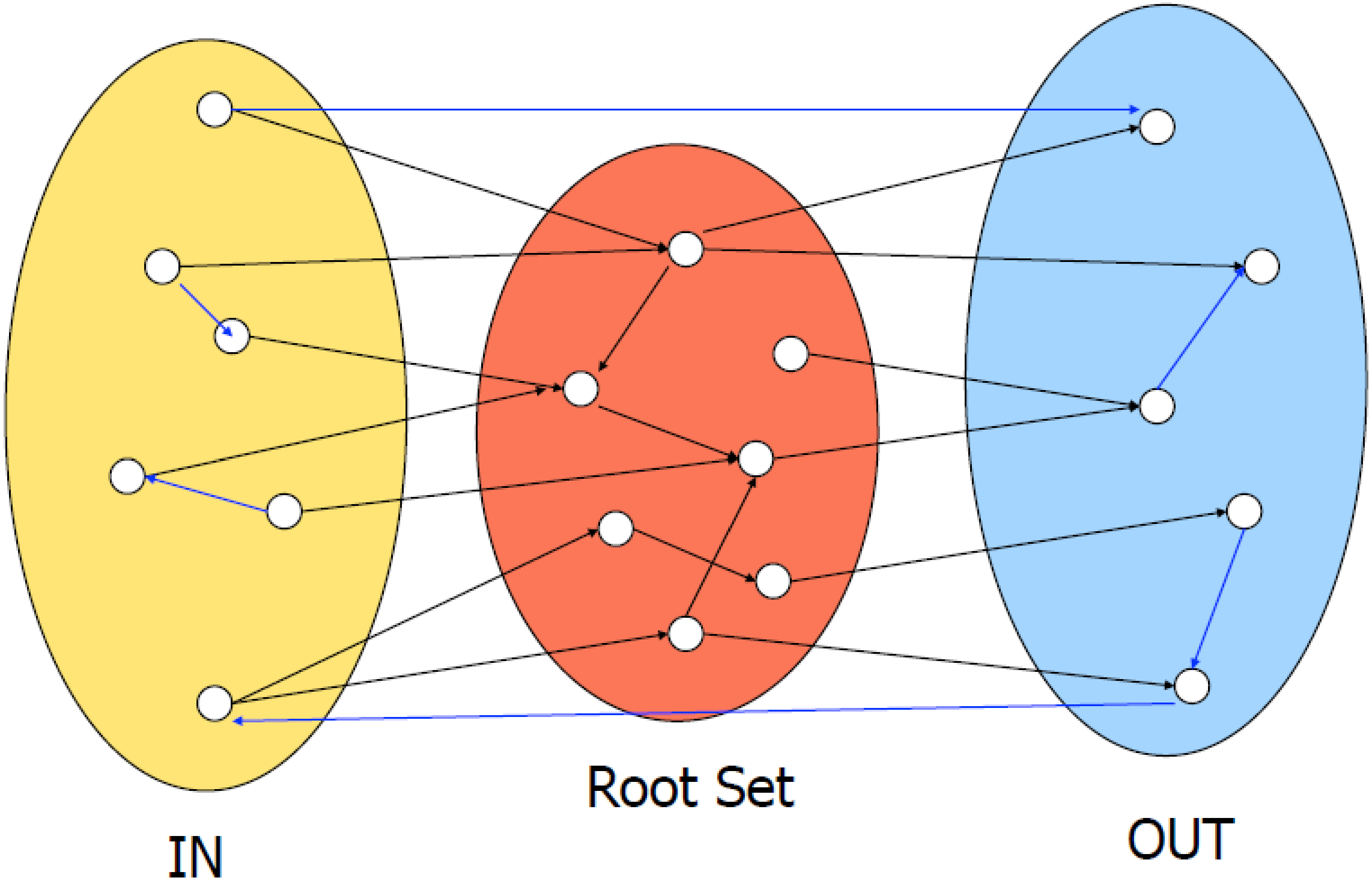
- *Jon M. Kleinberg. 1999. Authoritative sources in a hyperlinked environment. J. ACM 46, 5 (September 1999), 604-632. [DOI]*
- Query-dependent algorithm
  - Get pages matching the query
  - Expand to 1-hop neighborhood
  - Find pages with good out-links (“hubs”)
  - Find pages with good in-links (“authorities”)

Root set = matches the query



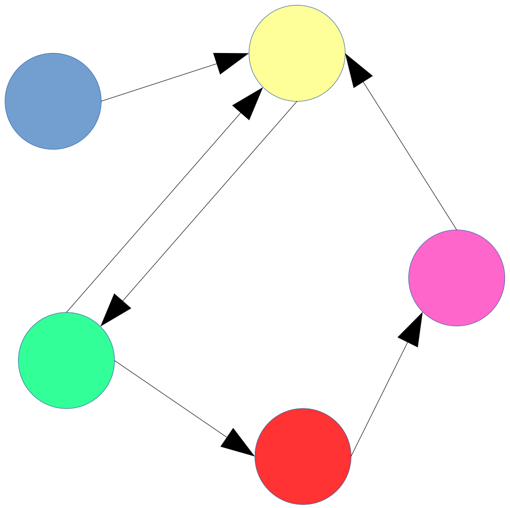
Root Set

Base set S = root set plus 1-hop neighbors

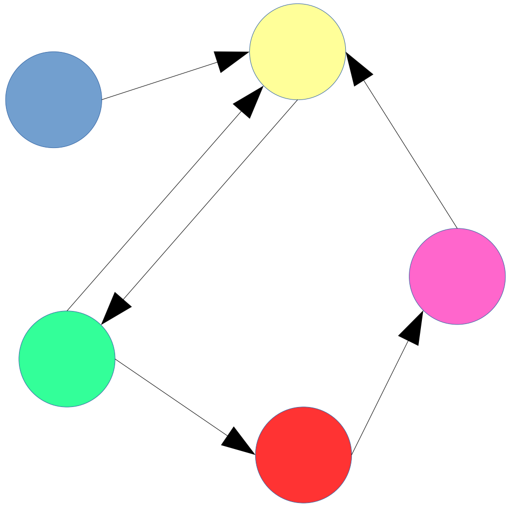


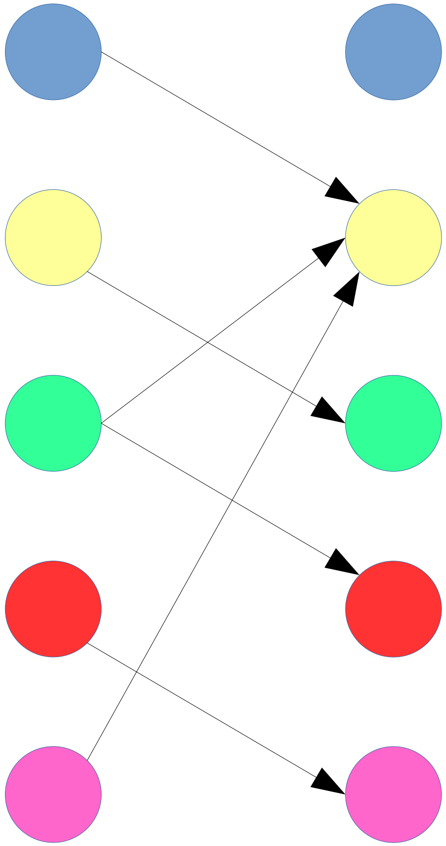
How to compute hubs and authorities

Base graph S of n nodes



Bipartite graph of $2n$ nodes





Bipartite graph of *2n* nodes

0) Initialization:

$$\mathrm{h}_i = \hat{h}_i = 1$$

1) Iteration:

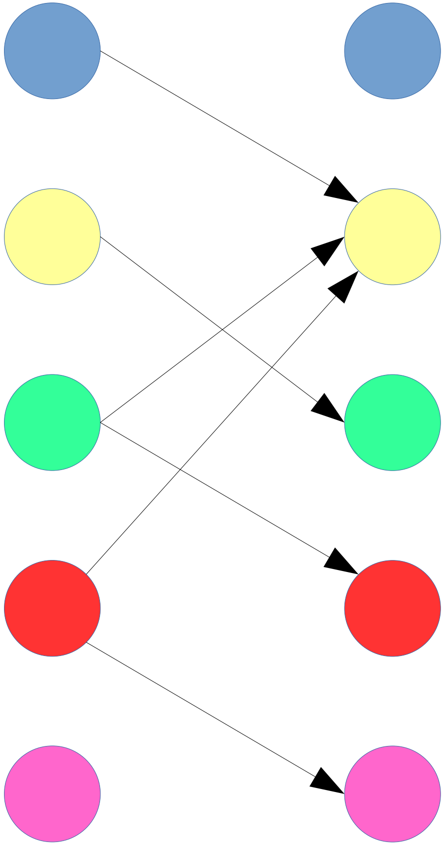
$$a_i = \sum_{j \to i} \hat{h}_j$$

$$h_i = \sum_{i \to j} \hat{a}_j$$

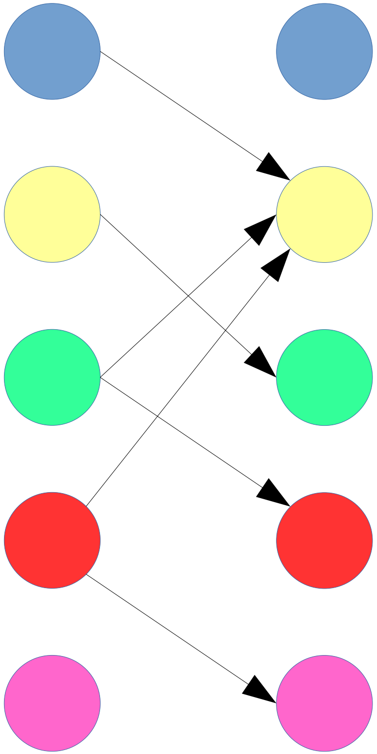
2) Normalization:

$$\hat{a}_i = \frac{a_i}{\sum_j a_j}$$

$$\hat{h}_i = \frac{h_i}{\sum_j h_j}$$



Exercise



Ĥ(1)	A(1)	Â(1)	H(2)	**Ĥ(2)**	A(2)	**Â(2)**
1	0					
1	3					
1	1					
1	1					
1	1					

Complete the table

Which one is the largest hub?

Which the largest authority?

Compare rankings

Rank by indegree = rank by auth?

Rank by outdegree = rank by hub?

Spreadsheet links: https://upfbarcelona.padlet.org/chato/shyq9m6f2g2dh1bw

What are we computing?

$$\begin{aligned}
a^t &= A^T h^{t-1} \\
h^t &= A a^t \\
\text{replacing}: \ a^t &= A^T A a^{t-1} \\
\text{after convergence}: \ a &= A^T A a
\end{aligned}$$

- Vector a is an eigenvector of $A^T A$
- Conversely, vector h is an eigenvector of $A A^T$

Dealing with weighted graphs

(this is an option that does not normalize weights,
one can alternatively normalize them)

$$\mathrm{h}_i = \hat{h}_i = 1$$

1) Iteration:

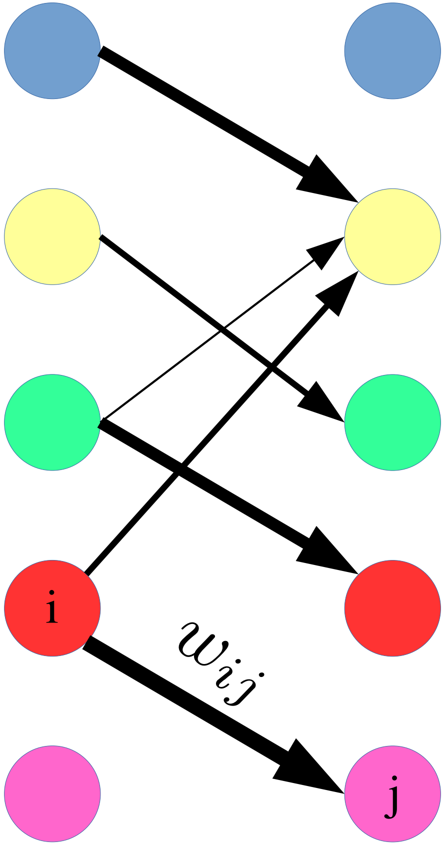
$$a_i = \sum_{j \to i} \left(w_{ji} \cdot \hat{h}_j \right)$$

$$h_i = \sum_{i \to j} (w_{ij} \cdot \hat{a}_j)$$

2) Normalization:

$$\hat{a}_i = \frac{a_i}{\sum_j a_j}$$

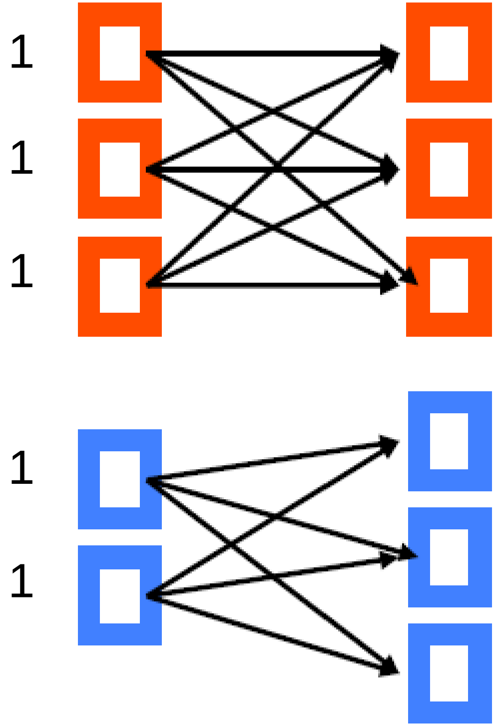
$$\hat{h}_i = \frac{h_i}{\sum_j h_j}$$



Problem: cliques and quasi-cliques

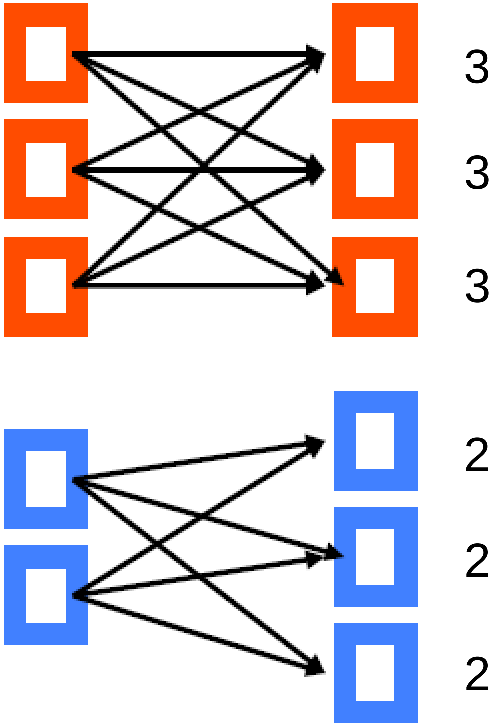
Problem: tightly-knit communities

- Example: a graph made of a (3,3)-clique and a (2,3)-clique



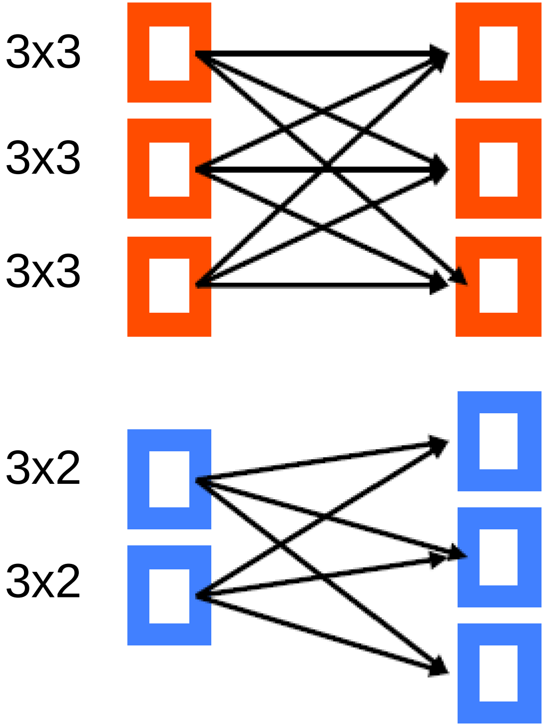
Problem: tightly-knit communities

- Example: a graph made of a (3,3)-clique and a (2,3)-clique



Problem: tightly-knit communities

- Example: a graph made of a (3,3)-clique and a (2,3)-clique

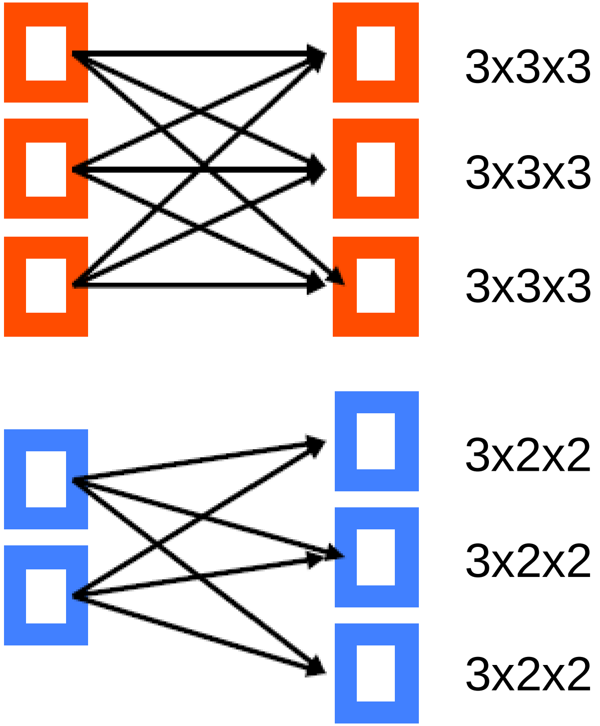


Problem: tightly-knit communities

- Example: a graph made of a (3,3)-clique and a (2,3)-clique

What happens after n iterations?

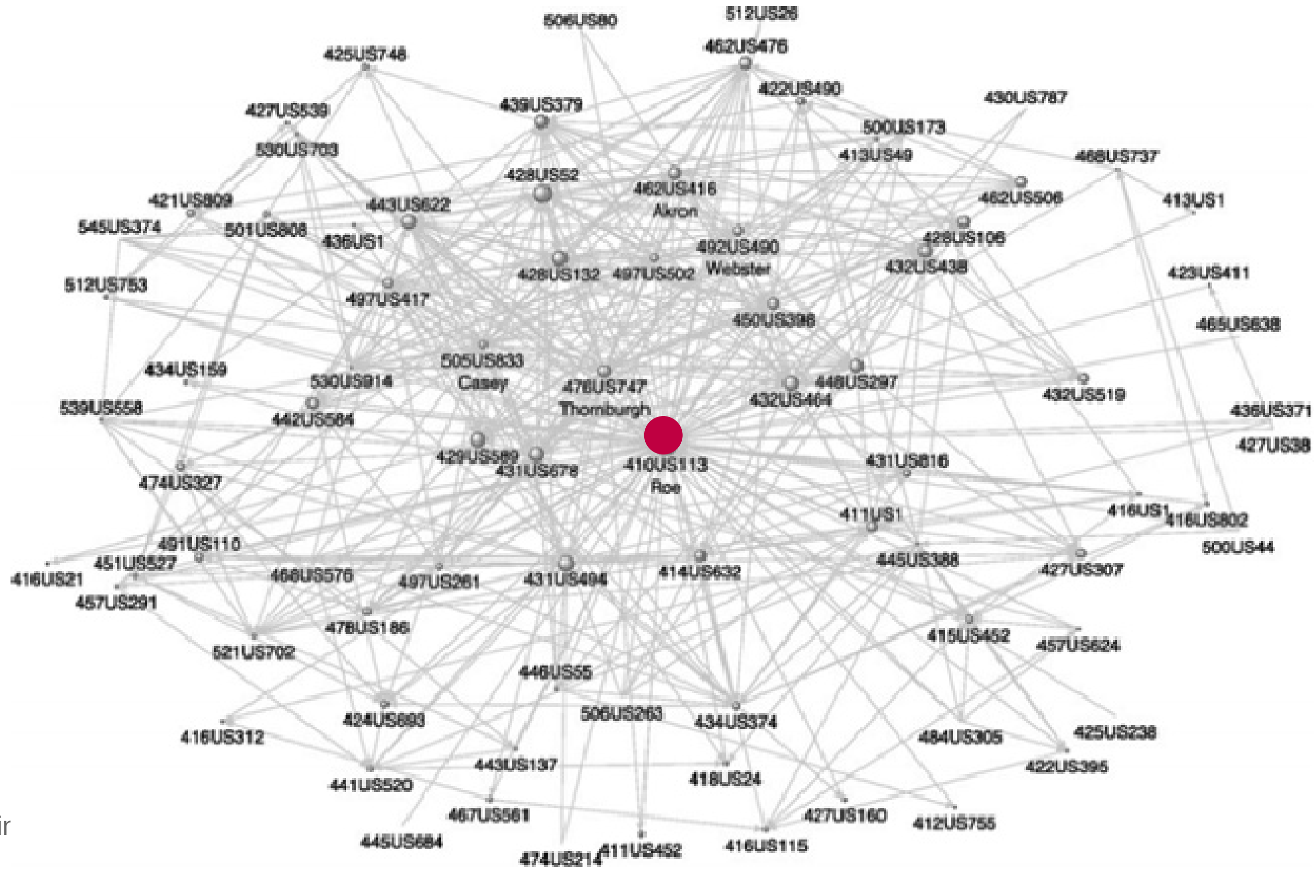
Which community "wins" (i.e., has the largest sum of scores)?



A different application of hubs and authorities

The legal precedent network

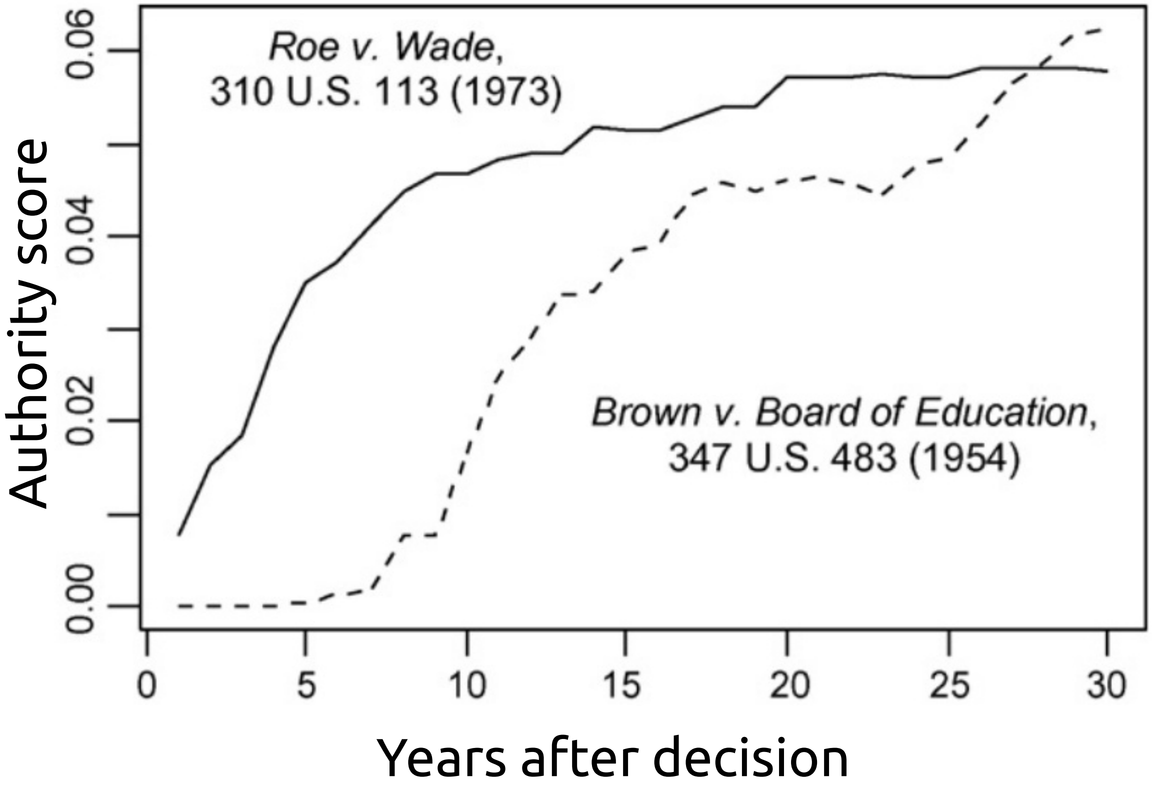
- Roe v Wade legalized abortion in the US
- Many cases reference it as a legal precedent
- This is a representation of some of those cases



Carmichael, I., Wudel, J., Kim, M., & Jushchuk, J. (2017). Examinir the evolution of legal precedent through citation network analysis. NCL Rev., 96, 227.

Hubs and authorities on the legal precedent network

- We can compute authority in this network
- Re-compute every year
- **Different cases acquire authority at different speeds!**



(Roe v Wade legalized abortion, Brown v Board of Education declared race-segregated schools unconstitutional)

Fowler, J. H., & Jeon, S. (2008). The authority of Supreme Court precedent. Social networks, 30(1), 16-30.

Summary

Things to remember

- What is the hubs and authority algorithm
- How to execute it step by step
- Practice with graphs on your own

Practice on your own

- Consider a directed bi-partite graph $G = (V_L \cup V_R, E)$ in which $V_L = \{a, b, c, d\}$ and $V_R = \{1, 2, \ldots, 120\}$, and in which all edges go from a node in V_L to a node in V_R :
  - Node a is connected to nodes 1, 2, . . . 120.
  - Node b is connected to nodes 1, 2, . . . 60.
  - Node c is connected to nodes 1, 2, . . . 30.
  - Node d is connected to nodes 1, 2, . . . 15.
- Starting with $\hat{h}(1)(i) = 1$ for $i \in \{a, b, c, d, 1, 2, \ldots, 120\}$.
  - 1. Compute $a(1)(i)$ for $i \in \{1, 2, \ldots, 120\}$
  - 2. Compute $\hat{a}(1)(i)$ for $i \in \{1, 2, \ldots, 120\}$
  - 3. Compute $h(2)(i)$ for $i \in \{a, b, c, d\}$