Applications

of

Nonnegative Matrices:

Ranking

and

Clustering

Amy Langville
Mathematics Department
College of Charleston

Hamilton Institute 8/7/2008

Collaborators

- Carl Meyer, N. C. State University
- David Gleich, Stanford University
- Michael Berry, University of Tennessee-Knoxville
- College of Charleston (current and former students)
  - Luke Ingram
  - Ibai Basabe
  - Kathryn Pedings
  - Neil Goodson
  - Colin Stephenson
  - Emmie Douglas

Nonnegative Matrices

- Ranking webpages
- Ranking sports teams
- Recommendation systems
- Meta-algorithms
  - Rank and rating aggregation
  - Cluster aggregation

Ranking webpages

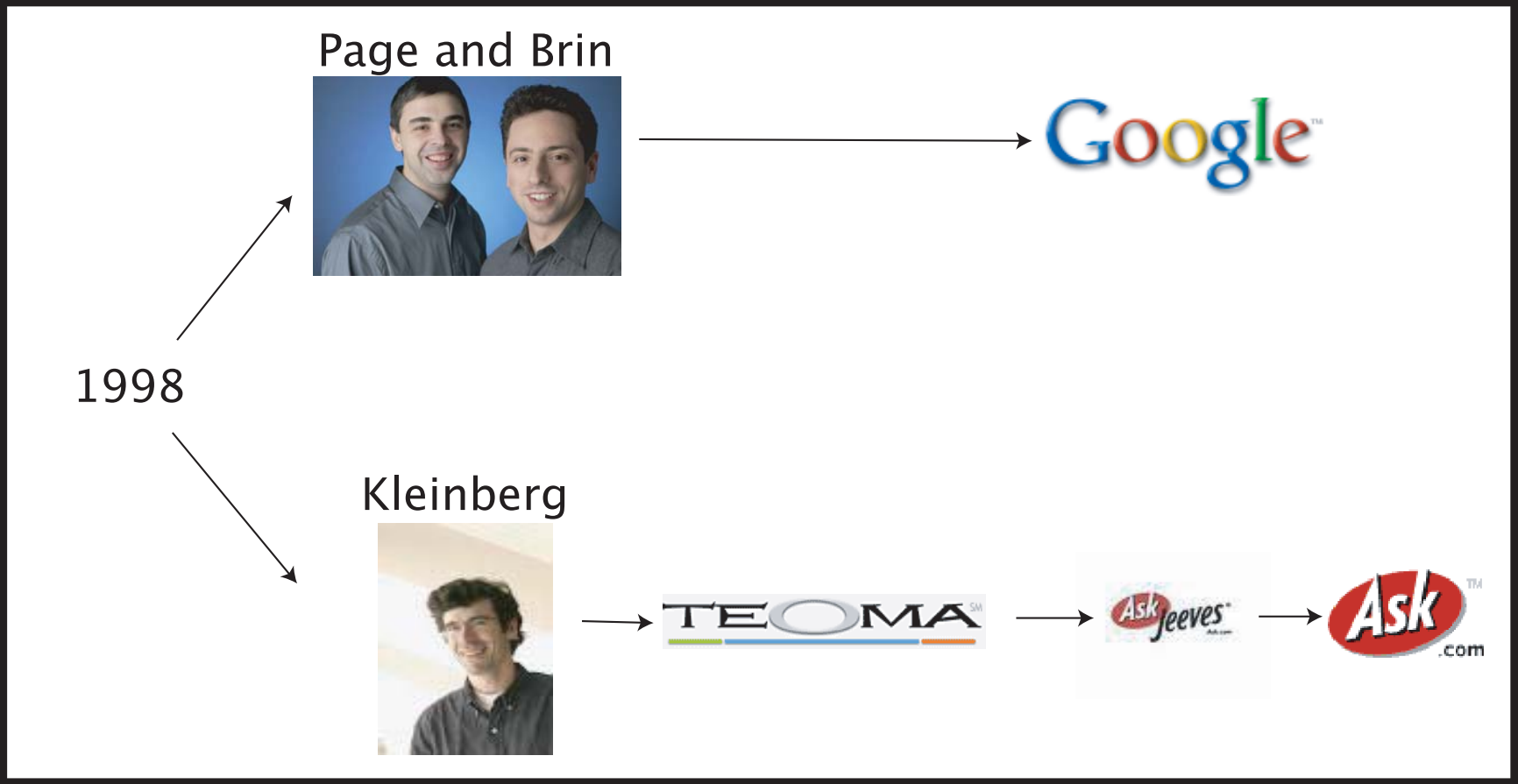
Google's PageRank and Beyond

THE SCIENCE OF SEARCH ENGINE RANKINGS

AMY N. LANGVILLE and CARL D. MEYER

1998: enter Link Analysis

- uses *hyperlink* structure to focus the relevant set
- combine traditional IR score with popularity score



Web Information Retrieval

IR before the Web = traditional IR
IR on the Web = web IR

Web Information Retrieval

IR before the Web = traditional IR
IR on the Web = web IR

How is the Web different from other document collections?

Web Information Retrieval

IR before the Web = traditional IR
IR on the Web = web IR

How is the Web different from other document collections?

- It's huge.
  - over 10 billion pages, average page size of 500KB
  - 20 times size of Library of Congress print collection
  - Deep Web - 400 X bigger than Surface Web

Web Information Retrieval

IR before the Web = traditional IR
IR on the Web = web IR

How is the Web different from other document collections?

- It's huge.
  - over 10 billion pages, average page size of 500KB
  - 20 times size of Library of Congress print collection
  - Deep Web - 400 X bigger than Surface Web
- It's dynamic.
  - content changes: 40% of pages change in a week, 23% of .com change daily
  - size changes: billions of pages added each year

Web Information Retrieval

IR before the Web = traditional IR
IR on the Web = web IR

How is the Web different from other document collections?

- It's huge.
  - over 10 billion pages, average page size of 500KB
  - 20 times size of Library of Congress print collection
  - Deep Web - 400 X bigger than Surface Web
- It's dynamic.
  - content changes: 40% of pages change in a week, 23% of .com change daily
  - size changes: billions of pages added each year
- It's self-organized.
  - no standards, review process, formats
  - errors, falsehoods, link rot, and spammers!

Web Information Retrieval

IR before the Web = traditional IR
IR on the Web = web IR

How is the Web different from other document collections?

- It's huge.
  - over 10 billion pages, average page size of 500KB
  - 20 times size of Library of Congress print collection
  - Deep Web - 400 X bigger than Surface Web
- It's dynamic.
  - content changes: 40% of pages change in a week, 23% of .com change daily
  - size changes: billions of pages added each year
- It's self-organized.
  - no standards, review process, formats
  - errors, falsehoods, link rot, and spammers!

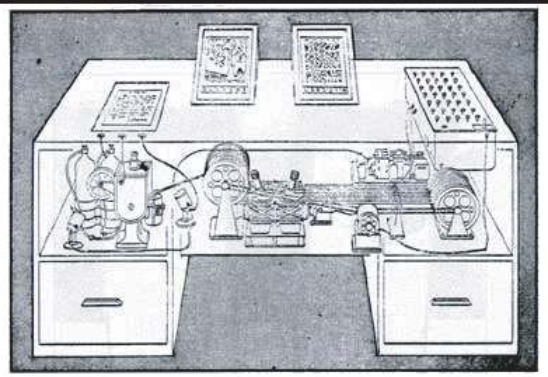
A Herculean Task!

Web Information Retrieval

IR before the Web = traditional IR
IR on the Web = **web IR**

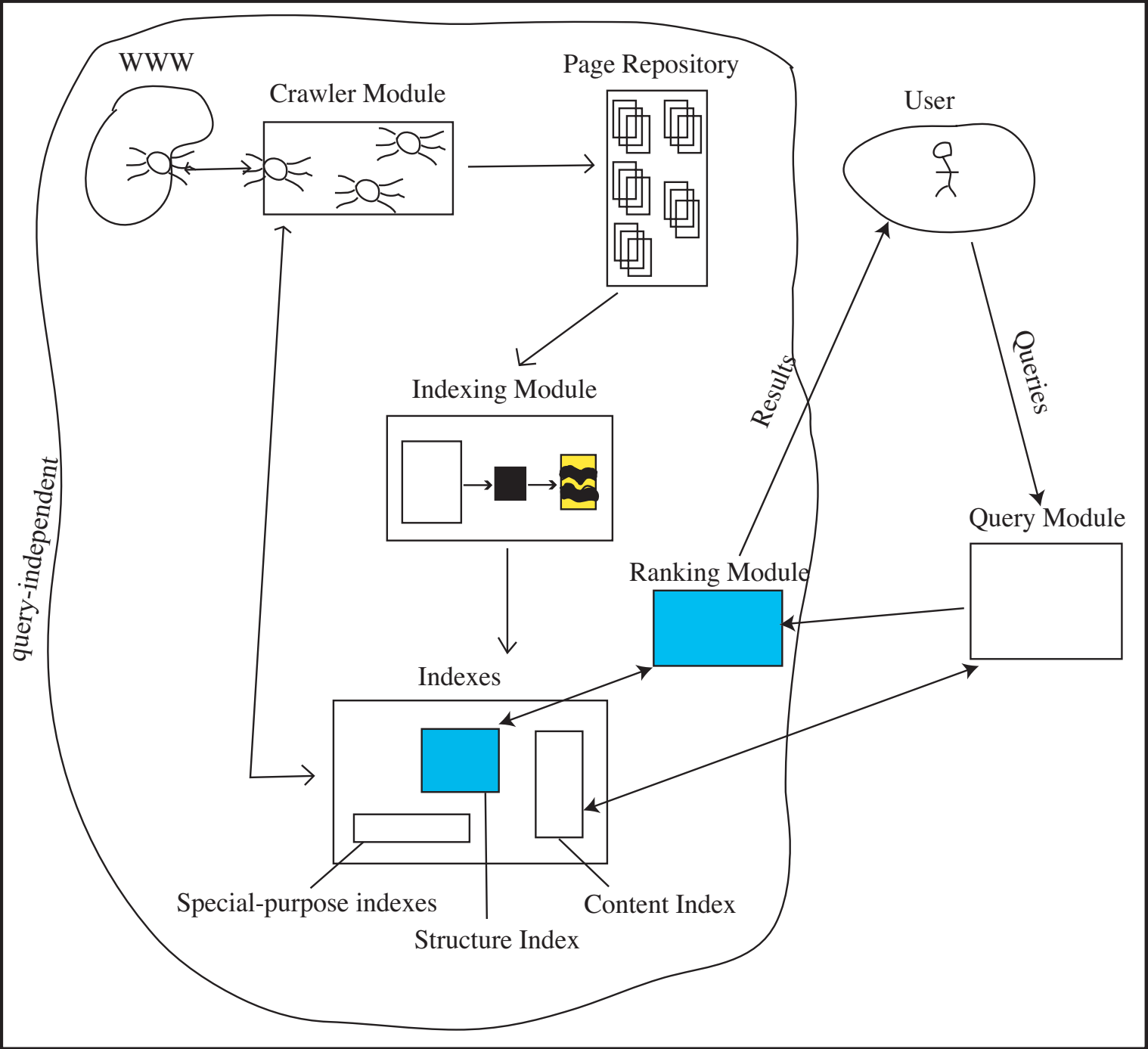
How is the Web different from other document collections?

- It's huge.
  - over 10 billion pages, each about 500KB
  - 20 times size of Library of Congress print collection
  - Deep Web - 400 X bigger than Surface Web
- It's dynamic.
  - content changes: 40% of pages change in a week, 23% of .com change daily
  - size changes: billions of pages added each year
- It's self-organized.
  - no standards, review process, formats
  - errors, falsehoods, link rot, and spammers!
- Ah, but it's *hyperlinked* !
  - Vannevar Bush's 1945 memex



Memex in the form of a desk would instantly bring files and material on any subject to the operator's fingertips. Slanting translucent viewing screens magnify supermicrofilm filed by code numbers. At left is a mechanism which automatically photographs longhand notes, pictures and letters, then files them in the desk for future reference (*LIFE* 19(11), p. 123).

Elements of a Web Search Engine



The Ranking Module (generates popularity scores)

- Measure the importance of each page

The Ranking Module (generates popularity scores)

- Measure the importance of each page
- The measure should be Independent of any query
  - Primarily determined by the link structure of the Web
  - Tempered by some content considerations

The Ranking Module (generates popularity scores)

- Measure the importance of each page
- The measure should be Independent of any query
  - Primarily determined by the link structure of the Web
  - Tempered by some content considerations
- Compute these measures off-line long before any queries are processed

The Ranking Module (generates popularity scores)

- Measure the importance of each page
- The measure should be Independent of any query
  - Primarily determined by the link structure of the Web
  - Tempered by some content considerations
- Compute these measures off-line long before any queries are processed
- Google's PageRank© technology distinguishes it from all competitors

The Ranking Module (generates popularity scores)

- Measure the importance of each page
- The measure should be Independent of any query
  - Primarily determined by the link structure of the Web
  - Tempered by some content considerations
- Compute these measures off-line long before any queries are processed
- Google's PageRank© technology distinguishes it from all competitors

Google's PageRank = Google's $$$$$

Sign in

Google

Web Images Video News Maps more »

business intelligence

Search

Advanced Search
Preferences

Web Results **1** - **10** of about **122,000,000** for **business intelligence**. (**0.10** seconds)

SAS Business Intelligence Sponsored Link
www.SAS.com Get Better Answers Faster w/ SAS' Award-winning BI Software. Get Info

Business intelligence - Wikipedia, the free encyclopedia
Business intelligence (**BI**) is a **business** management term which refers to applications and technologies which are used to gather, provide access to, ...
en.wikipedia.org/wiki/**Business_intelligence** - 43k - Cached - Similar pages

Business Intelligence .com :: The Resource for **Business Intelligence**
The **Business Intelligence** resource for **business** and technical professionals covering a wide range of topics including Performance Management, Data Warehouse ...
www.**businessintelligence**.com/ - 74k - Apr 15, 2007 - Cached - Similar pages

Business Intelligence and Performance Management Software ...
Business intelligence and **business** performance management software. Reporting, analytics software, budgeting software, balanced scorecard software, ...
Stock quote for COGN
www.cognos.com/ - 32k - Cached - Similar pages

Oracle **Business Intelligence** Solutions
The First Comprehensive, Cost-Effective BI Solution Only Oracle delivers a complete, pre-integrated technology foundation to reduce the cost and complexity ...
www.oracle.com/solutions/ **business_intelligence**/index.html - 55k - Cached - Similar pages

Business Intelligence - Management Best Practice Reports
Business Intelligence: Providers of independent reports containing best practice advice, proprietary research findings and case studies for senior managers ...
www.**business-intelligence**.co.uk/ - 18k - Cached - Similar pages

Intelligent Enterprise: Better Insight for **Business** Decisions

Sponsored Links

SQL Database Management
Enterprise Data Mgmt Solutions From Dell™. Find Out More Here
www.dell.com

Business Intelligence
See what **business intelligence** can do for you (free interactive demo).
www.InformationBuilders.com

MCITP: BI Cert Boot Camp
9-Day MCITP Certification Boot Camp **Business Intelligence** All Inclusive
www.mcseclasses.com

Business Intelligence
Improve information integrity with real-time data integration software
www.DataMirror.com

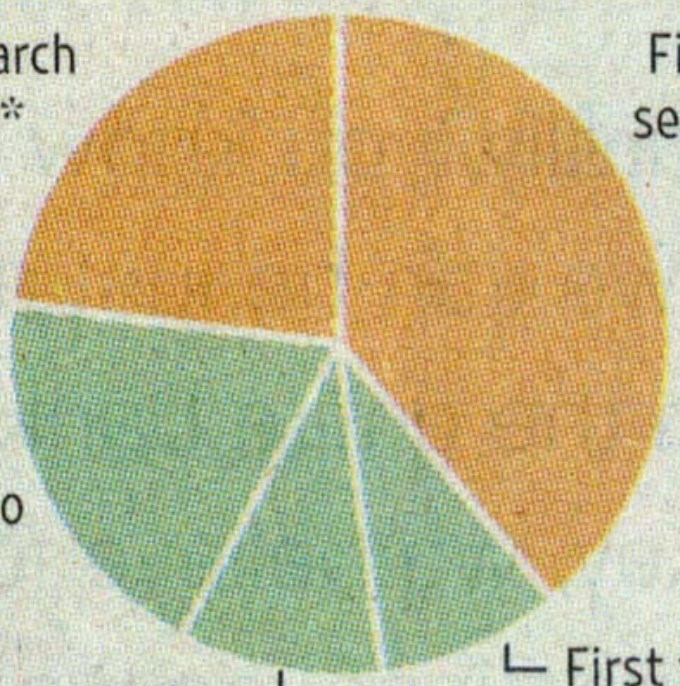
Love Data?
Empower yourself with MS BI Tools via SetFocus' Master's Program
www.SetFocus.com

Business Intelligence
Conquer DW/BI Slowdown. Get Faster Queries & Performance - Learn How.
www.Sybase.com

The Wall Street Journal, April 13, 2007

Take Your Pick

Amount of Internet search results that Web surfers typically scan before selecting one



*Top results without reading through the whole page

Note: Sample size is 2,369 people

Sources: JupiterResearch; iProspect

Sign in

Web Images Video News Maps more »

Google

business intelligence

Search

Advanced Search
Preferences

Web

Results **1** - **10** of about **122,000,000** for **business intelligence**. (**0.10** seconds)

SAS **Business Intelligence** Sponsored Link
www.SAS.com Get Better Answers Faster w/ SAS' Award-winning BI Software. Get Info

Business intelligence - Wikipedia, the free encyclopedia
Business intelligence (**BI**) is a **business** management term which refers to applications and technologies which are used to gather, provide access to, ...
en.wikipedia.org/wiki/**Business_intelligence** - 43k - Cached - Similar pages

Business Intelligence .com :: The Resource for **Business Intelligence**
The **Business Intelligence** resource for **business** and technical professionals covering a wide range of topics including Performance Management, Data Warehouse ...
www.**businessintelligence**.com/ - 74k - Apr 15, 2007 - Cached - Similar pages

Business Intelligence and Performance Management Software ...
Business intelligence and **business** performance management software. Reporting, analytics software, budgeting software, balanced scorecard software, ...
Stock quote for COGN
www.cognos.com/ - 32k - Cached - Similar pages

Oracle **Business Intelligence** Solutions
The First Comprehensive, Cost-Effective BI Solution Only Oracle delivers a complete, pre-integrated technology foundation to reduce the cost and complexity ...
www.oracle.com/solutions/ **business_intelligence**/index.html - 55k - Cached - Similar pages

Business Intelligence - Management Best Practice Reports
Business Intelligence: Providers of independent reports containing best practice advice, proprietary research findings and case studies for senior managers ...
www.**business-intelligence**.co.uk/ - 18k - Cached - Similar pages

Intelligent Enterprise: Better Insight for Business Decisions

Sponsored Links

SQL Database Management
Enterprise Data Mgmt Solutions From Dell™. Find Out More Here
www.dell.com

Business Intelligence
See what **business intelligence** can do for you (free interactive demo).
www.InformationBuilders.com

MCITP: BI Cert Boot Camp
9-Day MCITP Certification Boot Camp **Business Intelligence** All Inclusive
www.mcseclasses.com

Business Intelligence
Improve information integrity with real-time data integration software
www.DataMirror.com

Love Data?
Empower yourself with MS BI Tools via SetFocus' Master's Program
www.SetFocus.com

Business Intelligence
Conquer DW/BI Slowdown. Get Faster Queries & Performance - Learn How.
www.Sybase.com

Sign in

Google

Web Images Video News Maps more »

business intelligence

Search

Advanced Search
Preferences

Web

Results **1** - **10** of about **122,000,000** for **business intelligence**. (**0.10** seconds)

SAS **Business Intelligence** Sponsored Link
www.SAS.com Get Better Answers Faster w/ SAS' Award-winning BI Software. Get Info

Business intelligence - Wikipedia, the free encyclopedia
Business intelligence (**BI**) is a **business** management term which refers to applications and technologies which are used to gather, provide access to, ...
en.wikipedia.org/wiki/**Business_intelligence** - 43k - Cached - Similar pages

Business Intelligence .com :: The Resource for **Business Intelligence**
The **Business Intelligence** resource for **business** and technical professionals covering a wide range of topics including Performance Management, Data Warehouse ...
www.**businessintelligence**.com/ - 74k - Apr 15, 2007 - Cached - Similar pages

Business Intelligence and Performance Management Software ...
Business intelligence and **business** performance management software. Reporting, analytics software, budgeting software, balanced scorecard software, ...
Stock quote for COGN
www.cognos.com/ - 32k - Cached - Similar pages

Oracle **Business Intelligence** Solutions
The First Comprehensive, Cost-Effective BI Solution Only Oracle delivers a complete, pre-integrated technology foundation to reduce the cost and complexity ...
www.oracle.com/solutions/ **business_intelligence**/index.html - 55k - Cached - Similar pages

Business Intelligence - Management Best Practice Reports
Business Intelligence: Providers of independent reports containing best practice advice, proprietary research findings and case studies for senior managers ...
www.**business-intelligence**.co.uk/ - 18k - Cached - Similar pages

Intelligent Enterprise: Better Insight for **Business** Decisions

Sponsored Links

SQL Database Management
Enterprise Data Mgmt Solutions From Dell™. Find Out More Here
www.dell.com

Business Intelligence
See what **business intelligence** can do for you (free interactive demo).
www.InformationBuilders.com

MCITP: BI Cert Boot Camp
9-Day MCITP Certification Boot Camp **Business Intelligence** All Inclusive
www.mcseclasses.com

Business Intelligence
Improve information integrity with real-time data integration software
www.DataMirror.com

Love Data?
Empower yourself with MS BI Tools via SetFocus' Master's Program
www.SetFocus.com

Business Intelligence
Conquer DW/BI Slowdown. Get Faster Queries & Performance - Learn How.
www.Sybase.com


Newsweek
March 29
$3.95
newsweek.msnbc.com
The Next Frontiers
The New Age of
Google
The Search Giant Has Changed Our Lives. Can Anybody Catch These Guys? By Steven Levy
PLUS: The Future of Digital Voting
Google founders Larry Page and Sergey Brin

Google's PageRank

(Lawrence Page & Sergey Brin 1998)

The Google Goals

- Create a PageRank $r(P)$ that is not query dependent
  - ▷ Off-line calculations — No query time computation
- Let the Web vote with in-links
  - ▷ But not by simple link counts
    - — One link to P from Yahoo! is important
    - — Many links to P from me is not
- Share The Vote
  - ▷ Yahoo! casts many "votes"
    - — value of vote from $Yahoo$! is diluted
  - ▷ If Yahoo! "votes" for n pages
    - — Then P receives only $r(Y)/n$ credit from Y

Google's PageRank

(Lawrence Page & Sergey Brin 1998)

The Google Goals

- Create a PageRank $r(P)$ that is not query dependent
  - ▷ Off-line calculations — No query time computation
- Let the Web vote with in-links
  - ▷ But not by simple link counts
    - — One link to P from Yahoo! is important
    - — Many links to P from me is not
- Share The Vote
  - ▷ Yahoo! casts many "votes"
    - — value of vote from $Yahoo$! is diluted
  - ▷ If Yahoo! "votes" for n pages
    - — Then P receives only $r(Y)/n$ credit from Y

Google's PageRank

(Lawrence Page & Sergey Brin 1998)

The Google Goals

- Create a PageRank $r(P)$ that is not query dependent
  - ▷ Off-line calculations — No query time computation
- Let the Web vote with in-links
  - ▷ But not by simple link counts
    - — One link to P from Yahoo! is important
    - — Many links to P from me is not
- Share The Vote
  - ▷ Yahoo! casts many "votes"
    - — value of vote from $Yahoo$! is diluted
  - ▷ If Yahoo! "votes" for n pages
    - — Then P receives only $r(Y)/n$ credit from Y

Google's PageRank

(Lawrence Page & Sergey Brin 1998)

The Google Goals

- Create a PageRank $r(P)$ that is not query dependent
  - ▷ Off-line calculations — No query time computation
- Let the Web vote with in-links
  - ▷ But not by simple link counts
    - — One link to P from Yahoo! is important
    - — Many links to P from me is not
- Share The Vote
  - ▷ Yahoo! casts many "votes"
    - — value of vote from $Yahoo$! is diluted
  - ▷ If Yahoo! "votes" for n pages
    - — Then P receives only $r(Y)/n$ credit from Y

PageRank

The Definition

$$r(P) = \sum_{P \in \mathcal{B}_P} \frac{r(P)}{|P|}$$

$\mathcal{B}_P$ = {all pages pointing to P}

$|P|$ = number of out links from P

PageRank

The Definition

$$r(P) = \sum_{P \in \mathcal{B}_P} \frac{r(P)}{|P|}$$

$\mathcal{B}_P$ = {all pages pointing to P}

$|P|$ = number of out links from P

Successive Refinement

Start with $r_0(P_i) = 1/n$ for all pages $P_1, P_2, \ldots, P_n$

PageRank

The Definition

$$r(P) = \sum_{P \in \mathcal{B}_P} \frac{r(P)}{|P|}$$

$\mathcal{B}_P$ = {all pages pointing to P}

$|P|$ = number of out links from P

Successive Refinement

Start with $r_0(P_i) = 1/n$ for all pages $P_1, P_2, \ldots, P_n$

Iteratively refine rankings for each page

$$r_1(P_i) = \sum_{P \in \mathcal{B}_{P_i}} \frac{r_0(P)}{|P|}$$

PageRank

The Definition

$$r(P) = \sum_{P \in \mathcal{B}_P} \frac{r(P)}{|P|}$$

$\mathcal{B}_P$ = {all pages pointing to P}

$|P|$ = number of out links from P

Successive Refinement

Start with $r_0(P_i) = 1/n$ for all pages $P_1, P_2, \ldots, P_n$

Iteratively refine rankings for each page

$$r_1(P_i) = \sum_{P \in \mathcal{B}_{P_i}} \frac{r_0(P)}{|P|}$$

$$r_2(P_i) = \sum_{P \in \mathcal{B}_{P_i}} \frac{r_1(P)}{|P|}$$

PageRank

The Definition

$$r(P) = \sum_{P \in \mathcal{B}_P} \frac{r(P)}{|P|}$$

$\mathcal{B}_P$ = {all pages pointing to P}

$|P|$ = number of out links from P

Successive Refinement

Start with $r_0(P_i) = 1/n$ for all pages $P_1, P_2, \ldots, P_n$

Iteratively refine rankings for each page

$$r_1(P_i) = \sum_{P \in \mathcal{B}_{P_i}} \frac{r_0(P)}{|P|}$$

$$r_2(P_i) = \sum_{P \in \mathcal{B}_{P_i}} \frac{r_1(P)}{|P|}$$

$$\ddots$$

$$r_{j+1}(P_i) = \sum_{P \in \mathcal{B}_{P_i}} \frac{r_j(P)}{|P|}$$

In Matrix Notation

After Step k

- $\boldsymbol{\pi}_k^T = [r_k(P_1),\, r_k(P_2),\, \cdots,\, r_k(P_n)]$

In Matrix Notation

After Step k

- $\boldsymbol{\pi}_k^T = [r_k(P_1), r_k(P_2), \cdots, r_k(P_n)]$

- $\boldsymbol{\pi}_{k+1}^T = \boldsymbol{\pi}_k^T \mathbf{H}$ where $h_{ij} = \begin{cases} 1/|P_i| & \text{if } i \to j \\ 0 & \text{otherwise} \end{cases}$

In Matrix Notation

After Step k

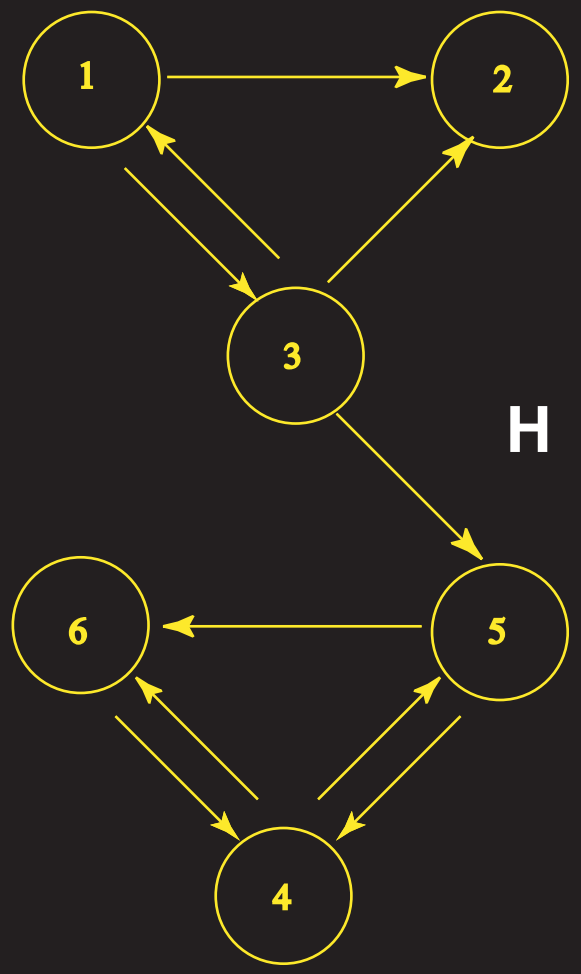
- $\boldsymbol{\pi}_k^T = [r_k(P_1), r_k(P_2), \cdots, r_k(P_n)]$

- $\boldsymbol{\pi}_{k+1}^T = \boldsymbol{\pi}_k^T \mathbf{H}$ where $h_{ij} = \begin{cases} 1/|P_i| & \text{if } i \to j \\ 0 & \text{otherwise} \end{cases}$

- PageRank vector = $\boldsymbol{\pi}^T = \lim_{k \to \infty} \boldsymbol{\pi}_k^T$ = eigenvector for $\mathbf{H}$

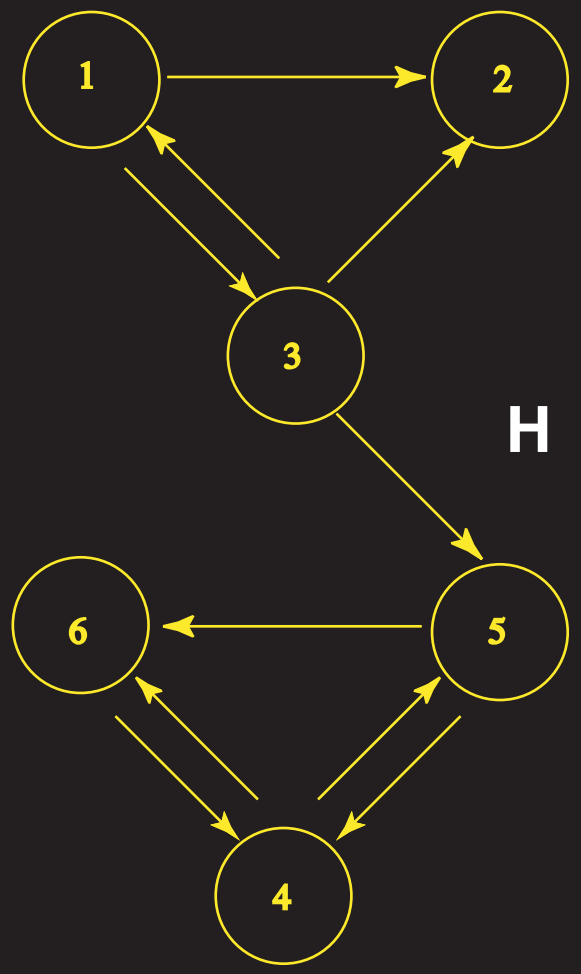
Provided that the limit exists

Tiny Web



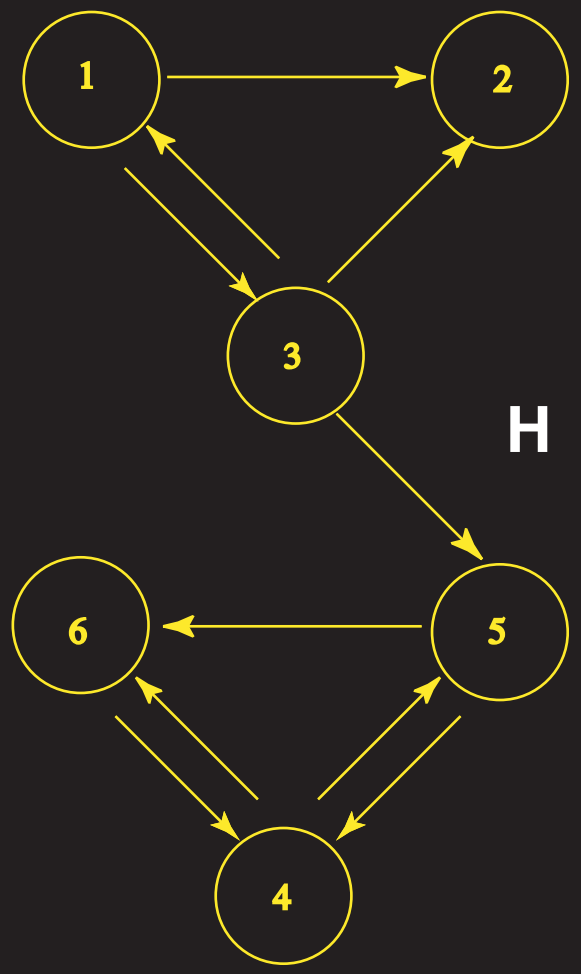
$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{pmatrix} \end{array}$$

Tiny Web



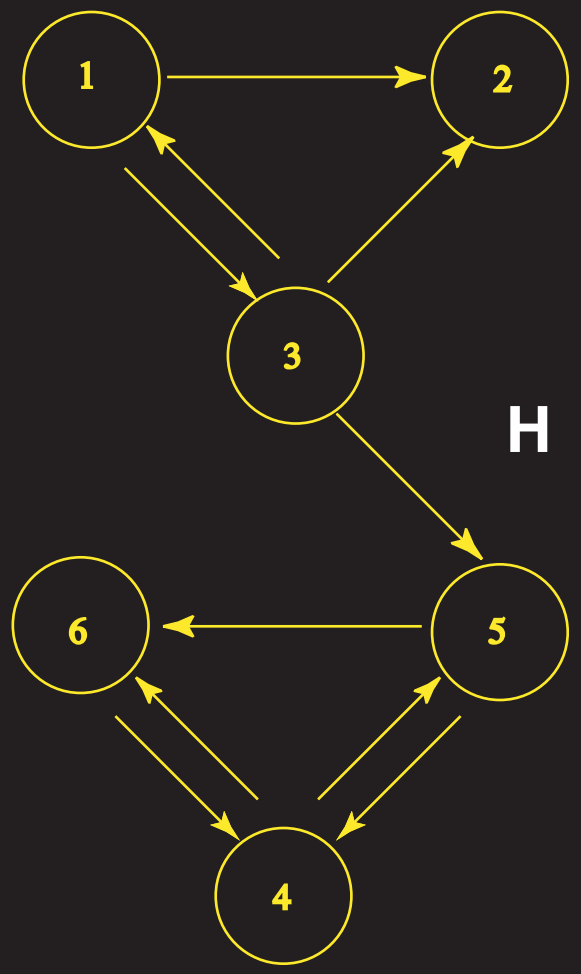
$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{pmatrix} \end{array}$$

Tiny Web



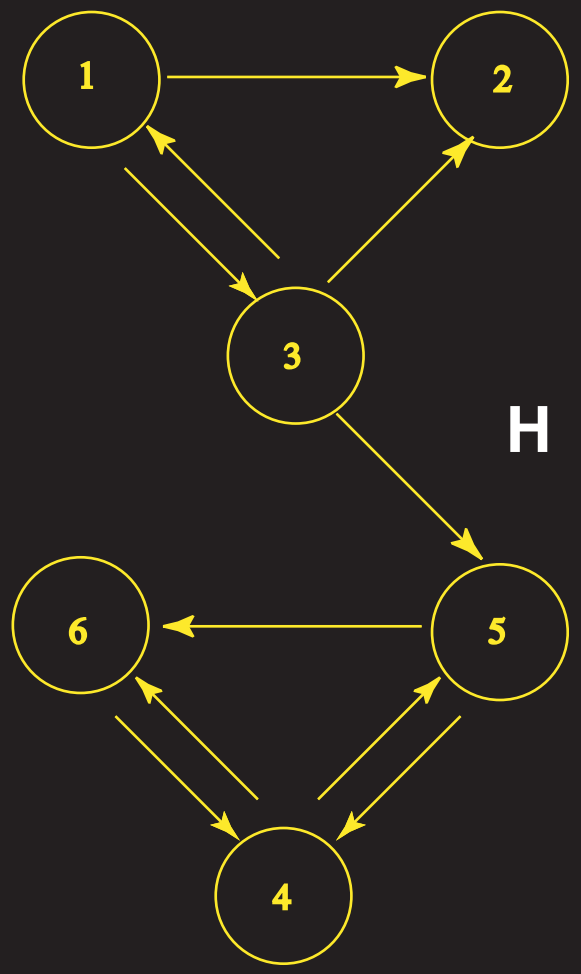
$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{pmatrix} \end{array}$$

Tiny Web



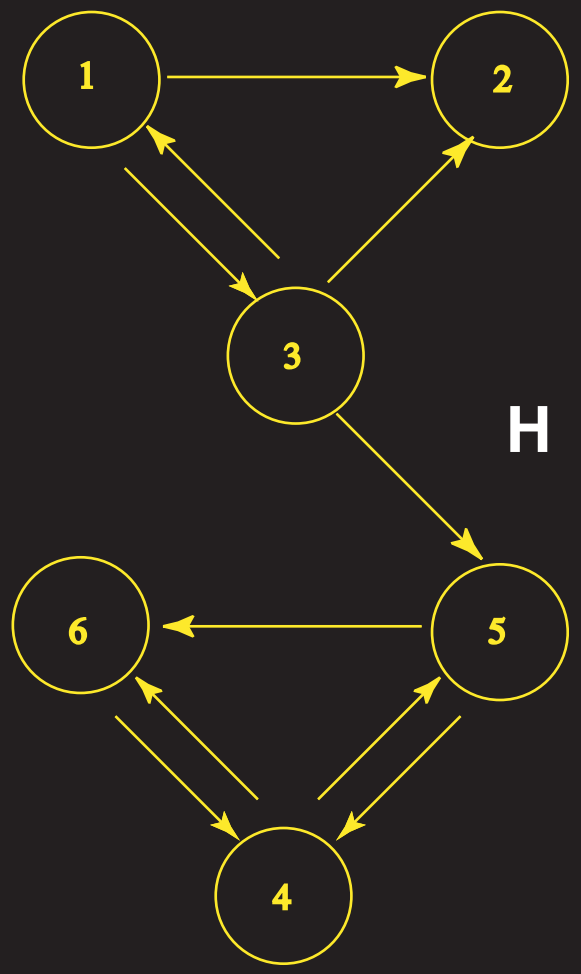
$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ & & & & & \\ & & & & & \\ & & & & & \end{pmatrix} \end{array}$$

Tiny Web



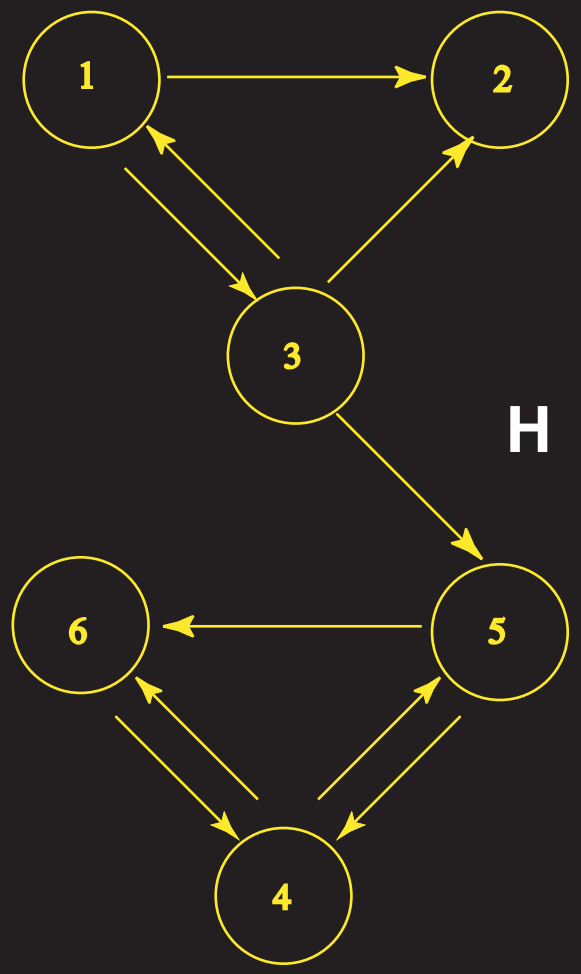
$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ & & & & & \\ & & & & & \end{pmatrix} \end{array}$$

Tiny Web



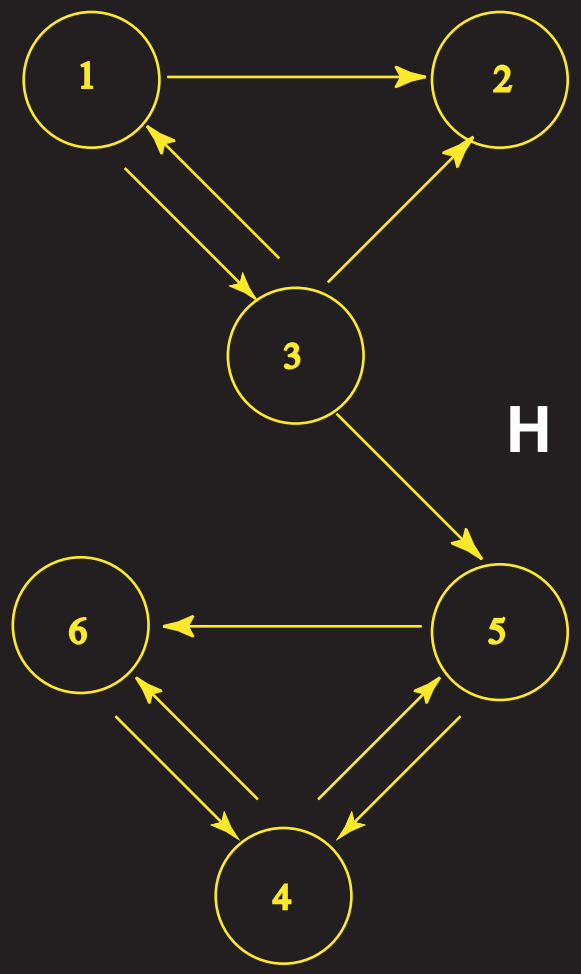
$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ & & & & & \end{pmatrix} \end{array}$$

Tiny Web



$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

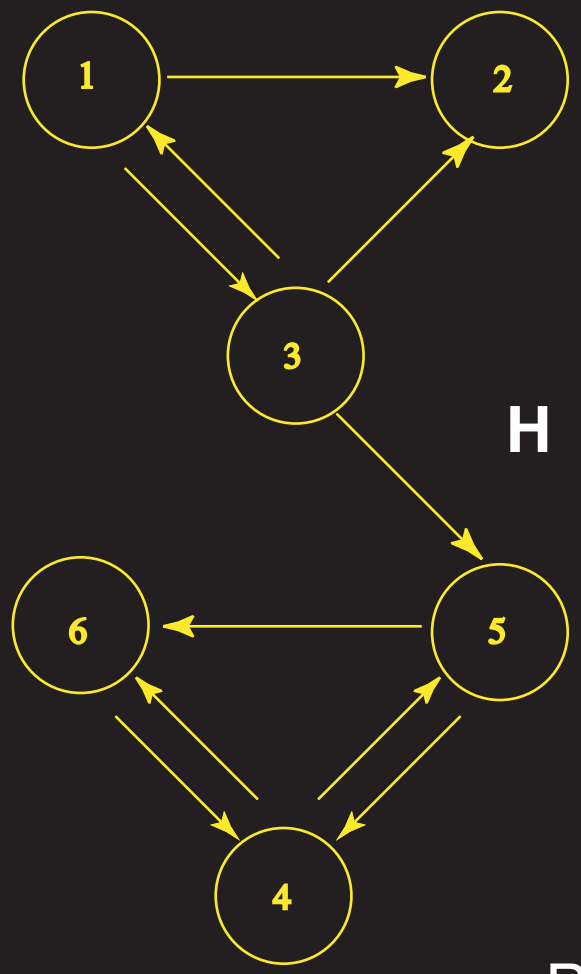
Tiny Web



$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

- A random walk on the Web Graph

Tiny Web



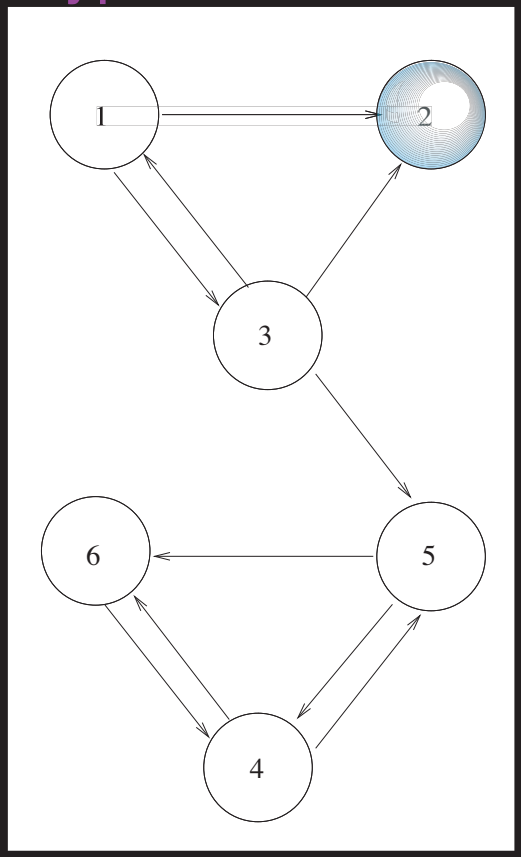
$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

- A random walk on the Web Graph
- PageRank = π_i = amount of time spent at P_i

Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

Hyperlink as vote

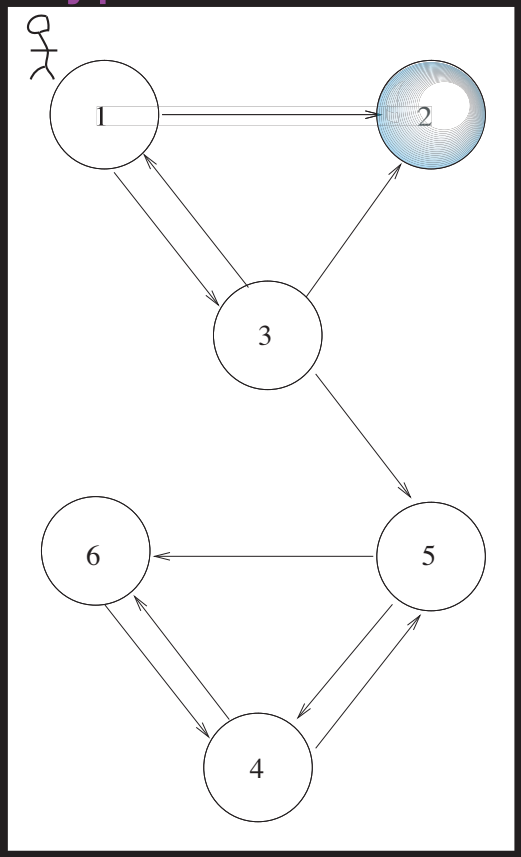


Markov chain

Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

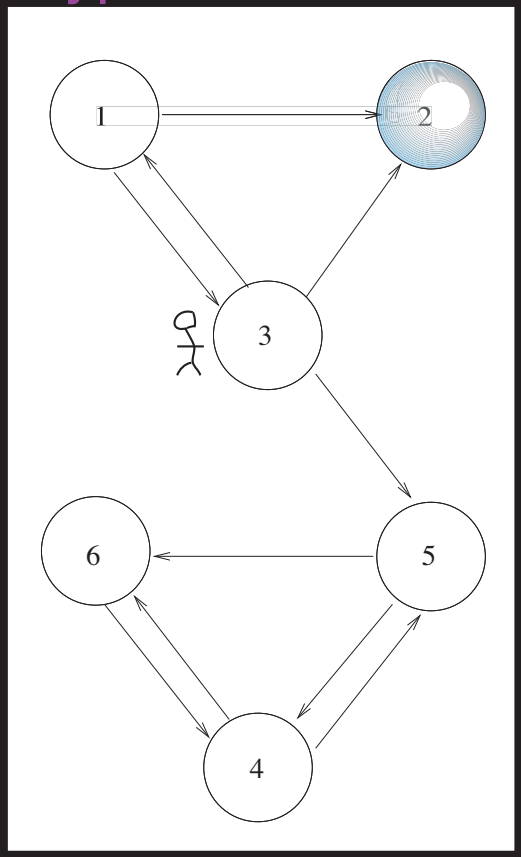
Hyperlink as vote



Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

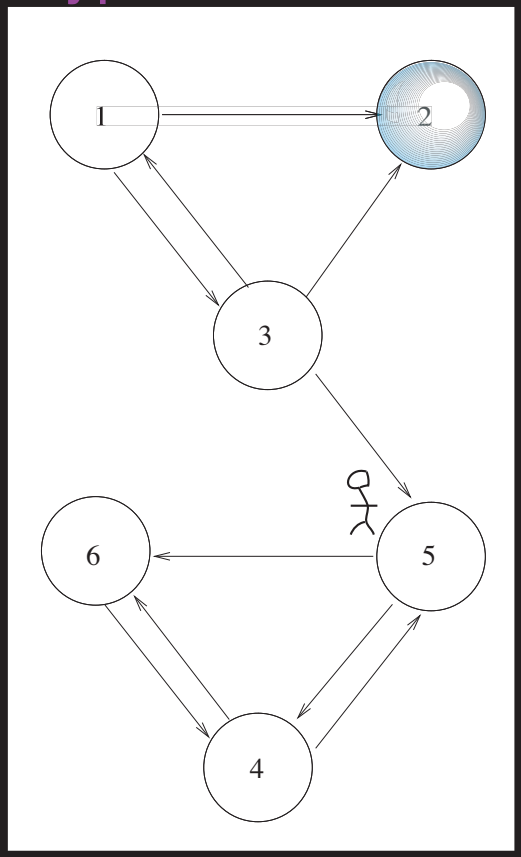
Hyperlink as vote



Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

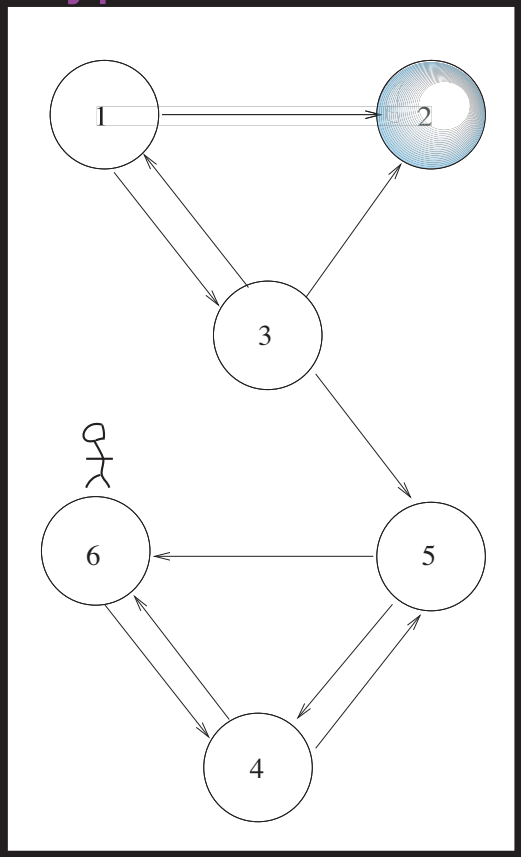
Hyperlink as vote



Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

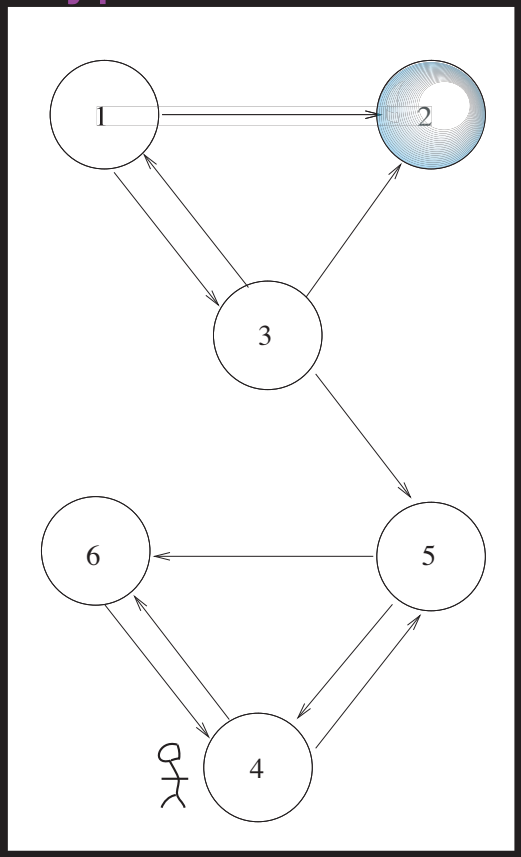
Hyperlink as vote



Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

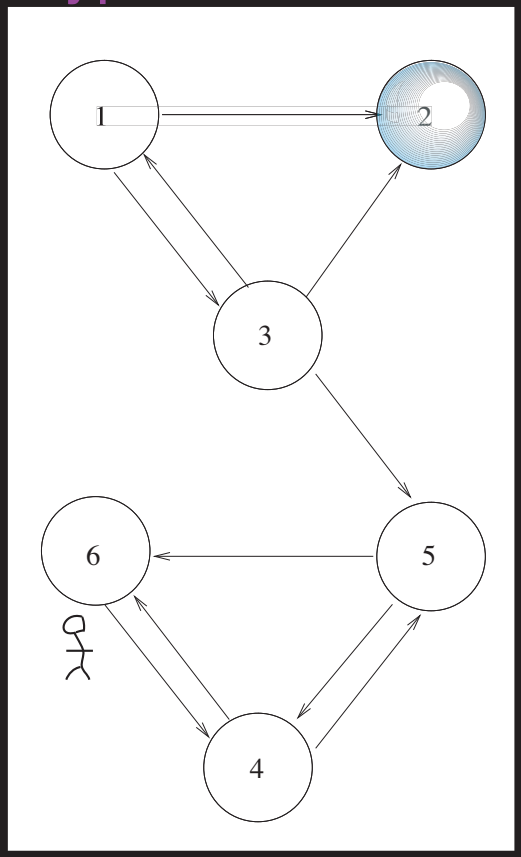
Hyperlink as vote



Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

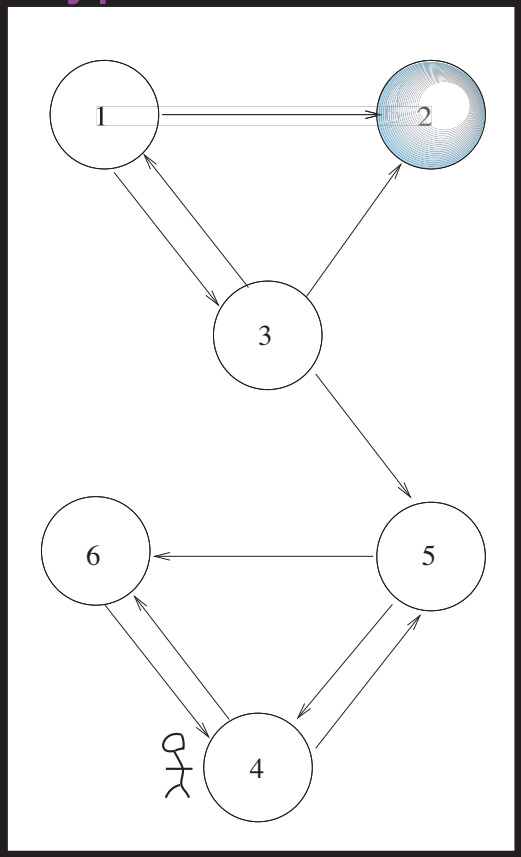
Hyperlink as vote



Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

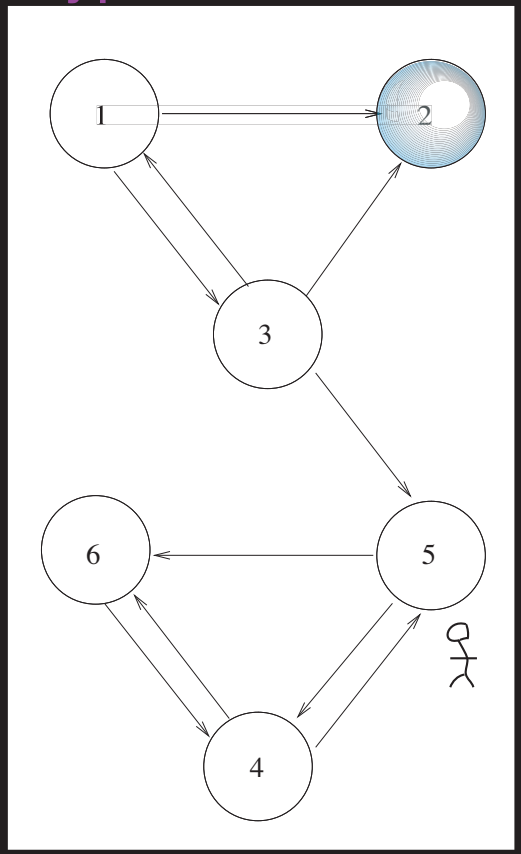
Hyperlink as vote



Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

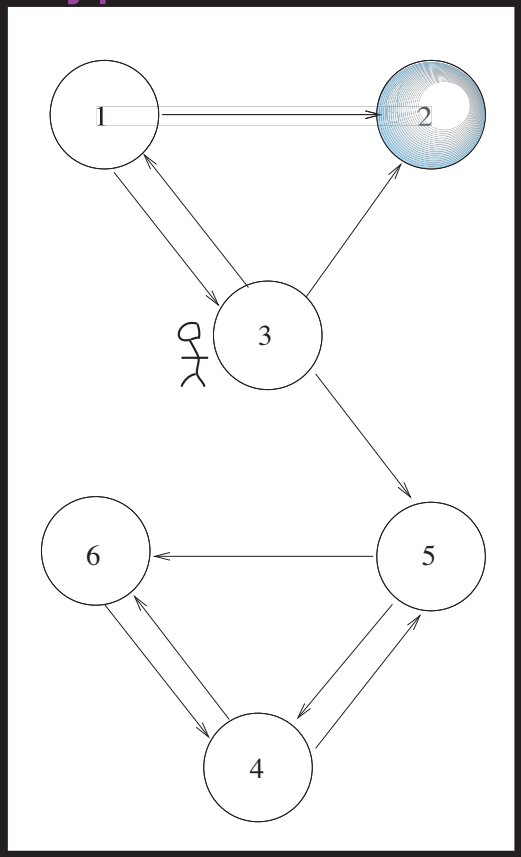
Hyperlink as vote



Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

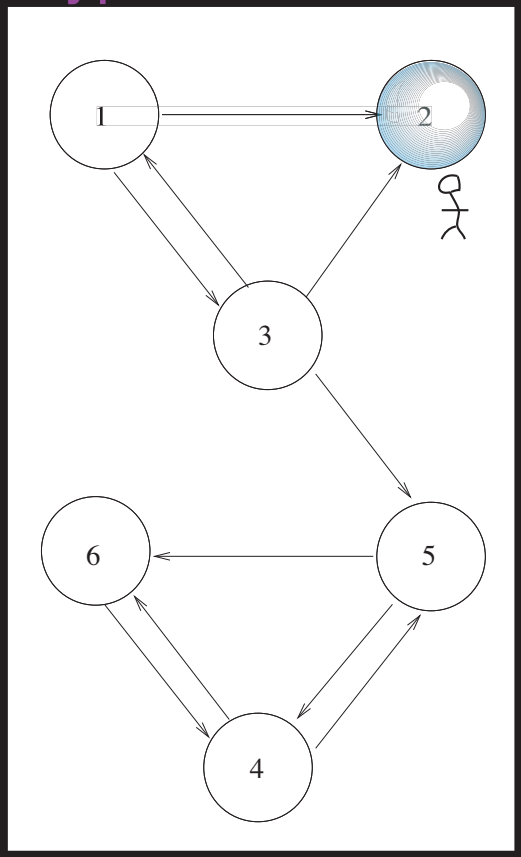
Hyperlink as vote



Ranking with a Random Surfer

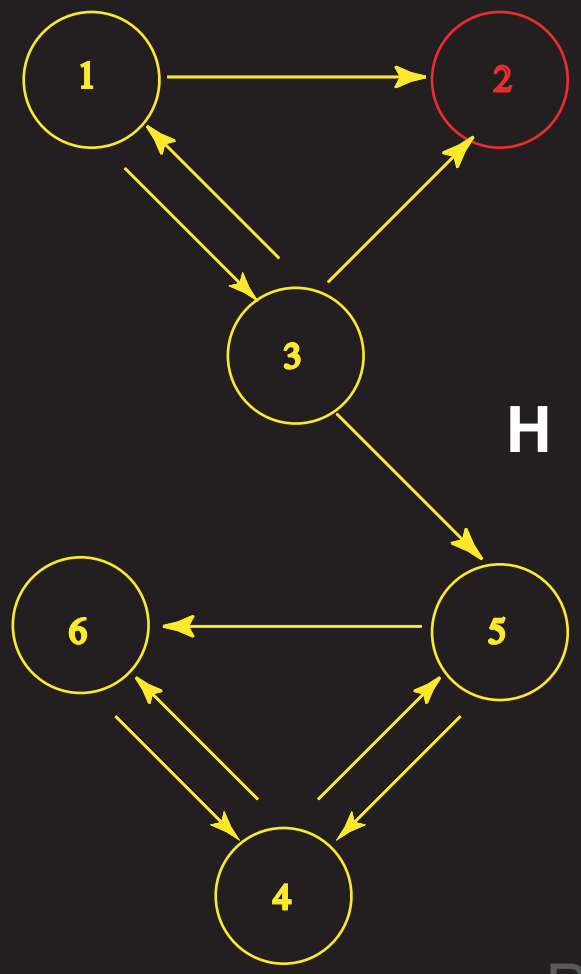
- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

Hyperlink as vote



page 2 is a dangling node

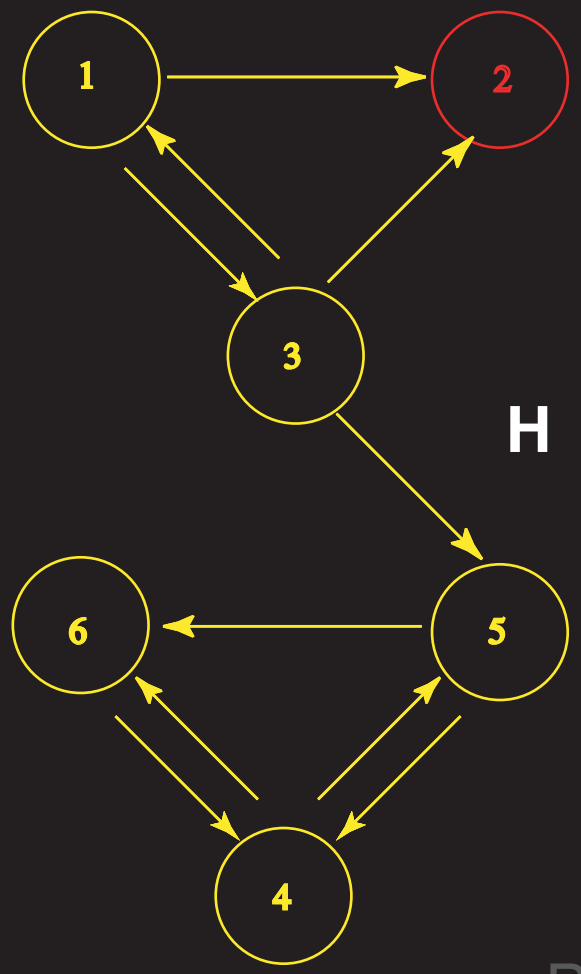
Tiny Web



$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

- A random walk on the Web Graph
- PageRank = π_i = amount of time spent at P_i
- Dead end page (nothing to click on) — a “dangling node”

Tiny Web



$$\mathbf{H} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

- ▷ A random walk on the Web Graph
- ▷ PageRank = π_i = amount of time spent at P_i
- ▷ Dead end page (nothing to click on) — a "dangling node"
- ▷ $\boldsymbol{\pi}^T = (0, 1, 0, 0, 0, 0)$ = e-vector $\Longrightarrow$ Page P_2 is a "rank sink"

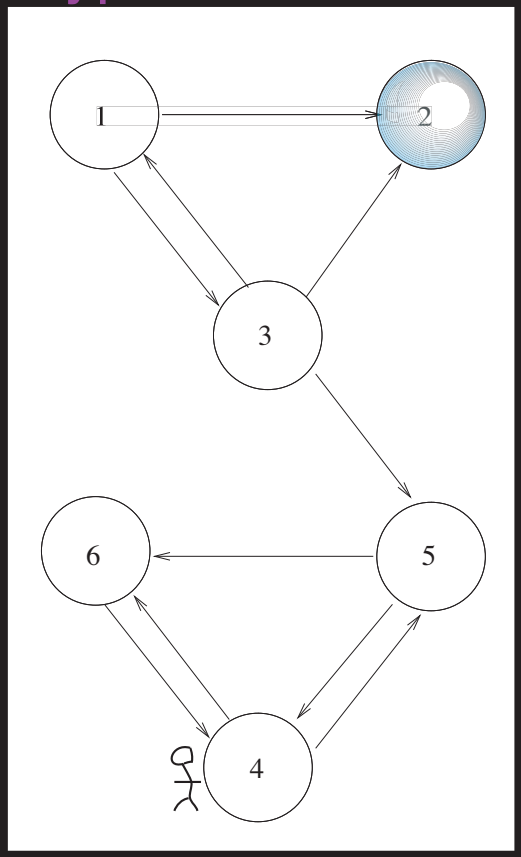
The Fix

Allow Web Surfers To Make Random Jumps

Ranking with a Random Surfer

- Rank each page corresponding to a search term by number and *quality* of votes cast for that page.

Hyperlink as vote



surfer “teleports”

The Fix

Allow Web Surfers To Make Random Jumps

- Replace zero rows with $\frac{\mathbf{e}^T}{n} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$

$$\mathbf{S} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

The Fix

Allow Web Surfers To Make Random Jumps

- Replace zero rows with $\frac{\mathbf{e}^T}{n} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$

$$\mathbf{S} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

- $\mathbf{S} = \mathbf{H} + \frac{\mathbf{a}\,\mathbf{e}^T}{6}$ is now row stochastic $\implies \rho(\mathbf{S}) = 1$

The Fix

Allow Web Surfers To Make Random Jumps

- Replace zero rows with $\frac{\mathbf{e}^T}{n} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)$

$$\mathbf{S} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

- $\mathbf{S} = \mathbf{H} + \frac{\mathbf{a}\,\mathbf{e}^T}{6}$ is now row stochastic $\Longrightarrow$ $\rho(\mathbf{S}) = 1$
- Perron says $\exists$ $\boldsymbol{\pi}^T \geq 0$ s.t. $\boldsymbol{\pi}^T = \boldsymbol{\pi}^T\mathbf{S}$ with $\sum_i \pi_i = 1$

Nasty Problem

The Web Is Not Strongly Connected

Nasty Problem

The Web Is Not Strongly Connected

- S is reducible

$$
\mathbf{S} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \end{array}
\begin{array}{c}
\begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\
\left(\begin{array}{ccc|ccc}
0 & 1/2 & 1/2 & 0 & 0 & 0 \\
1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\
1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\
\hline
0 & 0 & 0 & 0 & 1/2 & 1/2 \\
0 & 0 & 0 & 1/2 & 0 & 1/2 \\
0 & 0 & 0 & 1 & 0 & 0
\end{array}\right)
\end{array}
$$

Nasty Problem

The Web Is Not Strongly Connected

∴ S is reducible

$$\mathbf{S} = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ \hline P_4 \\ P_5 \\ P_6 \end{array} \begin{array}{c} \begin{array}{cccccc} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \end{array} \\ \left(\begin{array}{ccc|ccc} 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/3 & 1/3 & 0 & 0 & 1/3 & 0 \\ \hline 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{array}\right) \end{array}$$

- Reducible $\Longrightarrow$ PageRank vector is not well defined
- Frobenius says $\mathbf{S}$ needs to be *irreducible* to ensure a unique $\pi^T > 0$ s.t. $\pi^T = \pi^T \mathbf{S}$ with $\sum_i \pi_i = 1$

Irreducibility Is Not Enough

Could Get Trapped Into A Cycle $(P_i \rightarrow P_j \rightarrow P_i)$

Irreducibility Is Not Enough

Could Get Trapped Into A Cycle $(P_i \to P_j \to P_i)$

- The powers $\mathbf{S}^k$ fail to converge

Irreducibility Is Not Enough

Could Get Trapped Into A Cycle $(P_i \rightarrow P_j \rightarrow P_i)$

- The powers $\mathbf{S}^k$ fail to converge
- $\boldsymbol{\pi}_{k+1}^T = \boldsymbol{\pi}_k^T \mathbf{S}$ fails to convergence

Irreducibility Is Not Enough

Could Get Trapped Into A Cycle $(P_i \to P_j \to P_i)$

- The powers $\mathbf{S}^k$ fail to converge
- $\pi_{k+1}^T = \pi_k^T \mathbf{S}$ fails to convergence

Convergence Requirement

- Perron–Frobenius requires $\mathbf{S}$ to be primitive

Irreducibility Is Not Enough

Could Get Trapped Into A Cycle $(P_i \to P_j \to P_i)$

- The powers $\mathbf{S}^k$ fail to converge
- $\pi_{k+1}^T = \pi_k^T \mathbf{S}$ fails to convergence

Convergence Requirement

- Perron–Frobenius requires $\mathbf{S}$ to be primitive
- No eigenvalues other than $\lambda = 1$ on unit circle

Irreducibility Is Not Enough

Could Get Trapped Into A Cycle $(P_i \to P_j \to P_i)$

- The powers $\mathbf{S}^k$ fail to converge
- $\pi_{k+1}^T = \pi_k^T \mathbf{S}$ fails to convergence

Convergence Requirement

- Perron–Frobenius requires $\mathbf{S}$ to be primitive
- No eigenvalues other than $\lambda = 1$ on unit circle
- Frobenius proved $\mathbf{S}$ is primitive $\iff \mathbf{S}^k > 0$ for some k

The Google Fix

Allow A Random Jump From Any Page

- $\mathbf{G} = \alpha\mathbf{S} + (1-\alpha)\mathbf{E} > 0, \quad \mathbf{E} = \mathbf{e}\mathbf{e}^T/n, \quad 0 < \alpha < 1$

The Google Fix

Allow A Random Jump From Any Page

- $\mathbf{G} = \alpha\mathbf{S} + (1-\alpha)\mathbf{E} > 0, \quad \mathbf{E} = \mathbf{e}\mathbf{e}^T/n, \quad 0 < \alpha < 1$
- $\mathbf{G} = \alpha\mathbf{H} + \mathbf{u}\mathbf{v}^T > 0 \qquad \mathbf{u} = \alpha\mathbf{a} + (1-\alpha)\mathbf{e}, \quad \mathbf{v}^T = \mathbf{e}^T/n$

The Google Fix

Allow A Random Jump From Any Page

- $\mathbf{G} = \alpha\mathbf{S} + (1-\alpha)\mathbf{E} > 0, \quad \mathbf{E} = \mathbf{e}\mathbf{e}^T/n, \quad 0 < \alpha < 1$
- $\mathbf{G} = \alpha\mathbf{H} + \mathbf{u}\mathbf{v}^T > 0 \qquad \mathbf{u} = \alpha\mathbf{a} + (1-\alpha)\mathbf{e}, \quad \mathbf{v}^T = \mathbf{e}^T/n$
- PageRank vector $\qquad \pi^T$ = left-hand Perron vector of $\mathbf{G}$

Ranking with a Random Surfer

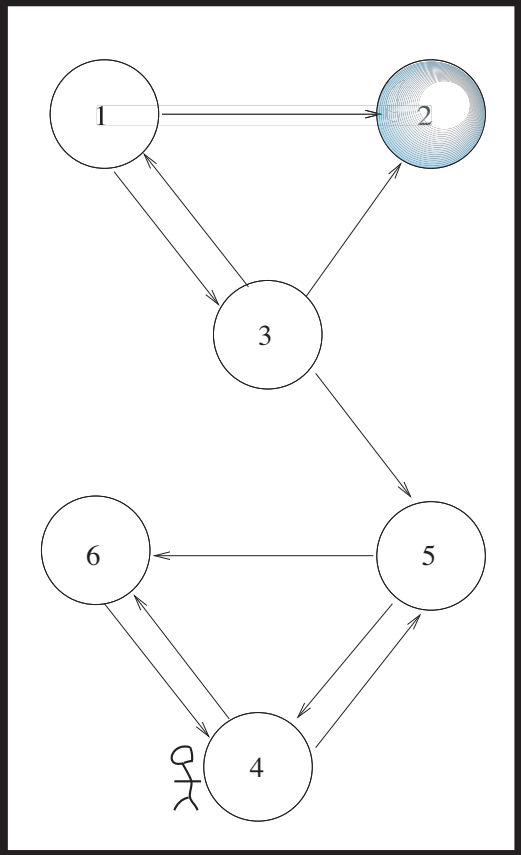
- If a page is “important,” it gets lots of votes from other important pages, which means the random surfer visits it often.
- Simply count the number of times, or $proportion\ of\ time$, the surfer spends on each page to create ranking of webpages.

Ranking with a Random Surfer

- If a page is "important," it gets lots of votes from other important pages, which means the random surfer visits it often.
- Simply count the number of times, or *proportion of time*, the surfer spends on each page to create ranking of webpages.

Proportion of Time

Page 1 = .04
Page 2 = .05
Page 3 = .04
Page 4 = .38
Page 5 = .20
Page 6 = .29



Ranked List of Pages

Page 4
Page 6
Page 5
Page 2
Page 1
Page 3

The Google Fix

Allow A Random Jump From Any Page

- $\mathbf{G} = \alpha\mathbf{S} + (1-\alpha)\mathbf{E} > 0, \quad \mathbf{E} = \mathbf{e}\mathbf{e}^T/n, \quad 0 < \alpha < 1$
- $\mathbf{G} = \alpha\mathbf{H} + \mathbf{u}\mathbf{v}^T > 0 \qquad \mathbf{u} = \alpha\mathbf{a} + (1-\alpha)\mathbf{e}, \quad \mathbf{v}^T = \mathbf{e}^T/n$
- PageRank vector $\qquad \pi^T$ = left-hand Perron vector of $\mathbf{G}$

Some Happy Accidents

- $\mathbf{x}^T\mathbf{G} = \alpha\mathbf{x}^T\mathbf{H} + \beta\mathbf{v}^T$ Sparse computations with the original link structure

The Google Fix

Allow A Random Jump From Any Page

- $\mathbf{G} = \alpha\mathbf{S} + (1-\alpha)\mathbf{E} > 0, \quad \mathbf{E} = \mathbf{e}\mathbf{e}^T/n, \quad 0 < \alpha < 1$
- $\mathbf{G} = \alpha\mathbf{H} + \mathbf{u}\mathbf{v}^T > 0 \qquad \mathbf{u} = \alpha\mathbf{a} + (1-\alpha)\mathbf{e}, \quad \mathbf{v}^T = \mathbf{e}^T/n$
- PageRank vector $\qquad \pi^T$ = left-hand Perron vector of $\mathbf{G}$

Some Happy Accidents

- $\mathbf{x}^T\mathbf{G} = \alpha\mathbf{x}^T\mathbf{H} + \beta\mathbf{v}^T$ Sparse computations with the original link structure
- $\lambda_2(\mathbf{G}) = \alpha$ Convergence rate controllable by Google engineers

The Google Fix

Allow A Random Jump From Any Page

- $\mathbf{G} = \alpha\mathbf{S} + (1-\alpha)\mathbf{E} > 0, \quad \mathbf{E} = \mathbf{e}\mathbf{e}^T/n, \quad 0 < \alpha < 1$
- $\mathbf{G} = \alpha\mathbf{H} + \mathbf{u}\mathbf{v}^T > 0 \qquad \mathbf{u} = \alpha\mathbf{a} + (1-\alpha)\mathbf{e}, \quad \mathbf{v}^T = \mathbf{e}^T/n$
- PageRank vector $\qquad \pi^T$ = left-hand Perron vector of $\mathbf{G}$

Some Happy Accidents

- $\mathbf{x}^T\mathbf{G} = \alpha\mathbf{x}^T\mathbf{H} + \beta\mathbf{v}^T$ Sparse computations with the original link structure
- $\lambda_2(\mathbf{G}) = \alpha$ Convergence rate controllable by Google engineers
- $\mathbf{v}^T$ can be any positive probability vector in $\mathbf{G} = \alpha\mathbf{H} + \mathbf{u}\mathbf{v}^T$

The Google Fix

Allow A Random Jump From Any Page

- $\mathbf{G} = \alpha\mathbf{S} + (1-\alpha)\mathbf{E} > 0, \quad \mathbf{E} = \mathbf{e}\mathbf{e}^T/n, \quad 0 < \alpha < 1$
- $\mathbf{G} = \alpha\mathbf{H} + \mathbf{u}\mathbf{v}^T > 0 \qquad \mathbf{u} = \alpha\mathbf{a} + (1-\alpha)\mathbf{e}, \quad \mathbf{v}^T = \mathbf{e}^T/n$
- PageRank vector $\qquad \pi^T$ = left-hand Perron vector of $\mathbf{G}$

Some Happy Accidents

- $\mathbf{x}^T\mathbf{G} = \alpha\mathbf{x}^T\mathbf{H} + \beta\mathbf{v}^T$ Sparse computations with the original link structure
- $\lambda_2(\mathbf{G}) = \alpha$ Convergence rate controllable by Google engineers
- $\mathbf{v}^T$ can be any positive probability vector in $\mathbf{G} = \alpha\mathbf{H} + \mathbf{u}\mathbf{v}^T$
- The choice of $\mathbf{v}^T$ allows for personalization

Google matrix G > 0

Perron-Frobenius guarantees

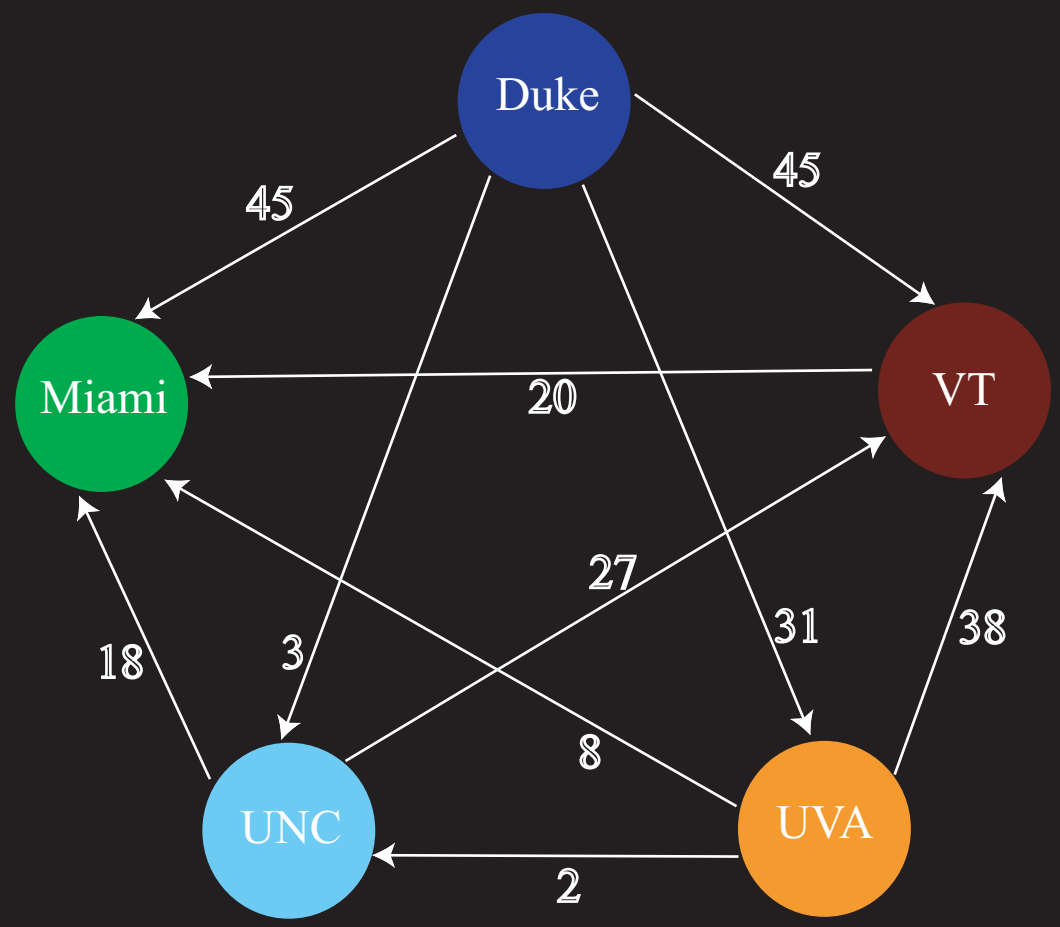
- existence of π
- uniqueness of π
- convergence of algorithm
- rate of convergence: $\lambda_2(\mathbf{G}) = \alpha$

Ranking Sports Teams

PageRank applied to Sports

(joint work with Carl Meyer)

- webpages vote with hyperlinks
- losing teams vote with point differentials

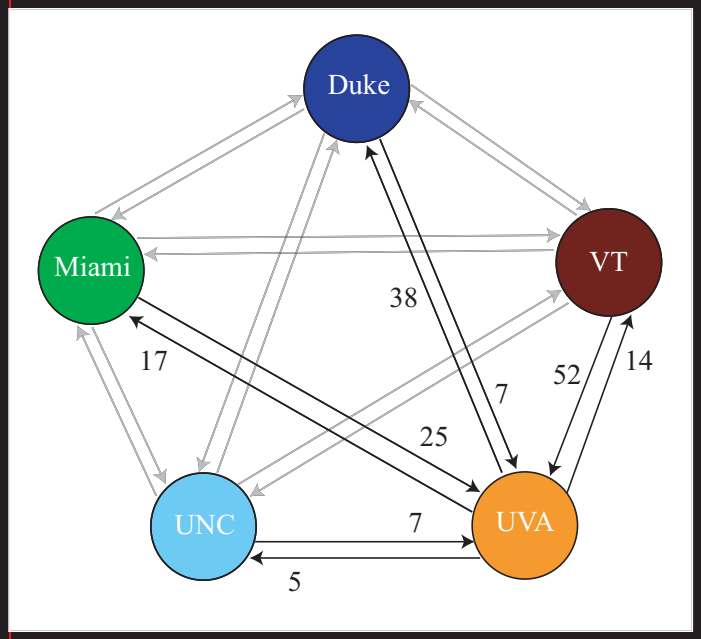


- not as successful at ranking and predicting winners as mHITS

mHITS

each team i gets both offensive rating $\mathbf{o}_i$ and defensive rating $\mathbf{d}_i$

- **mHITS Thesis:** A team is a good defensive team (i.e., deserves a high defensive rating d_j) when it holds its opponents (particularly strong offensive teams) to low scores. A team is a good offensive team (i.e., deserves a high offensive rating o_i) when it scores many points against its opponents (particularly opponents with high defensive ratings).



Graph

$$\mathbf{P} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 7 & 21 & 7 & 0 \\ 52 & 0 & 34 & 25 & 27 \\ 24 & 16 & 0 & 7 & 3 \\ 38 & 17 & 5 & 0 & 14 \\ 45 & 7 & 30 & 52 & 0 \end{pmatrix} \end{array}$$

Point Matrix $\mathbf{P} \geq 0$

mHITS Equations

Summation Notation

$$d_j = \sum_{i \in I_j} p_{ij} \frac{1}{o_i} \qquad \text{and} \qquad o_i = \sum_{j \in L_i} p_{ij} \frac{1}{d_j}$$

Matrix Notation: iterative procedure

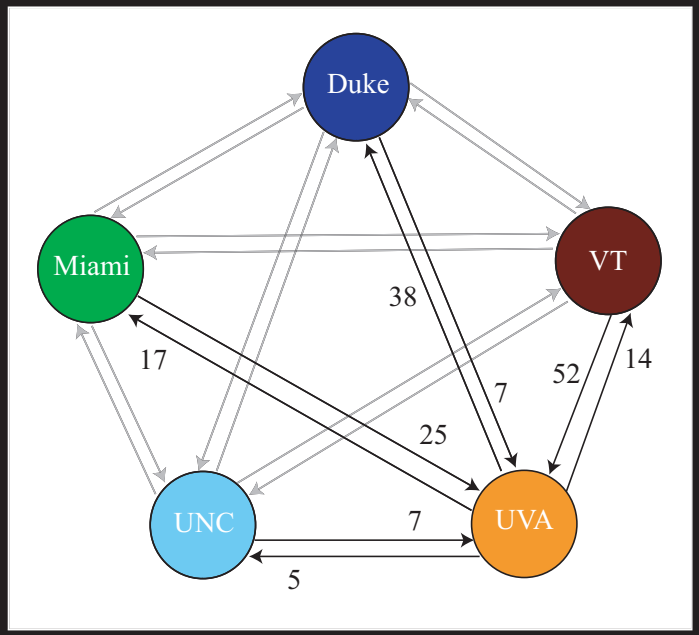
$$\mathbf{d}^{(k)} = \mathbf{P}^T \frac{1}{\mathbf{o}^{(k)}} \qquad \text{and} \qquad \mathbf{o}^{(k)} = \mathbf{P} \frac{1}{\mathbf{d}^{(k-1)}}$$

- related to the Sinkhorn-Knopp algorithm for matrix balancing
  (uses successive row and column scaling to transform $\mathbf{P} \geq 0$ into doubly stochastic matrix $\mathbf{S}$)
- P. Knight uses Sinkhorn-Knopp algorithm to rank webpages

mHITS Results: tiny NCAA

(data from Luke Ingram)

Team	Off. Rating o	Off. Rank	Def. Rating d	Def. Rank
Duke	.4	5th	140	5th
Miami	1.8	1st	67	2nd
UNC	.6	4th	97	4th
UVA	1.0	3rd	80	3rd
VT	1.4	2nd	34	1st



	r = o/d Rating	r = o/d Rank
Duke	0.003	5th
Miami	0.027	2nd
UNC	0.006	4th
UVA	0.012	3rd
VT	0.041	1st

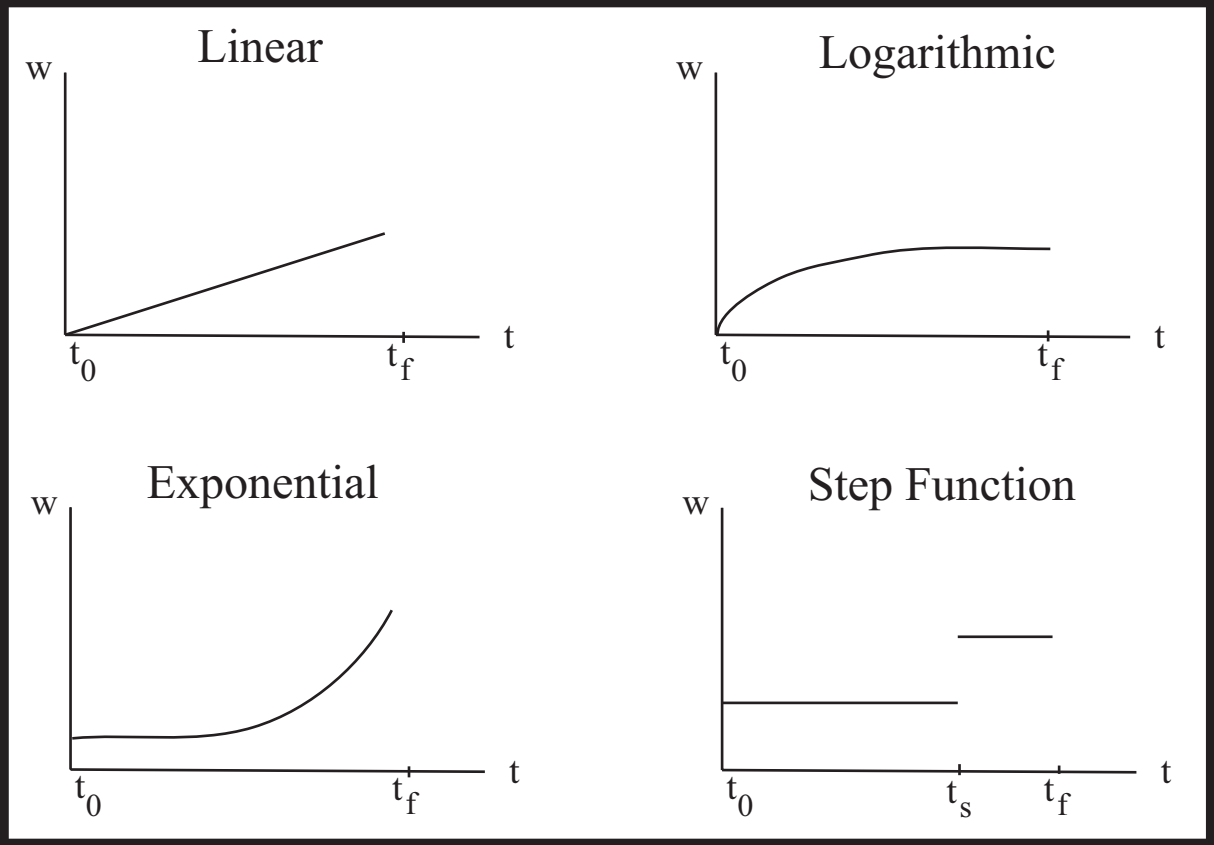
Weighted mHITS

- Weighted Point Matrix $\bar{\mathbf{P}} \geq 0$

$$\bar{\mathbf{p}}_{ij} = w_{ij} \ \mathbf{p}_{ij}$$

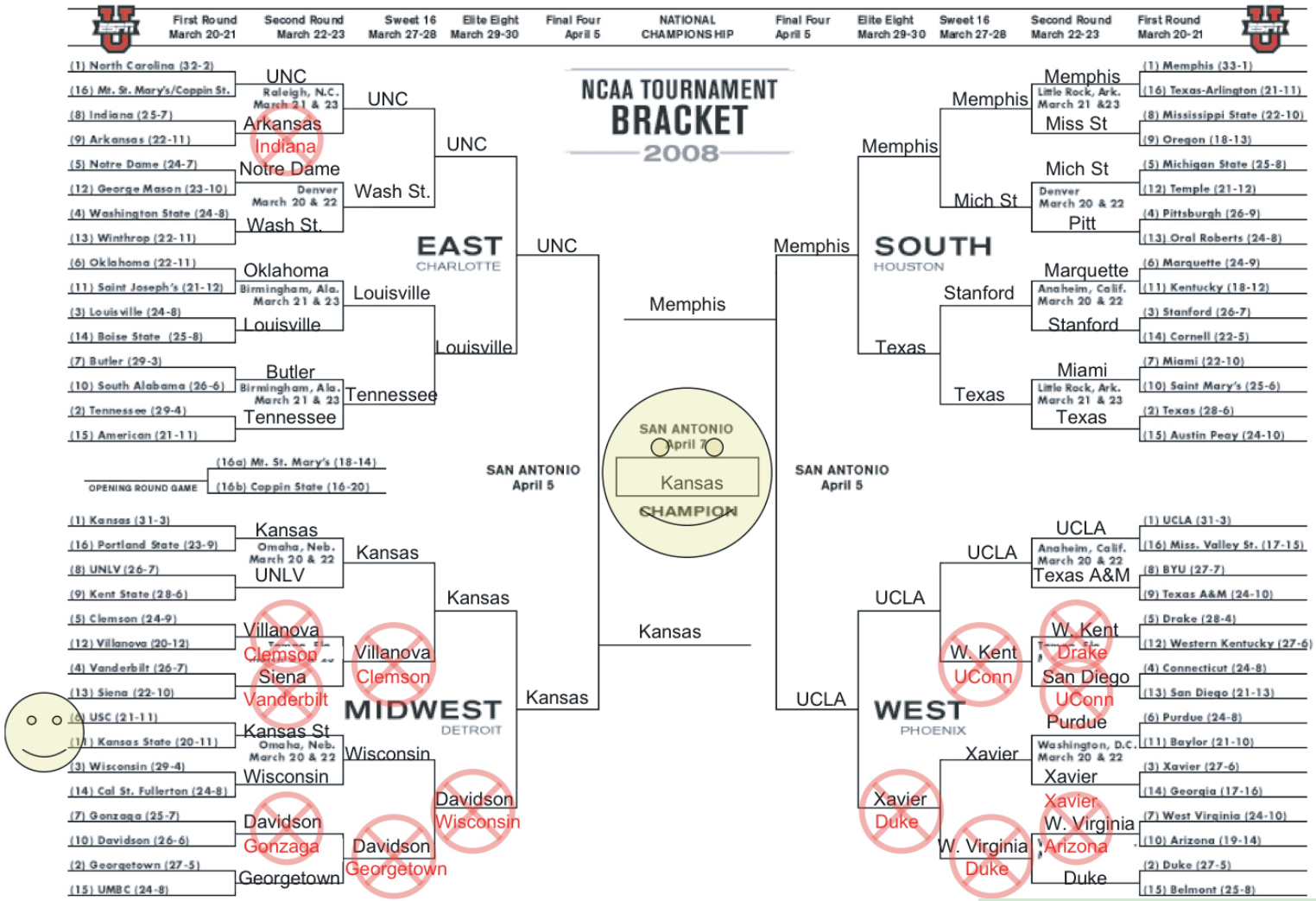
(w_{ij} = weight of matchup between teams i and j)

- possible weightings w_{ij}



mHITS Results: full NCAA

(image from Neil Goodson and Colin Stephenson)



mHITS - Log Weighted

ESPN Pts: 1450 out of 1680
National Rank: 834 out 3.6 mil.
Percentile: 100%`

mHITS Results: full NCAA

Method	ESPN score
Massey Linear	1450
Massey Log	1450
mHITS Log	1450
Massey Step	1420
Massey Exponential	1400
mHITS Step	1320
Massey Uniform	1310
mHITS Linear	1310
mHITS Uniform	1310
Colley Linear	1100
Colley Log	1010

- weightings can easily be applied to all ranking methods
  - $\Rightarrow$ interesting possibilities for weighted webpage ranking

mHITS Point Matrix $\mathbf{P} \geq 0$

Perron-Frobenius guarantees (for irreducible $\mathbf{P}$ with total support)

- existence of $\mathbf{o}$ and $\mathbf{d}$
- uniqueness of $\mathbf{o}$ and $\mathbf{d}$
- convergence of mHITS algorithm
- rate of convergence of mHITS algorithm

  $\sigma_2^2(\mathbf{S})$, where $\mathbf{S} = \mathbf{D}\,(1/\mathbf{o})\;\mathbf{P}\;\mathbf{D}\,(1/\mathbf{d})$

Recommendation Systems

Recommendation Systems

$$\mathbf{A} \geq 0$$

purchase history matrix

$$\mathbf{A} = \begin{array}{c} \\ \text{Item 1} \\ \text{Item 2} \\ \vdots \\ \text{Item m} \end{array} \begin{array}{c} \begin{array}{cccc} \text{User 1} & \text{User 2} & \dots & \text{User n} \end{array} \\ \begin{pmatrix} 1 & 5 & \dots & 0 \\ 0 & 0 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \dots & 2 \end{pmatrix} \end{array}$$

- Find similar users
- Find similar items
- Cluster users into classes

EX: use NMF so that $\mathbf{A}_{m\times n} \approx \mathbf{W}_{m\times k}\mathbf{H}_{k\times n}$

NMF Algorithm: Lee and Seung 2000

MEAN SQUARED ERROR OBJECTIVE FUNCTION

$$\min \|\mathbf{A} - \mathbf{W}\mathbf{H}\|_F^2$$

$$s.t. \quad \mathbf{W}, \mathbf{H} \geq 0$$

```
W = abs(randn(m,k));
H = abs(randn(k,n));
for i = 1 : maxiter
    H = H .* (W^T A) ./ (W^T W H + 10^-9);
    W = W .* (A H^T) ./ (W H H^T + 10^-9);
end
```

Many parameters affect performance (k, obj. function, sparsity constraints, algorithm, etc.).

— NMF is not unique!

(proof of convergence to fixed point based on E-M convergence proof)

Interpretation with NMF

- columns of **W** are the underlying basis vectors, i.e., each of the n columns of $\mathbf{A}_{m\times n}$ can be built from k columns of $\mathbf{W}_{m\times k}$.

- columns of **H** give the weights associated with each basis vector.

$$\mathbf{A}_{*1} \approx \mathbf{W}\mathbf{H}_{*1} = \begin{bmatrix} \vdots \\ \mathbf{w}_1 \\ \vdots \end{bmatrix} h_{11} + \begin{bmatrix} \vdots \\ \mathbf{w}_2 \\ \vdots \end{bmatrix} h_{21} + \cdots + \begin{bmatrix} \vdots \\ \mathbf{w}_k \\ \vdots \end{bmatrix} h_{k1}$$

$$\mathbf{A}_{*2} = \begin{array}{l} \text{item 1} \\ \text{item 2} \\ \text{item 3} \\ \text{item 4} \\ \text{item 5} \end{array} \begin{pmatrix} 5 \\ 0 \\ 0 \\ 11 \\ 1 \end{pmatrix} \approx \mathbf{W}\mathbf{H}_{*1} = \begin{bmatrix} 1.4 \\ 0 \\ 0 \\ 2.4 \\ 0 \end{bmatrix} .7 + \begin{bmatrix} 0 \\ 3.2 \\ 5.1 \\ 0 \\ 0 \end{bmatrix} .003 + \cdots + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 4.4 \\ 2.8 \end{bmatrix} .2$$

- **W**, **H** $\geq 0 \Rightarrow$ immediate interpretation (additive parts-based rep.)

Netflix

17770 movies, $\approx$ .5 million users

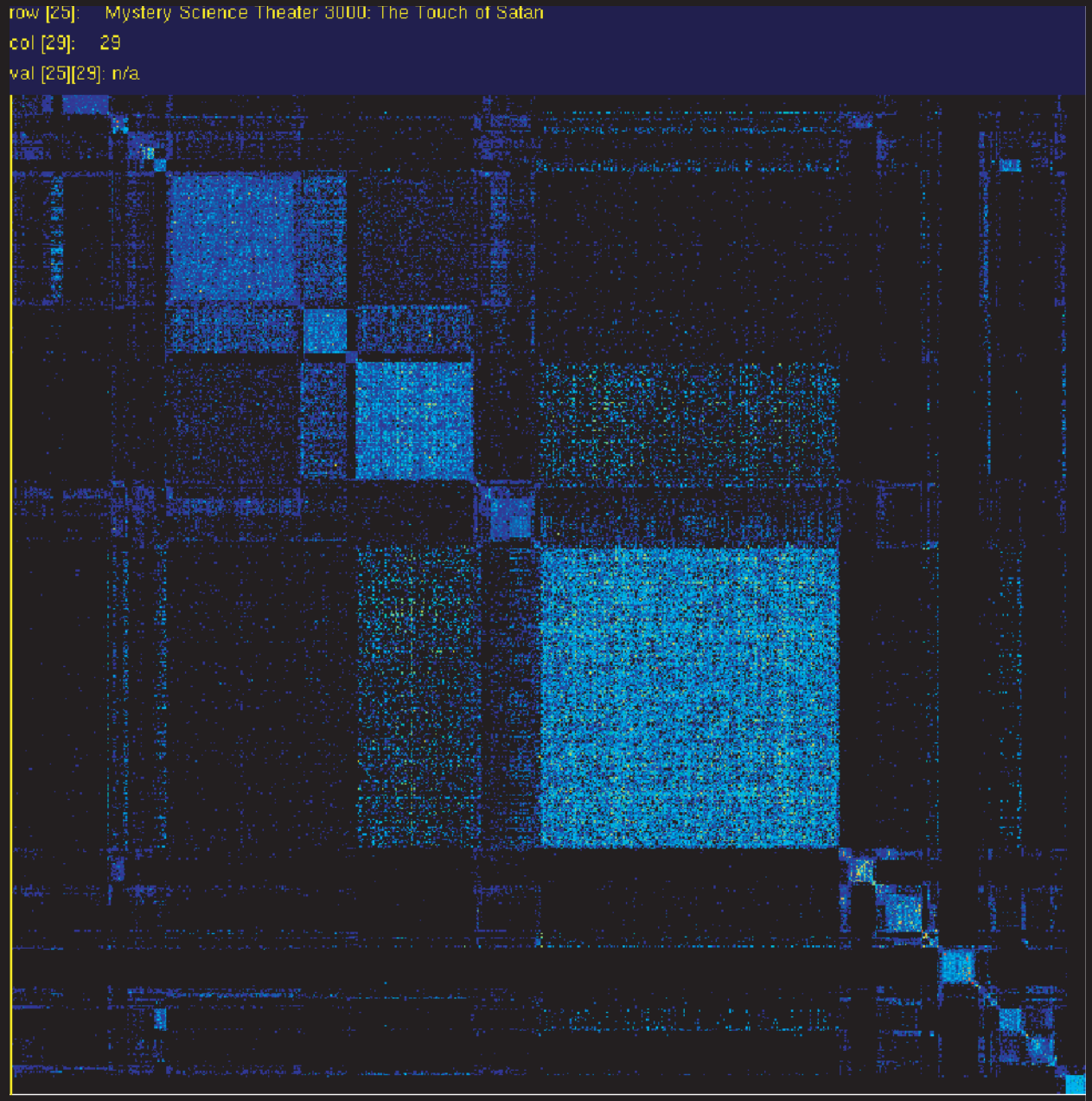
user rating matrix

$$\mathbf{A} = \begin{array}{c} \\ \text{movie 1} \\ \text{movie 2} \\ \vdots \\ \text{movie m} \end{array} \begin{array}{c} \begin{array}{cccc} \text{User 1} & \text{User 2} & \dots & \text{User n} \end{array} \\ \begin{pmatrix} 1 & 5 & \dots & 0 \\ 0 & 0 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \dots & 2 \end{pmatrix} \end{array}$$

- Find similar users
- Find similar movies
- Cluster users and movies into classes
- Rank users
- Rank movies

Clustering Netflix

(movie-movie matrix; 17770 x 17770)



(vismatrix tool - David Gleich)

Netflix

17770 movies, $\approx$.5 million users

user rating matrix

$$\mathbf{A} = \begin{array}{c} \\ \text{movie 1} \\ \text{movie 2} \\ \vdots \\ \text{movie m} \end{array} \begin{array}{c} \begin{array}{cccc} \text{User 1} & \text{User 2} & \dots & \text{User n} \end{array} \\ \begin{pmatrix} 1 & 5 & \dots & 0 \\ 0 & 0 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \dots & 2 \end{pmatrix} \end{array}$$

- Find similar users
- Find similar movies
- Cluster users and movies into classes
- Rank users
- Rank movies

mHITS on Netflix

each movie i gets a rating $\mathbf{m}_i$ and each user gets a rating $\mathbf{u}_j$

- mHITS Thesis: A movie is a good (i.e., deserves a high rating m_i) if it gets high ratings from good (i.e., discriminating) users. A user is good (i.e., deserves a high rating u_j) when his or her ratings match the true rating of a movie.

mHITS Netflix Algorithm

$\mathbf{u} = \mathbf{e}$;

for i = 1 : maxiter

$\mathbf{m} = \mathbf{A}\,\mathbf{u}$;

$\mathbf{m} = \frac{5(\mathbf{m} - min(\mathbf{m}))}{max(\mathbf{m}) - min(\mathbf{m})}$;

$\mathbf{u} = \frac{1}{((\mathbf{R} - (\mathbf{R} > 0).*(\mathbf{e}\mathbf{m}^T)).^2)\mathbf{e}}$;

end

Netflix mHITS results

subset: 1500 "super users" (rate $\geq$ 1000 movies)

1st	Raiders of the Lost Ark
2nd	Silence of the Lambs
3rd	The Sixth Sense
4th	Shawshank Redemption
5th	LOR: Fellowship of the Ring
6th	The Matrix
7th	LOR: The Two Towers
8th	Pulp Fiction
9th	LOR: The Return of the King
10th	Forrest Gump
11th	The Usual Suspects
12th	American Beauty
13th	Pirates of the Carribbean: Black Pearl
14th	The Godfather
15th	Braveheart

The Enron Email Dataset

(data from Mike Berry)

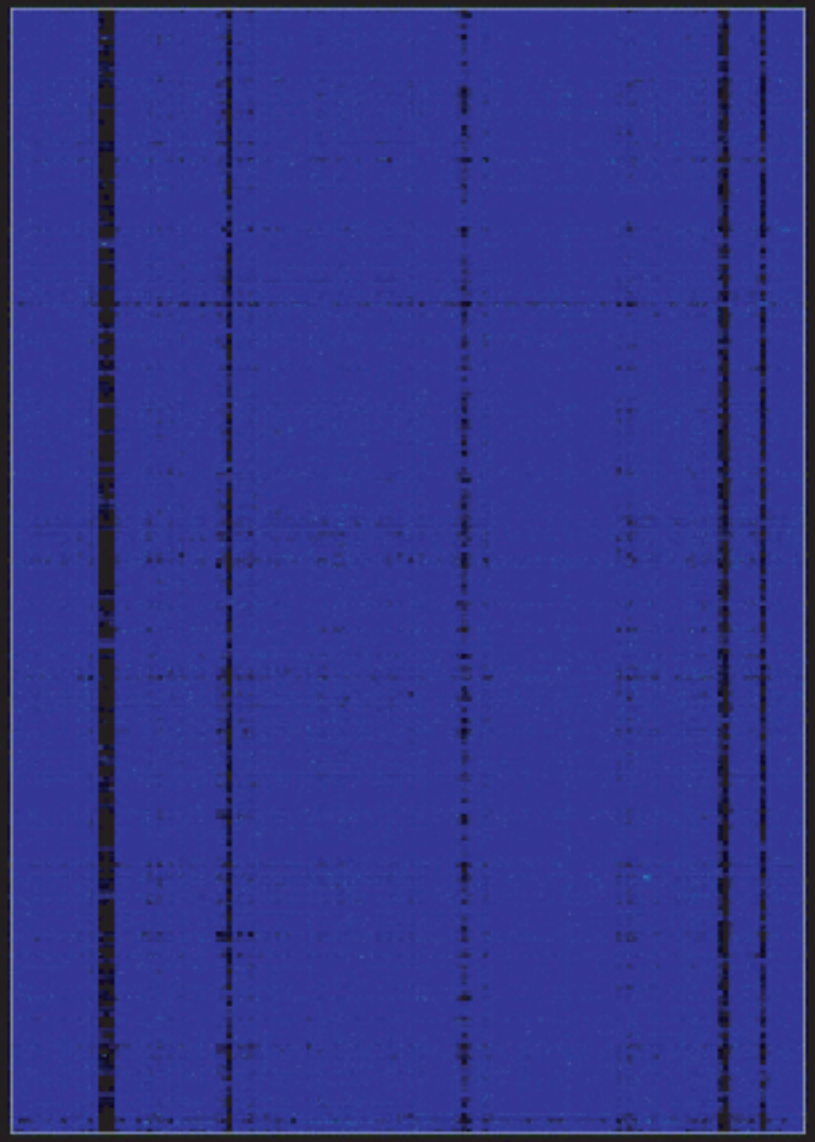
- PRIVATE email collection of 150 Enron employees during 2001
- 92,000 terms and 65,000 messages
- Term-by-Message Matrix

$$
\begin{array}{c}
\\
\vdots \\
subpoena \\
dynegy \\
\vdots
\end{array}
\begin{array}{c}
\begin{array}{cccc}
fastow\mathbf{1} & fastow\mathbf{2} & skilling\mathbf{1} & \ldots
\end{array} \\
\begin{pmatrix}
\vdots & \vdots & \vdots & \ldots \\
\mathbf{2} & \mathbf{0} & \mathbf{1} & \ldots \\
\mathbf{0} & \mathbf{3} & \mathbf{0} & \ldots \\
\vdots & \vdots & \vdots & \ldots
\end{pmatrix}
\end{array}
$$

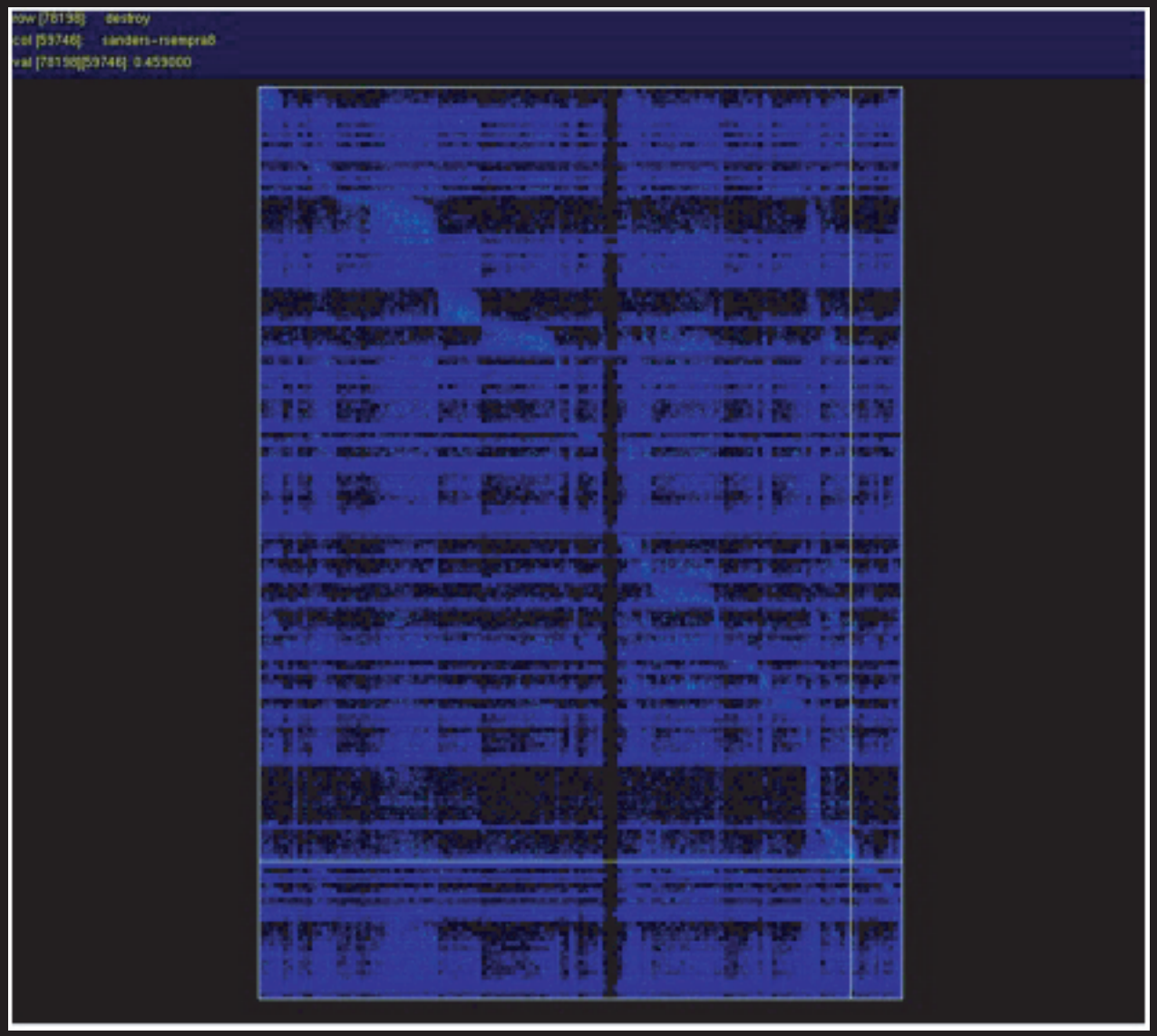
Text Mining Applications

- Data compression
- Find similar terms
- Find similar documents
- Cluster documents
- Cluster terms
- Topic detection and tracking

Unclustered Enron A



NMF-clustered Enron A



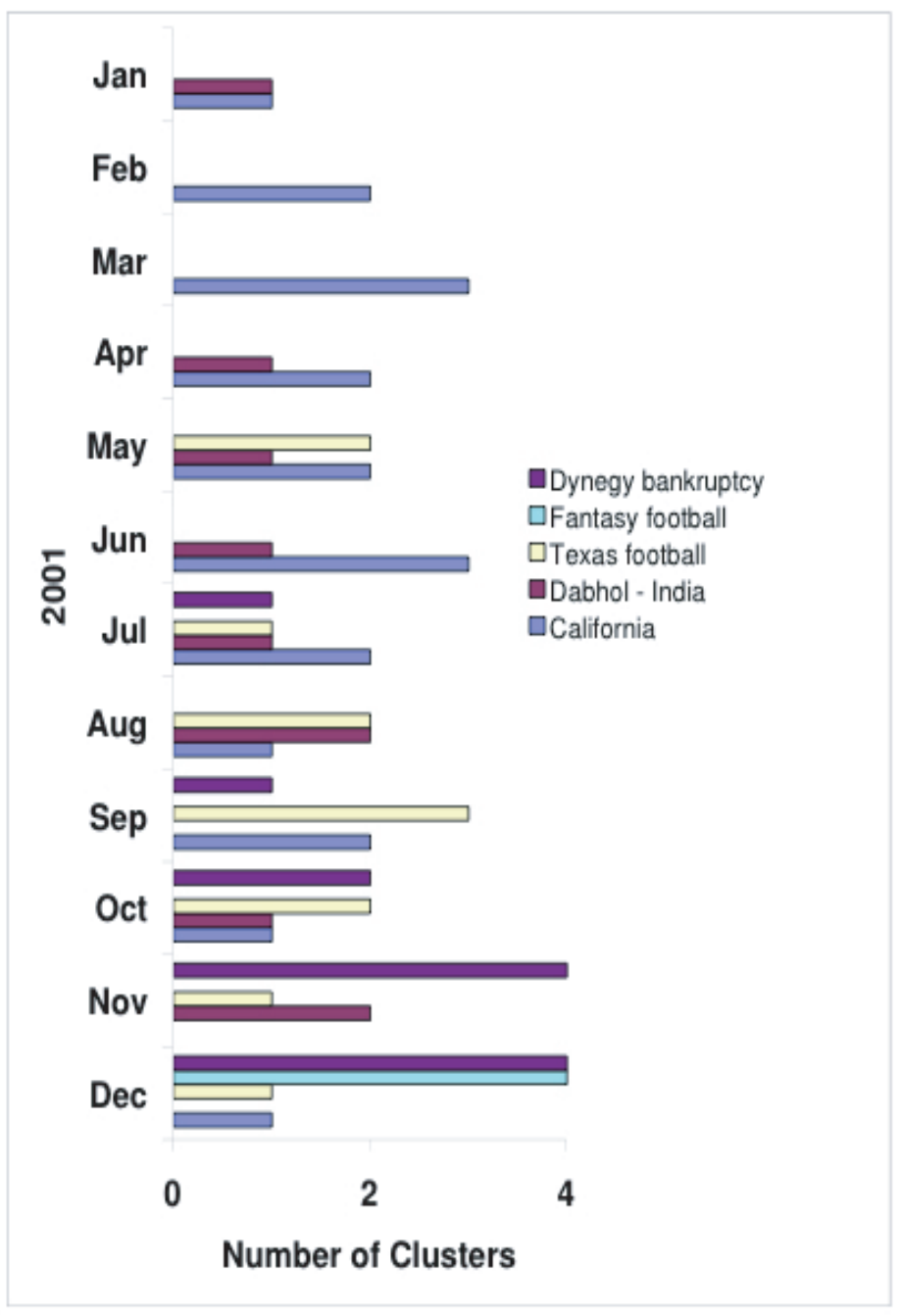
Clustering the Enron Email Dataset

(image from Mike Berry)

Feature Index (k)	Cluster Size	Topic Description	Dominant Terms
10	497	California	ca, **cpuc,** gov, **socalgas,** sempra, org, sce, gmssr, aelaw, ci
23	43	Louise Kitchen named top woman by Fortune	evp, **fortune,** britain, woman, **ceo**, avon, fiorinai, cfo, hewlett, packard
26	231	Fantasy football	game, wr, qb, play, rb, season, injury, updated, fantasy, image
33	233	Texas longhorn football newsletter	UT, orange, longhorn[s], texas, true, truorange, recruiting, oklahoma defensive
34	65	Enron collapse	**partnership[s], fastow**, shares, **sec**, stock, shareholder, investors, equity, **lay**
39	235	Emails about India	**dahhol, dpc, india, mseb, maharashtra,** indian, lenders, delhi, foreign, minister
46	127	Enron collapse	dow, debt, reserved, wall, copyright jones, cents, analysts, reuters, spokesman

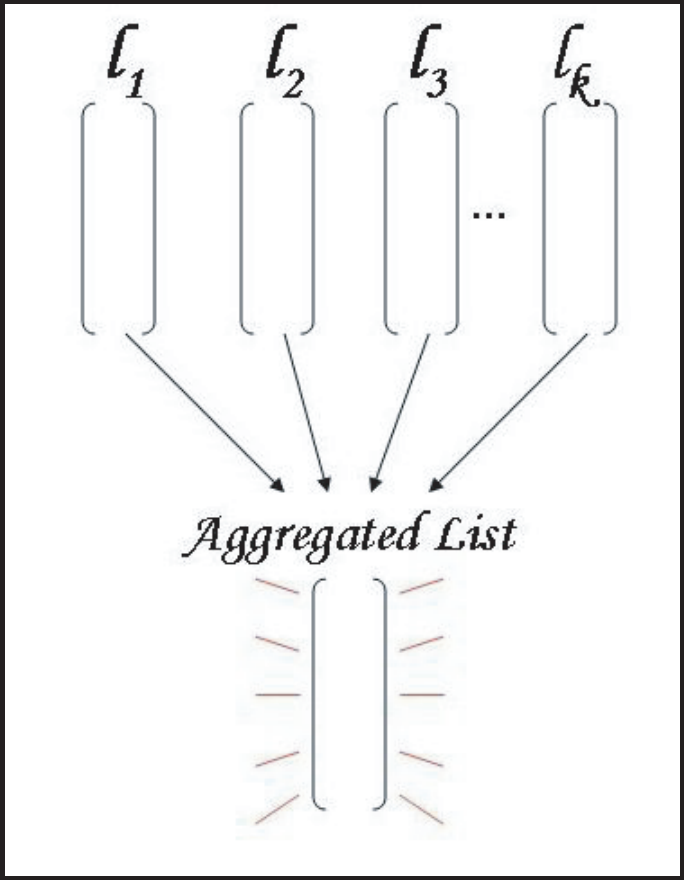
Tracking Enron clusters over time

(image from Mike Berry)



Meta-Algorithms

Rank Aggregation



Rank Aggregation

- average rank
- Borda count
- simulated data
- graph theory

Average Rank

	mHITS ($\mathbf{r} = \mathbf{o}/\mathbf{d}$)	Massey	Colley	Average Rating	Average Rank
Duke	5th	5th	5th	5	5th
Miami	2nd	1st	1st	$1.\overline{3}$	1st
UNC	4th	4th	3rd	$3.\overline{6}$	4th
UVA	3rd	3rd	4th	$3.\overline{3}$	3rd
VT	1st	2nd	2nd	$1.\overline{6}$	2nd

Borda Count

- for each ranked list, each item receives a score equal to the number of items it outranks.

	mHITS ($\mathbf{r} = \mathbf{o}/\mathbf{d}$)	Massey	Colley	Borda Count	Borda Rank
Duke	0	0	0	0	5th
Miami	3	4	4	11	1st
UNC	1	1	2	4	4th
UVA	2	2	1	5	3rd
VT	4	3	3	10	2nd

Simulated Data

3 ranked lists

	mHITS		*Massey*		*Colley*
1st	VT	1st	Miami	1st	Miami
2nd	Miami	2nd	VT	2nd	VT
3rd	UVA	3rd	UVA	3rd	UNC
4th	UNC	4th	UNC	4th	UVA
5th	Duke	5th	Duke	5th	Duke

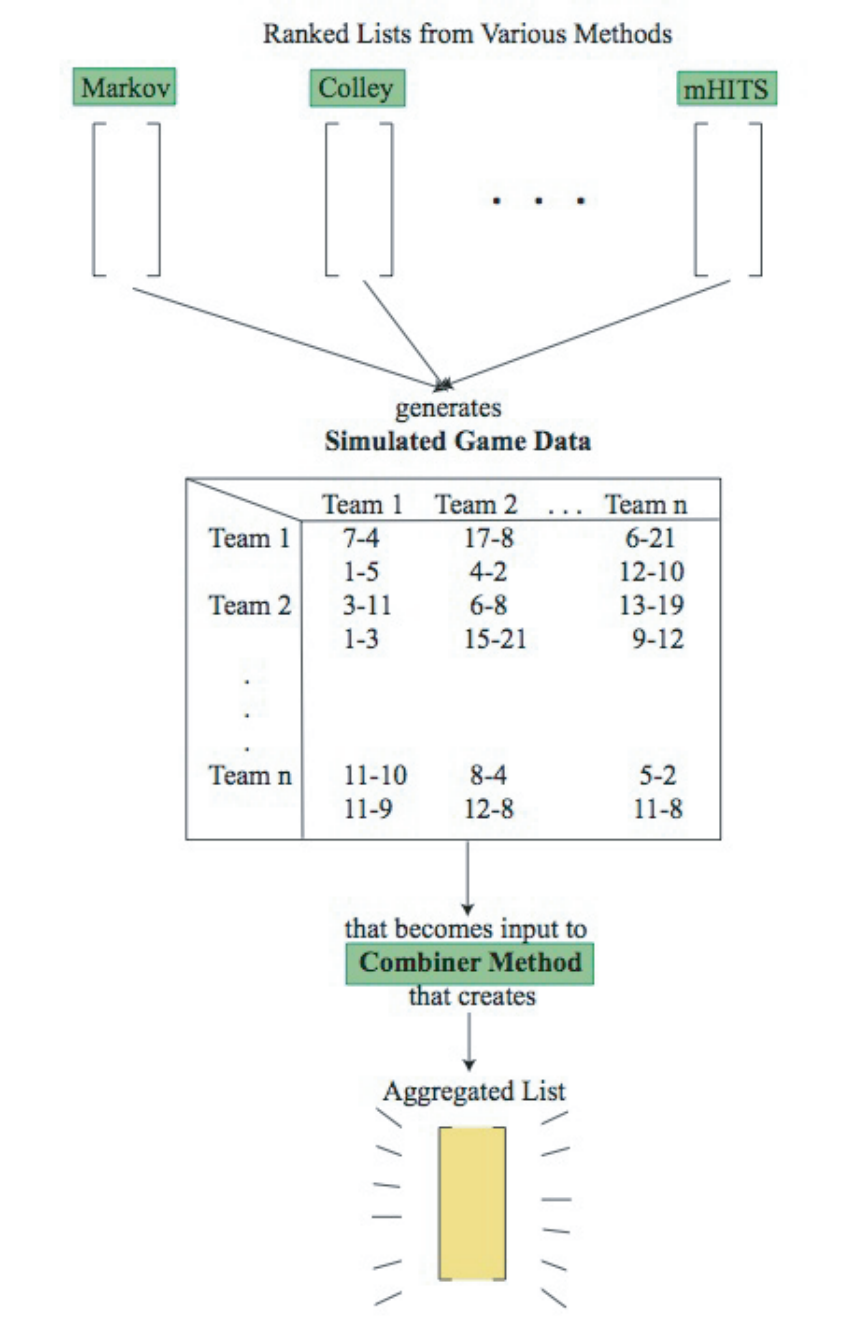
mHITS

- VT beats Miami by 1 point, UVA by 2 points, . . .
- Miami beats UVA by 1 point, UNC by 2 points, . . .
- UVA beats UNC by 1 point, Duke by 2 points
- UNC beats Duke by 1 point

repeat for each ranked list $\Rightarrow$ generates game scores for teams

Simulated Game Data

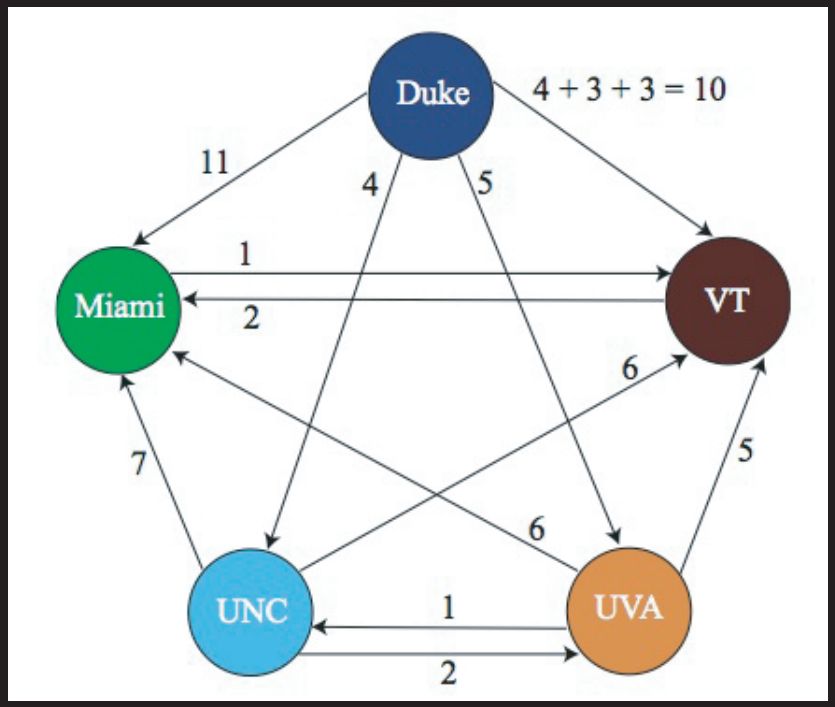
Simulated Data



Graph Theory

more voting

- ranked lists are used to form weighted graph
- possible weights
  - w_{ij} = # of ranked lists having i below j
  - w_{ij} = sum of rank differences of lists having i below j



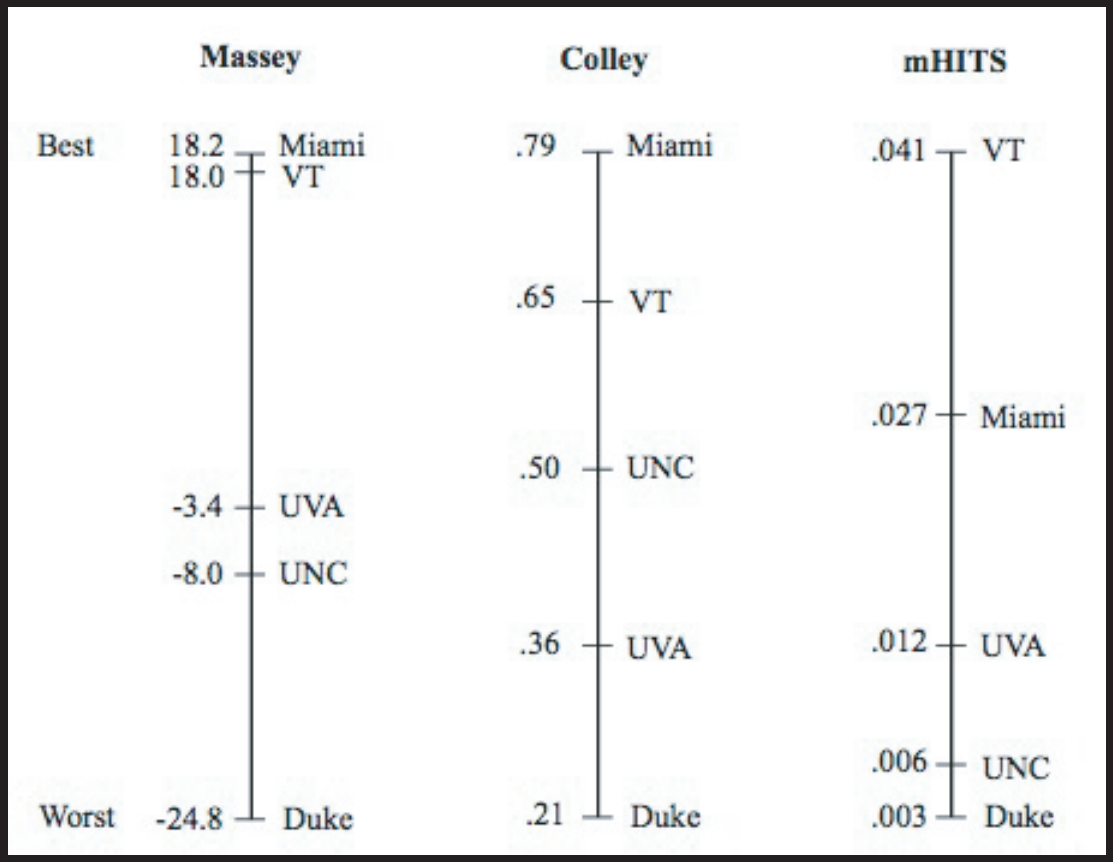
- run algorithm (e.g., Markov, PageRank, HITS) to determine most important nodes

Aggregation

Rating Aggregation

Rating Aggregation

rating vectors



- form rating differential matrix **R** for each rating vector

Rating Aggregation

rating differential matrices $\mathbf{R} \geq 0$

$$\mathbf{R}_{Massey} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 43 & 0 & 26.2 & 21.6 & .2 \\ 32.8 & 0 & 0 & 0 & 0 \\ 21.4 & 0 & 4.6 & 0 & 0 \\ 42.8 & 0 & 26 & 21.4 & 0 \end{pmatrix} \end{array}$$

$$\mathbf{R}_{Colley} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .58 & 0 & .29 & .43 & .14 \\ .29 & 0 & 0 & .14 & 0 \\ .15 & 0 & 0 & 0 & 0 \\ .44 & 0 & .15 & .29 & 0 \end{pmatrix} \end{array}$$

$$\mathbf{R}_{mHITS} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .024 & 0 & .021 & .015 & 0 \\ .003 & 0 & 0 & 0 & 0 \\ .009 & 0 & .006 & 0 & 0 \\ .038 & .014 & .035 & .029 & 0 \end{pmatrix} \end{array}$$

- differing scales $\longrightarrow$ NORMALIZE

Rating Aggregation

rating differential matrices

$$\bar{\mathbf{R}}_{Massey} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .1792 & 0 & .1092 & .09 & .0008 \\ .1367 & 0 & 0 & 0 & 0 \\ .0892 & 0 & .0192 & 0 & 0 \\ .1783 & 0 & .1083 & .0892 & 0 \end{pmatrix} \end{array}$$

$$\bar{\mathbf{R}}_{Colley} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .2 & 0 & .1 & .1483 & .0483 \\ .1 & 0 & 0 & .0483 & 0 \\ .0517 & 0 & 0 & 0 & 0 \\ .1517 & 0 & .0517 & .1 & 0 \end{pmatrix} \end{array}$$

$$\bar{\mathbf{R}}_{mHITS} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .1237 & 0 & .1082 & .0773 & 0 \\ .0155 & 0 & 0 & 0 & 0 \\ .0464 & 0 & .0309 & 0 & 0 \\ .1959 & .0722 & .1804 & .1495 & 0 \end{pmatrix} \end{array}$$

Rating Aggregation

rating differential matrices

$$\bar{\mathbf{R}}_{Massey} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .1792 & 0 & .1092 & .09 & .0008 \\ .1367 & 0 & 0 & 0 & 0 \\ .0892 & 0 & .0192 & 0 & 0 \\ .1783 & 0 & .1083 & .0892 & 0 \end{pmatrix} \end{array}$$

$$\bar{\mathbf{R}}_{Colley} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .2 & 0 & .1 & .1483 & .0483 \\ .1 & 0 & 0 & .0483 & 0 \\ .0517 & 0 & 0 & 0 & 0 \\ .1517 & 0 & .0517 & .1 & 0 \end{pmatrix} \end{array}$$

$$\bar{\mathbf{R}}_{mHITS} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .1237 & 0 & .1082 & .0773 & 0 \\ .0155 & 0 & 0 & 0 & 0 \\ .0464 & 0 & .0309 & 0 & 0 \\ .1959 & .0722 & .1804 & .1495 & 0 \end{pmatrix} \end{array}$$

- combine into one matrix $\longrightarrow$ AVERAGE

Rating Aggregation

rating differential matrices

$$\bar{\mathbf{R}}_{Massey} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .1792 & 0 & .1092 & .09 & .0008 \\ .1367 & 0 & 0 & 0 & 0 \\ .0892 & 0 & .0192 & 0 & 0 \\ .1783 & 0 & .1083 & .0892 & 0 \end{pmatrix} \end{array}$$

$$\bar{\mathbf{R}}_{Colley} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .2 & 0 & .1 & .1483 & .0483 \\ .1 & 0 & 0 & .0483 & 0 \\ .0517 & 0 & 0 & 0 & 0 \\ .1517 & 0 & .0517 & .1 & 0 \end{pmatrix} \end{array}$$

$$\bar{\mathbf{R}}_{mHITS} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ .1237 & 0 & .1082 & .0773 & 0 \\ .0155 & 0 & 0 & 0 & 0 \\ .0464 & 0 & .0309 & 0 & 0 \\ .1959 & .0722 & .1804 & .1495 & 0 \end{pmatrix} \end{array}$$

- combine into one matrix $\longrightarrow$ AVERAGE

$$\bar{\mathbf{R}}_{average} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.1676 & 0 & 0.1058 & 0.1052 & 0.0164 \\ 0.0841 & 0 & 0 & 0.0161 & 0 \\ 0.0624 & 0 & 0.0167 & 0 & 0 \\ 0.1753 & 0.0241 & 0.1135 & 0.1129 & 0 \end{pmatrix} \end{array}$$

Rating Aggregation

average rating differential matrix $\mathbf{R}_{average} \geq 0$

$$\bar{\mathbf{R}}_{average} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.1676 & 0 & 0.1058 & 0.1052 & 0.0164 \\ 0.0841 & 0 & 0 & 0.0161 & 0 \\ 0.0624 & 0 & 0.0167 & 0 & 0 \\ 0.1753 & 0.0241 & 0.1135 & 0.1129 & 0 \end{pmatrix} \end{array}$$

- run ranking method
  - Markov method on $\mathbf{R}^T_{average}$
  - row sums of $\mathbf{R}_{average}$ / col sums of $\mathbf{R}_{average}$
  - Perron vector of $\mathbf{R}_{average}$

Rating Aggregation

average rating differential matrix $\mathbf{R}_{average} \geq 0$

$$\bar{\mathbf{R}}_{average} = \begin{array}{c} \\ \text{Duke} \\ \text{Miami} \\ \text{UNC} \\ \text{UVA} \\ \text{VT} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Duke} & \text{Miami} & \text{UNC} & \text{UVA} & \text{VT} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.1676 & 0 & 0.1058 & 0.1052 & 0.0164 \\ 0.0841 & 0 & 0 & 0.0161 & 0 \\ 0.0624 & 0 & 0.0167 & 0 & 0 \\ 0.1753 & 0.0241 & 0.1135 & 0.1129 & 0 \end{pmatrix} \end{array}$$

- run ranking method
  - Markov method on $\mathbf{R}^T_{average}$
  - row sums of $\mathbf{R}_{average}$ / col sums of $\mathbf{R}_{average}$
  - Perron vector of $\mathbf{R}_{average}$

Team	Method 1 $\mathbf{r} = \mathbf{o}/\mathbf{d}$		Method 2 Markov $\mathbf{r}$		Method 3 Perron $\mathbf{r}$	
Duke	0	5th	.020	5th	.27	5th
Miami	16.4	2nd	.465	2nd	.58	2nd
UNC	.4	3rd	.025	3rd	.34	3rd
UVA	.3	4th	.024	4th	.33	4th
VT	26.0	1st	.466	1st	.61	1st

Cluster Aggregation

many clustering algorithms = many clustering results

$\Rightarrow$ Can we combine many results to make one super result?

CLUSTER AGGREGATION ALGORITHM

1. Create aggregation matrix $\mathbf{F} \geq 0$

   $\mathbf{f}_{ij}$ = # of methods having items i and j in the same cluster

2. Run favorite clustering method on $\mathbf{F}$

Cluster Aggregation Example

Method 1

item	cluster assignment
1	1
2	3
3	2
4	1
5	1
6	2

1 4 5

3 6 2

Method 2

item	cluster assignment
1	3
2	1
3	2
4	3
5	1
6	2

1 4

3 6 2 5

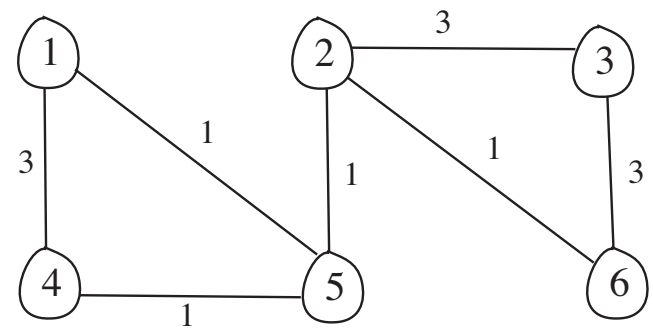
Method 3

item	cluster assignment
1	1
2	2
3	2
4	1
5	3
6	2

1 4

2 3 6 5

Cluster Aggregated Graph



Cluster Aggregated Results

Fiedler using just one eigenvector

1 4 5 2 3 6

Fiedler using two eigenvectors

1 4 3 6

5 2

Conclusions

- Applied problems often have nonnegative data.
- This nonnegativity is exploitable structure.

  $\Rightarrow$ proofs for convergence and rates of convergence

  $\Rightarrow$ proofs for existence and uniqueness
- Nonnegativity is easily interpretable. Often it pays to maintain nonnegativity throughout the modeling process.

  $\Rightarrow$ nonnegative matrix factorizations

  $\Rightarrow$ nonnegative rating vectors