

# Social and Information Network Analysis

## Link Analysis and Web Search



Department of Computer Science and Engineering  
University of North Texas

Acknowledgement: Ray Mooney

June 15, 2016

# Searching the Web: The Problem of Ranking

# Searching the Web




Florin Marin

[Web](#) [Images](#) [Maps](#) [Shopping](#) [News](#) [More ▾](#) [Search tools](#)

About 58,800,000 results (0.38 seconds)

**Cornell University**  
[www.cornell.edu/](http://www.cornell.edu/) ▾  
Cornell University contains seven undergraduate colleges plus the College of Veterinary Medicine, the Law School, the Samuel Curtis Johnson Graduate ...  
Score: **25** / 30 · **41 Google reviews** · [Write a review](#)

 410 Thurston Ave Ithaca, NY 14850  
(607) 255-5241



[Admissions](#) · [Academics](#) · [CUInfo](#)

**Cornell University** - Wikipedia, the free encyclopedia  
[en.wikipedia.org/wiki/Cornell\\_University](http://en.wikipedia.org/wiki/Cornell_University) ▾  
Cornell University is an American private Ivy League research university located in Ithaca, New York, United States. Founded in 1865 by Ezra Cornell and ...  
[History](#) · [Ithaca, New York](#) · [List of Cornell University alumni](#) · [Arts and Sciences](#)

**Cornell University Athletics**  
[www.cornellbigred.com/](http://www.cornellbigred.com/) ▾  
Official web site of Big Red athletics. Information about varsity sports, facilities, schedules, and the department, as well as an alumni section and Big Red Store.

**Cornell University (Cornell) on Twitter**  
<https://twitter.com/Cornell> ▾  
The latest from Cornell University (@Cornell). Cornell University Twitter feed. Ithaca, NY.

**Cornell Home**  
[www.cornellcollege.edu/](http://www.cornellcollege.edu/) ▾  
Residential liberal arts college established in 1853. Operates under the distinctive One-Course-At-A-Time academic calendar.



## Cornell University

80,190 followers on Google+

[Directions](#) [Follow](#)


Cornell University is an American private Ivy League research university located in Ithaca, New York, United States. Wikipedia

**Address:** 410 Thurston Ave, Ithaca, NY 14850  
**Acceptance rate:** 16.2% (2012)  
**Mascot:** Big Red Bear  
**Phone:** (607) 255-5241  
**Colors:** White, Carmelian  
**Founders:** [Andrew Dickson White](#), [Ezra Cornell](#)

### Recent posts



# Searching the Web



Florin Marin

[Web](#) [Images](#) [Maps](#) [Shopping](#) [News](#) [More](#) [Search tools](#)

About 58,800,000 results (0.38 seconds)

**Cornell University**  
[www.cornell.edu/](http://www.cornell.edu/) ✓  
Cornell University contains seven undergraduate colleges plus the College of Veterinary Medicine, the Law School, the Samuel Curtis Johnson Graduate ...  
Score: **25** / 30 · **41 Google reviews** · [Write a review](#)

410 Thurston Ave Ithaca, NY 14850  
(607) 255-5241



[Admissions](#) · [Academics](#) · [CUinfo](#)

**Cornell University - Wikipedia, the free encyclopedia**  
[en.wikipedia.org/wiki/Cornell\\_University](http://en.wikipedia.org/wiki/Cornell_University) ✓  
Cornell University is an American private Ivy League research university located in Ithaca, New York, United States. Founded in 1865 by Ezra Cornell and ...  
[History](#) · [Ithaca, New York](#) · [List of Cornell University alumni](#) · [Arts and Sciences](#)

**Cornell University Athletics**  
[www.cornellbigred.com/](http://www.cornellbigred.com/) ✓  
Official web site of Big Red athletics. Information about varsity sports, facilities, schedules, and the department, as well as an alumni section and Big Red Store.

**Cornell University (Cornell) on Twitter**  
<https://twitter.com/Cornell> ✓  
The latest from Cornell University (@Cornell). Cornell University Twitter feed. Ithaca, NY.

**Cornell Home**  
[www.cornellcollege.edu/](http://www.cornellcollege.edu/) ✓  
Residential liberal arts college established in 1853. Operates under the distinctive One-Course-At-A-Time academic calendar.



## Cornell University

80,190 followers on Google+

[Directions](#) [Follow](#)

Cornell University is an American private Ivy League research university located in Ithaca, New York, United States. [Wikipedia](#)

**Address:** 410 Thurston Ave, Ithaca, NY 14850  
**Acceptance rate:** 16.2% (2012)  
**Mascot:** Big Red Bear  
**Phone:** (607) 255-5241  
**Colors:** White, Carmelian  
**Founders:** [Andrew Dickson White](#), [Ezra Cornell](#)

Recent posts

How did Google “know” that “Cornell University” is the best answer?

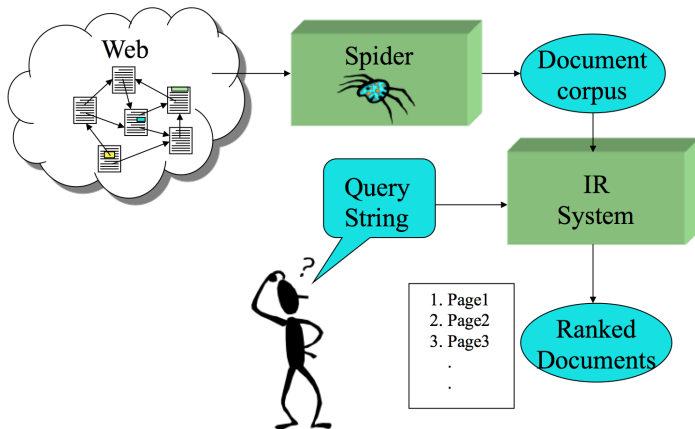
# Searching the Web: The Problem of Ranking

- When issuing the single-word query “Cornell,” a search engine does not have very much to go on.
  - Did the searcher want information about the university?
  - The university’s hockey team?
  - Cornell College in Iowa?
  - The Nobel-Prize-winning physicist Eric Cornell?

# Searching the Web: The Problem of Ranking

- Search engines determine how to rank pages using automated methods that look at the Web itself, not some external source of knowledge.
- There must be enough information *intrinsic* to the Web and its structure to figure out that “Cornell University” is the best answer.

# Web Search System



## Key issues for search engines:

- To filter, from among an enormous number of relevant documents, the few that are most important

# Web Search System

Understanding the network structure of Web pages is crucial for understanding what documents a search engine should return!

Back to the query “Cornell”:

- No internal features of the page [www.cornell.edu](http://www.cornell.edu) are really helpful:
  - “Cornell” does not necessarily occur more frequently within this page content than within others, relevant to the query



# Web Search System

Understanding the network structure of Web pages is crucial for understanding what documents a search engine should return!

Back to the query “Cornell”:

- No internal features of the page [www.cornell.edu](http://www.cornell.edu) are really helpful:
  - “Cornell” does not necessarily occur more frequently within this page content than within others, relevant to the query
- Rather, features extracted from the [link structure](#) are more helpful:
  - When a page is relevant to the query “Cornell”, very often it links to [www.cornell.edu](http://www.cornell.edu)

## Link Analysis using Hubs and Authorities

# Links are Essential to Ranking!

- We can use links to assess the **authority of a page on a topic**, through implicit endorsements that other pages on the topic confer through their links to it.
- **Experiment with the query “Cornell”:**
  - Collect pages that are relevant to “Cornell” using IR (text-only) techniques.
  - Let these pages “vote” through their links for pages on the Web.
  - Which page on the Web receives the greatest number of in-links from pages that are relevant to Cornell?

# Links are Essential to Ranking!

- We can use links to assess the **authority of a page on a topic**, through implicit endorsements that other pages on the topic confer through their links to it.
- **Experiment with the query “Cornell”:**
  - Collect pages that are relevant to “Cornell” using IR (text-only) techniques.
  - Let these pages “vote” through their links for pages on the Web.
  - Which page on the Web receives the greatest number of in-links from pages that are relevant to Cornell?
    - Answer: [www.cornell.edu](http://www.cornell.edu)

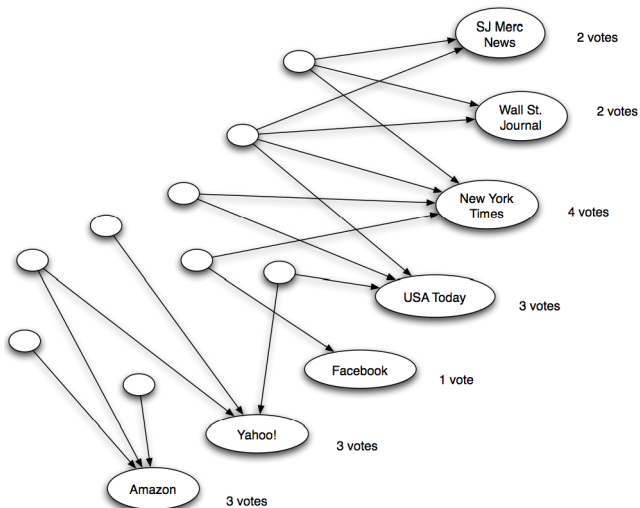
# Links are Essential to Ranking!

- Experiment with the query “newspapers”:
  - What is the “best” answer to the query “newspapers”?

# Links are Essential to Ranking!

- Experiment with the query “newspapers”:
  - What is the “best” answer to the query “newspapers”?
    - No single right answer
  - Best expected answer: [A list of most important ones](#)
  - Collect pages relevant to “newspapers” and let them vote through their links
    - **Result:** a mix of prominent newspapers along with pages that are going to receive a lot of in-links no matter what the query is - pages like Yahoo!, Facebook, and Amazon.

# In-Links

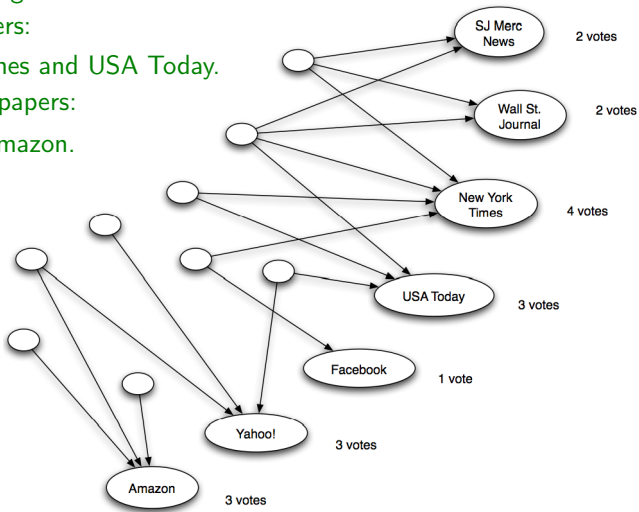


The unlabeled circles represent pages relevant to the query "newspaper."

# In-Links

Four highly-ranked pages:

- two are newspapers:
  - New York Times and USA Today.
- two are not newspapers:
  - Yahoo! and Amazon.



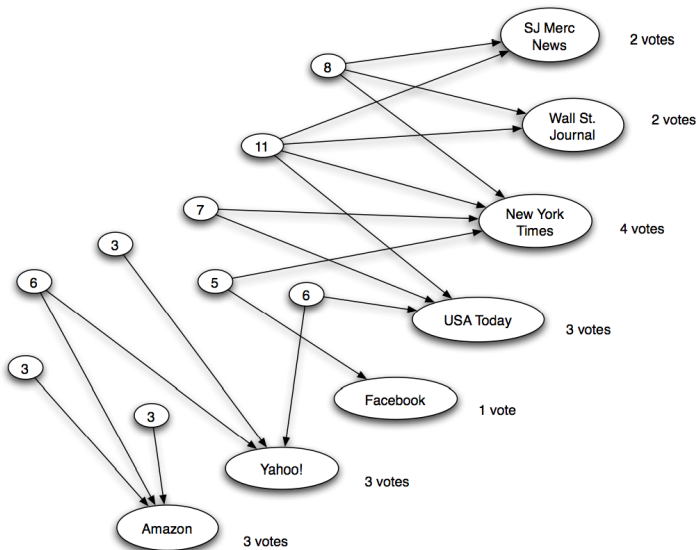
The unlabeled circles represent pages relevant to the query "newspaper."



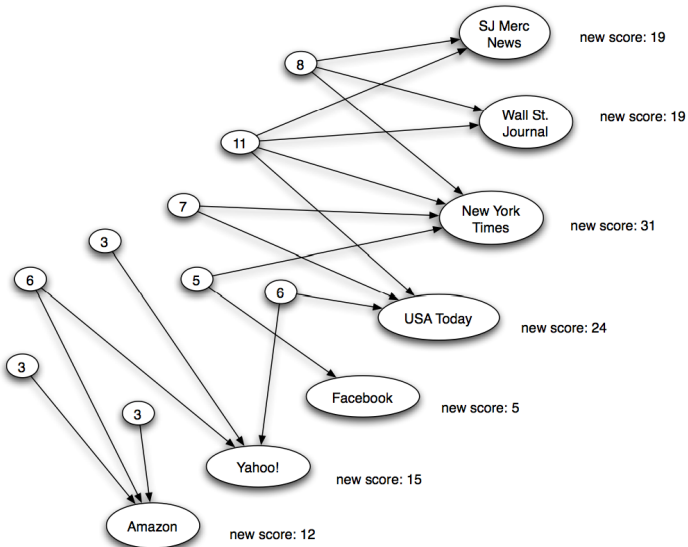
# Finding Good Lists

- In addition to the newspapers themselves, there is another kind of useful answer to our query: pages that compile lists of resources relevant to the topic.
- If we could find good list pages for newspapers, we would have another approach to the problem of finding the newspapers themselves.
- Intuitively, these pages have some sense where the good answers are, and we score them highly as lists.
  - A page's value as a list is equal to the sum of the votes received by all pages that it voted for.

# Finding Good Lists



# Re-Weighting



# Authorities and Hubs

- **Authorities for a query** are pages that are recognized as providing significant, trustworthy, and useful information on a topic
  - In-degree (number of pointers to a page) is one simple measure of authority
  - However in-degree treats all links as equal
  - Links from pages that are themselves authoritative should count more
- **Hubs for a query** are index pages that provide lots of useful links to relevant content pages (topic authorities)

# Authorities and Hubs - Examples

- Authorities:
  - Newspaper home pages
  - Course home pages
  - Home pages of auto manufacturers
- Hubs
  - List of newspapers
  - Course bulletin
  - List of US auto manufacturers

# Ranking by Hyperlink-Induced Topic Search (HITS) algorithm

- Algorithm developed by Kleinberg in 1998, as part of IBM's Clever search project
- Attempts to computationally determine hubs and authorities on a particular topic through analysis of a relevant subgraph of the web
- Based on mutually recursive facts:
  - Hubs point to lots of authorities
  - Authorities are pointed to by lots of hubs

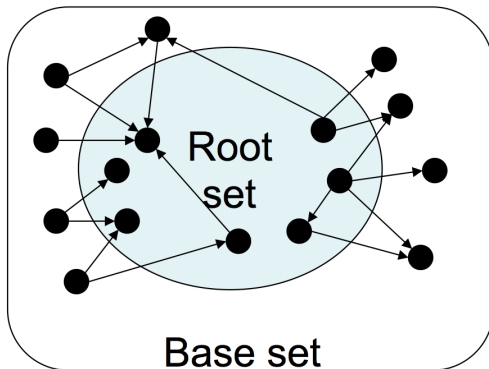
# The HITS Algorithm

- Computes hubs and authorities for a particular topic specified by a normal query
- First determines a set of relevant pages for the query called the base set  $S$
- Analyze the link structure of the web subgraph defined by  $S$  to find authority and hub pages in this set



## Constructing a Base Subgraph

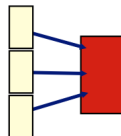
- For a specific query  $Q$ , let the set of documents returned by a standard search engine be called **the root set  $R$**
- Initialize **the base set  $S$**  to  $R$
- Add to  $S$  all pages pointed to by any page in  $R$
- Add to  $S$  all pages that point to any page in  $R$



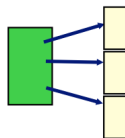
# HITS

Goal: Given a query, find:

- Good sources of content (authorities)



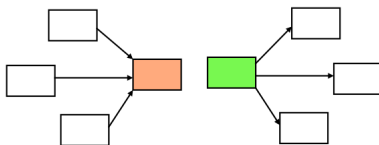
- Good sources of links (hubs)



# Intuition

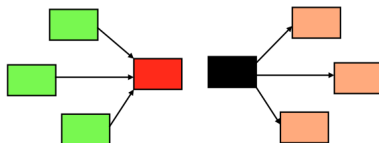
- **Authority** comes from in-edges.

Being a **good hub** comes from out-edges.



- **Better authority** comes from in-edges from **good hubs**.

Being a **better hub** comes from out-edges to **good authorities**.



# Iterative Algorithm

- Use an iterative algorithm to slowly converge on a mutually reinforcing set of hubs and authorities
- Maintain for each page  $p \in S$  :
  - Authority score:  $a_p$  (vector  $\mathbf{a}$ )
  - Hub score:  $h_p$  (vector  $\mathbf{h}$ )
- Initialize all  $a_p = h_p = 1$
- Maintain normalized scores:

$$\sum_{p \in S} (a_p)^2 = 1 \text{ and } \sum_{p \in S} (h_p)^2 = 1$$

# HITS Update Rules

- Authorities are pointed to by lots of good hubs:

$$a_p = \sum_{q:q \rightarrow p} h_q$$

- Hubs point to lots of good authorities:

$$h_p = \sum_{q:p \rightarrow q} a_q$$

- Repeat until vectors **a** and **h** converge

# The HITS Iterative Algorithm

- Initialize  $a_p = h_p = 1$  for all  $p \in S$
- For  $i = 1$  to  $k$ :
  - For all  $p \in S$  : update authority scores

$$a_p = \sum_{q:q \rightarrow p} h_q$$

- For all  $p \in S$  : normalize **a**

$$a_p = a_p / c, c : \sum_{p \in S} (a_p / c)^2 = 1$$

- For all  $p \in S$  : update hub scores

$$h_p = \sum_{q:p \rightarrow q} a_q$$

- For all  $p \in S$  : normalize **h**

$$h_p = h_p / c, c : \sum_{p \in S} (h_p / c)^2 = 1$$

# The HITS Iterative Algorithm

```
1 G := set of pages
2 for each page p in G do
3   p.auth = 1 // p.auth is the authority score of the page p
4   p.hub = 1 // p.hub is the hub score of the page p
5 function HubsAndAuthorities(G)
6   for step from 1 to k do // run the algorithm for k steps
7     norm = 0
8     for each page p in G do // update all authority values first
9       p.auth = 0
10      for each page q in p.incomingNeighbors do // p.incomingNeighbors is the set of pages that link to p
11        p.auth += q.hub
12      norm += square(p.auth) // calculate the sum of the squared auth values to normalise
13    norm = sqrt(norm)
14    for each page p in G do // update the auth scores
15      p.auth = p.auth / norm // normalise the auth values
16    norm = 0
17    for each page p in G do // then update all hub values
18      p.hub = 0
19      for each page r in p.outgoingNeighbors do // p.outgoingNeighbors is the set of pages that p links to
20        p.hub += r.auth
21      norm += square(p.hub) // calculate the sum of the squared hub values to normalise
22    norm = sqrt(norm)
23    for each page p in G do // then update all hub values
24      p.hub = p.hub / norm // normalise the hub values
```

# Convergence

What happens if we do this for larger and larger values of  $k$ ?

- Algorithm converges to a **fix-point** if iterated indefinitely
- In practice, 20 iterations produce fairly stable results
- Regardless of the **initial** hub and authority values (provided they are positive), we generally reach the same limiting values



# Finding Similar Pages Using Link Structure

- Given a page,  $p$ , let  $R$  (the root set) be  $t$  (e.g., 200) pages that point to  $p$ .
- Grow a base set  $S$  from  $R$ .
- Run HITS on  $S$ .
- Return the best authorities in  $S$  as the best similar-pages for  $p$ .
- Finds authorities in the “link neighborhood” of  $p$ .

# Similar Page Results

- Given “honda.com”
  - toyota.com
  - ford.com
  - bmwusa.com
  - saturncars.com
  - nissanmotors.com
  - audi.com
  - volvocars.com

# The PageRank Algorithm

# PageRank

## Query-independent ranking algorithm:

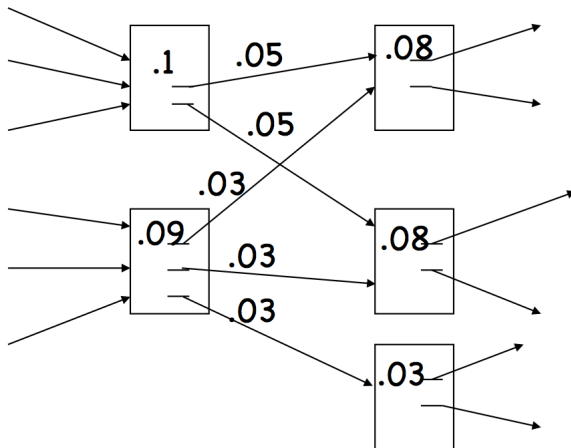
- Alternative link-analysis method used by Google (Brin & Page, 1998)
- Does not attempt to capture the distinction between hubs and authorities
- Ranks pages just by authority
- Applied to the entire web rather than a local neighborhood of pages surrounding the results of a query
- The endorsement that forms the basis for the PageRank measure of importance is that a page is important if it is cited by other important pages

# Idea Behind PageRank

- Just measuring in-degree (citation count) doesn't account for the authority of the source of a link
- PageRank starts with the simple “voting” based on in-links
- Nodes repeatedly pass endorsements across their out-going links, with the weight of a node's endorsement based on the current estimate of its PageRank
  - More important nodes make stronger endorsements

# Initial PageRank Idea

- Can view it as a process of PageRank “flowing” from pages to the pages they cite.



# Initial PageRank Algorithm

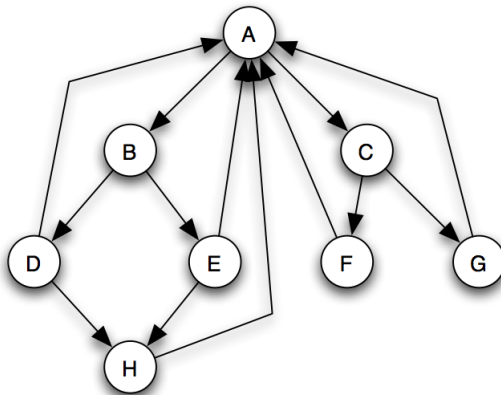
- Iterate rank-flowing process until convergence:
- Let  $S$  be the set of pages
- Initialize  $R(A) = \frac{1}{|S|} = \frac{1}{n}$  for all  $A \in S$
- Until ranks do not change (convergence)
  - For each  $A \in S$  :

$$R'(A) = \sum_{B \rightarrow A} \frac{1}{out(B)} R(B)$$

$$c = 1 / \sum_{A \in S} R'(A)$$

- For each  $A \in S$  :  $R(A) = cR'(A)$  (normalize)

## PageRank Algorithm - Example



What are the PageRank values after the first two updates?



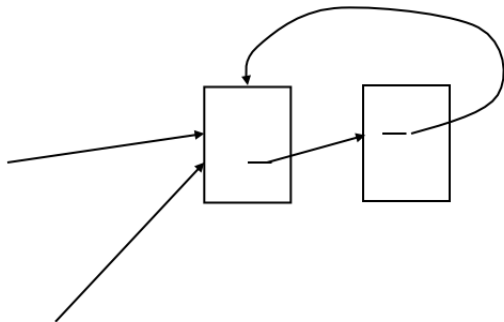
# PageRank Algorithm - Example

## Result

Step	A	B	C	D	E	F	G	H
1	$1/2$	$1/16$	$1/16$	$1/16$	$1/16$	$1/16$	$1/16$	$1/8$
2	$3/16$	$1/4$	$1/4$	$1/32$	$1/32$	$1/32$	$1/32$	$1/16$

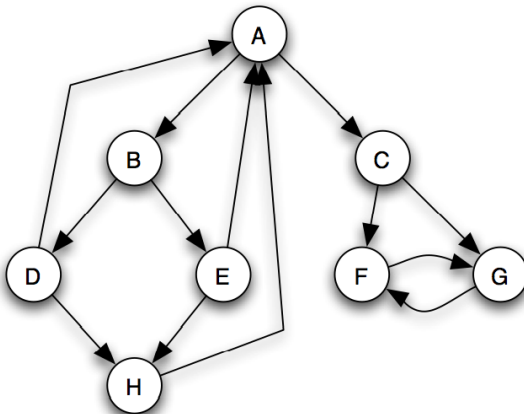
# Problem with Initial Idea

- The web is full of dead-ends
  - A group of pages that only point to themselves but are pointed to by other pages act as a “rank sink” and absorb all the rank in the system



Rank flows into cycle  
and can't get out

## PageRank Algorithm - Modified Example



PageRank that flows from C to F and G can never circulate back into the network - links out of C function as a “slow leak”

# Teleporting

- At a dead end, jump to a random web page
- At any non-dead end, with probability 10%, jump to a random web page
- With remaining probability (90%), go out on a random link
  - 10% - the  $\epsilon$  parameter

$$R(A) = c \left( \epsilon/n + (1 - \epsilon) \sum_{(B,A) \in G} R(B)/out(B) \right)$$

- $c$  is a normalizing constant set so that the rank of all pages always sums to 1
- Result of teleporting: it cannot get stuck locally

# The PageRank Algorithm

- Let  $S$  be the total set of pages and  $n = |S|$
- Choose  $\epsilon$  s.t.  $0 < \epsilon < 1$ , e.g., 0.15
- Initialize  $R(A) = \frac{1}{n}$  for all  $A \in S$
- Until ranks do not change (convergence)
  - For each  $A \in S$  :

$$R'(A) = \left[ (1 - \epsilon) \sum_{B \rightarrow A} \frac{R(B)}{\text{out}(B)} \right] + \frac{\epsilon}{n}$$

$$c = 1 / \sum_{A \in S} R'(A)$$

- For each  $A \in S$  :  $R(A) = cR'(A)$  (normalize)

# The Random Surfer Model

- PageRank can be seen as modeling a “random surfer” that starts on a random page and then at each point:
  - With probability  $\frac{\epsilon}{n}$  randomly jumps to page A
  - Otherwise, randomly follows a link on the current page
- $R(A)$  models the probability that this random surfer will be on page A at any given time
- “Jumps” are needed to prevent the random surfer from getting “trapped” in web sinks with no outgoing links

# Speed of Convergence

- Early experiments on Google used 322 million links
- PageRank algorithm converged (within small tolerance) in about 52 iterations

# PageRank Retrieval

- Preprocessing:
  - Given graph of links, compute the rank of each page A
- Query processing:
  - Retrieve pages meeting query
  - Rank them by their PageRank
  - Order is query-independent
- The reality
  - PageRank is used in Google, but so are many other clever heuristics



# PageRank vs. HITS

- Computation
  - Once for all documents and queries (offline)
- Query-independent
  - Requires combination with query-dependent criteria
- Computation
  - Requires computation for each query
- Query-dependent
- Quality depends on quality of start set
- Gives hubs as well as authorities

# Link Analysis Conclusions

- Link analysis uses information about the structure of the web graph to aid search
- It is one of the major innovations in web search
- It is the primary reason for Google's success