Web Searches and Link Analysis

Caroline Barrière
March 15th 2019

Outline

- Characterizing the Web
  - Early Development and Search Models
  - Size/Growth and Business Model
  - Web Challenges for IR
- Web Crawling
  - Crawler overview
  - Crawler architecture / strategies
- Link Analysis
  - Anchor Text Indexing
  - PageRank
- Using PageRank for retrieval
- Summary / Quiz 4

References

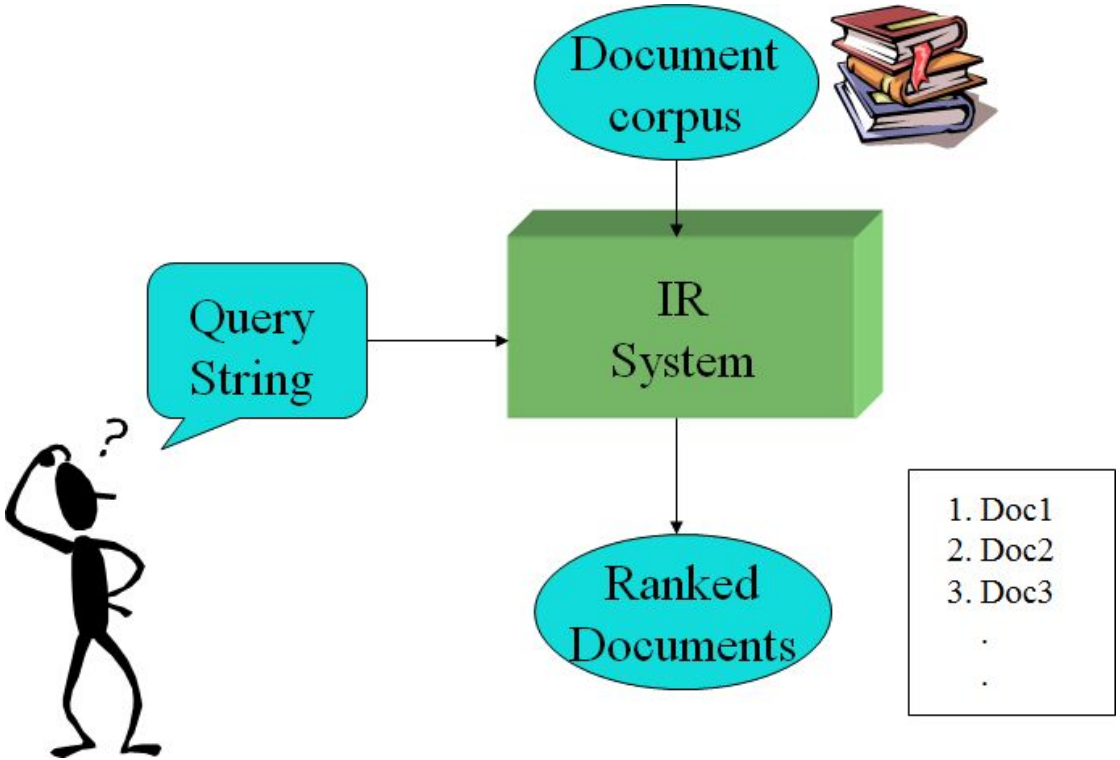
Textbook
Introduction to IR, by Manning et al. 2009

Chapter 19, Web Search basics
https://nlp.stanford.edu/IR-book/pdf/19web.pdf
Chapter 20, Web Crawling and indexes
https://nlp.stanford.edu/IR-book/pdf/20crawl.pdf
Chapter 21, Link Analysis
https://nlp.stanford.edu/IR-book/pdf/21link.pdf

Slides
Many slides on spidering taken from UTexas Course, given by R. Mooney
https://www.cs.utexas.edu/~mooney/ir-course/
Some slides on Link Analysis taken from CMU-Qatar University IR course
https://github.com/joaopalotti/cmu_67300

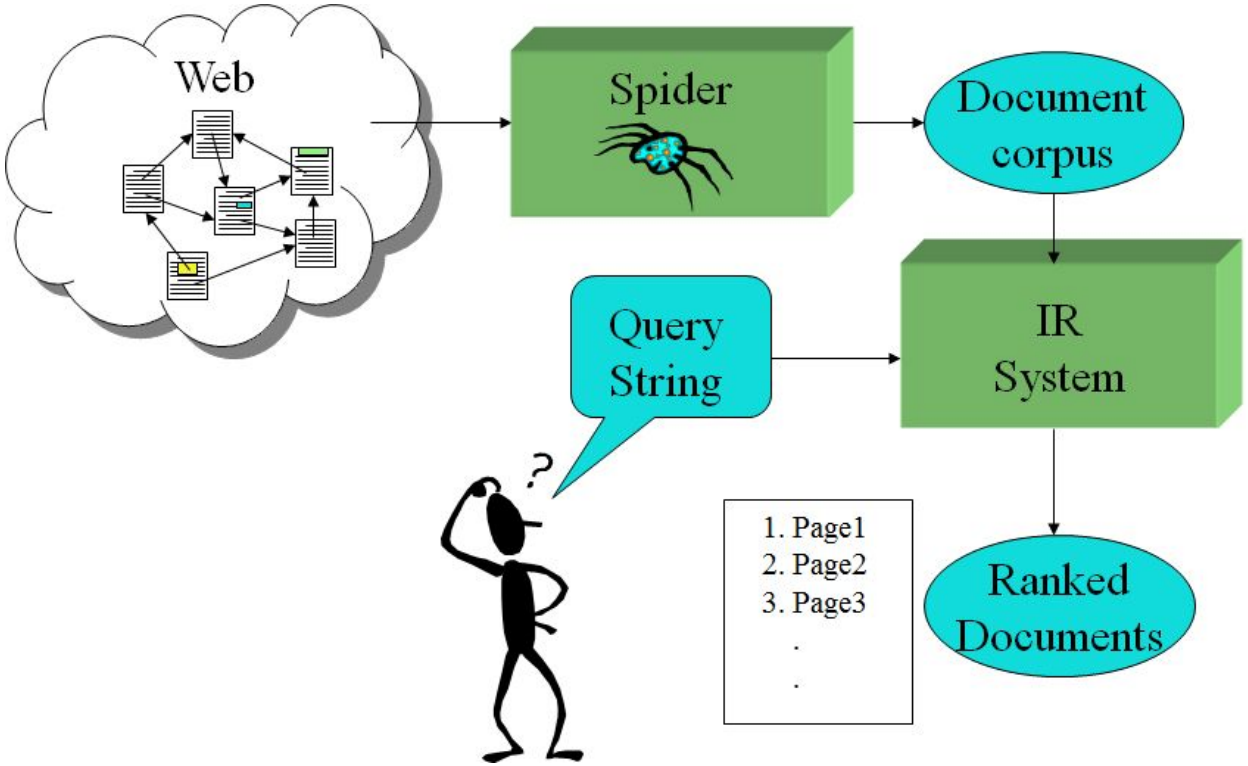
Information Retrieval on Internal Collection

Typical IR task



Web Searches

Finding documents on the Web

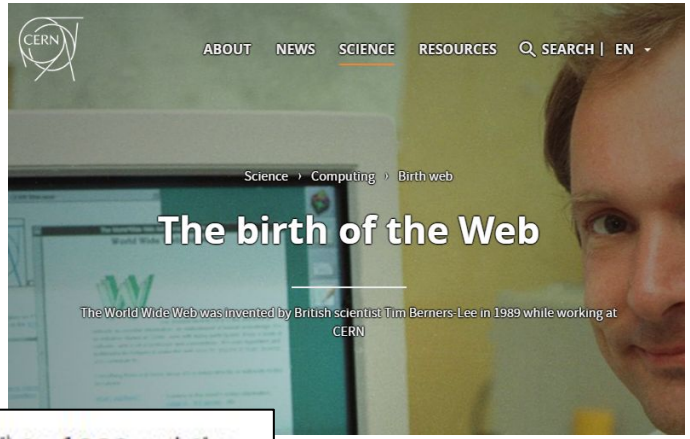


Characterizing the Web

Early development

World Wide Web

- Development in 1989 at CERN (European Organization for Nuclear Research) to organize documents available on the internet.
- Development of the early protocols, HTTP, URLs, and the first web server.



Tim Berners-Lee, a British scientist, invented the World Wide Web (WWW) in 1989, while working at CERN. The web was originally conceived and developed to meet the demand for automated information-sharing between scientists in universities and institutes around the world.

Early Search Engines (90s)

File Transfer Protocol

From Wikipedia, the free encyclopedia
(Redirected from Ftp)

"FTP" redirects here. For other uses, see FTP (disam

The **File Transfer Protocol** (**FTP**) is a standard network protocol used for the transfer of computer files between a client and server on a computer network.

Archie search engine

From Wikipedia, the free encyclopedia

Archie is a tool for indexing FTP archives, allowing people to find specific files. It is considered to be the first Internet search engine.[1] The original implementation was written in 1990 by Alan Emtage, then a postgraduate student at McGill University in Montreal, and Bill Heelan, who studied at Concordia University in Montreal and worked at McGill University at the same time.[2]

Early Search Engines (90s)

Gopher (protocol)

From Wikipedia, the free encyclopedia

The **Gopher** protocol /ˈgoʊfər/ is a TCP/IP application layer protocol designed for distributing, searching, and retrieving documents over the Internet. The Gopher protocol was strongly oriented towards a menu-document design and presented an alternative to the World Wide Web in its early stages, but ultimately Hypertext Transfer Protocol (HTTP) became the dominant protocol.

Veronica (search engine)

From Wikipedia, the free encyclopedia

Veronica was a search engine system for the Gopher protocol, released in November 1992[1] by Steven Foster and Fred Barrie at the University of Nevada, Reno.

Jughead (search engine)

From Wikipedia, the free encyclopedia

Jughead is a search engine system for the Gopher protocol. It is distinct from Veronica in that it searches a single server at a time.

Early Crawlers (90s)

World Wide Web Wanderer

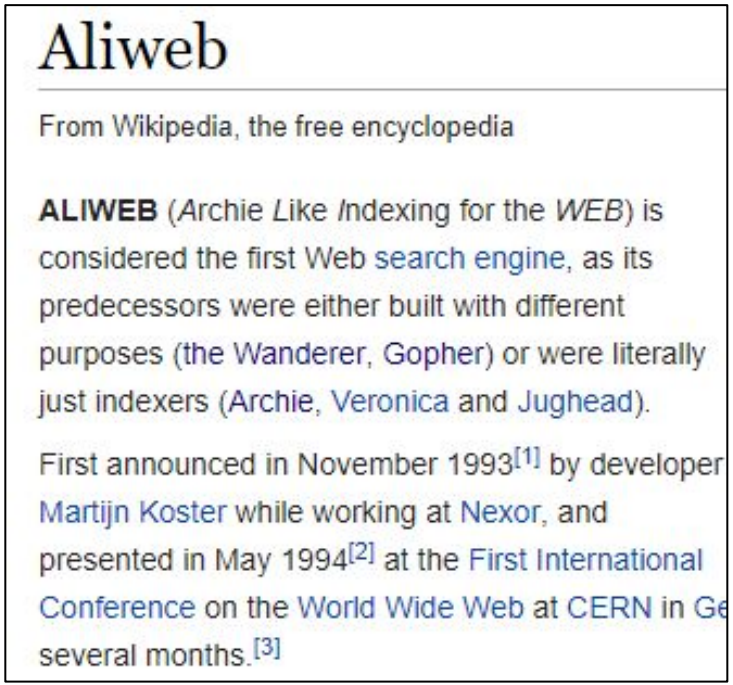
From Wikipedia, the free encyclopedia

World Wide Web Wanderer

Type of site	Web search engine
Launched	June 30, 1993; 25 years ago
Current status	Closed

The **World Wide Web Wanderer**, also referred to as just the **Wanderer**, was a Perl-based web crawler that was first deployed in June 1993 to measure the size of the World Wide Web. The Wanderer was developed at the Massachusetts Institute of Technology by Matthew Gray, who, as of 2017, has spent a decade as a software engineer at Google. The crawler was used to generate an index called the *Wandex* later in 1993. While the Wanderer was probably the first web robot, and, with its index, clearly had the potential to become a general-purpose WWW search engine, the author does not make this claim[1] and elsewhere[2] it is stated that this was not its purpose. The Wanderer charted the growth of the web until late 1995.

Early Crawlers (90s)



Aliweb

From Wikipedia, the free encyclopedia

ALIWEB (*A*rchie *L*ike *I*ndexing for the *WEB*) is considered the first Web search engine, as its predecessors were either built with different purposes (the Wanderer, Gopher) or were literally just indexers (Archie, Veronica and Jughead).

First announced in November 1993[1] by developer Martijn Koster while working at Nexor, and presented in May 1994[2] at the First International Conference on the World Wide Web at CERN in Ge several months.[3]

Early Search Model debate...

Taxonomy based (1996)

web.archive.org (to find older versions)

DMOZ (taxonomy)

Launched in 1998

Full-text search (1996)

Early Web development debate

TAXONOMY-BASED SEARCH

- **Hand-made hierarchies of topics**
- **Organized way to find information**
- **Maintenance is hard**
- **Does not scale well**
- **Much navigation to find documents**

FULL-TEXT INDEXING

- **Building inverted indexes**
- **Easy to use**
- **No need for external categorization**
- **Exposed to spammers**
  - **Any inclusion of additional words in a page changes the results**

Early Web development debate

Who won??

TAXONOMY-BASED SEARCH

- **Hand-made hierarchies of topics**
- **Organized way to find information**
- **Maintenance is hard**
- **Does not scale well**
- **Much navigation to find documents**

FULL-TEXT INDEXING

- **Building inverted indexes**
- **Easy to use**
- **No need for external categorization**
- **Exposed to spammers**
  - **Any inclusion of additional words in a page changes the results**

DMOZ (taxonomy)

Early commercial full-text search engines

- In 1998, Larry Page and Sergey Brin, Ph.D. students at Stanford, started Google. Main advance is use of *link analysis* to rank results.

- Microsoft launched MSN Search in 1998 based on Inktomi (started from UC Berkeley in 1996), changed to Live Search in 2007, and Bing in 2009.

Early commercial full-text search engines

- In 1998, La[illegible]ry Page and Sergey Brin, Ph.D. students at Stanford, started Google. M[illegible]n advance is use of *link analysis* to rank results.

- Micros[illegible]aunched MSN Search in 1998 b[illegible]ed on Inktomi (started from UC Be[illegible]ey in 1996), changed to Live Sear[illegible]in 2007, and Bing in 2009.

Business Model?

How does that work?

Business model... advertisement!

Early advertisement: Cost Per Click

- Highest bid for this search was $0.30
- The website owner has to pay $0.30 per time a user clicks on this link (Cost Per Click = CPC)
- Pages were ranked by bid value – No TF-IDF, no VSM, no LM, no BM25 anymore…

Early advertisement: Cost Per Click

- Highest bid for this search was $0.30
- The website owner has to pay $0.30 per time a user clicks on this link (Cost Per Click = CPC)
- Pages were ranked by bid value – No TF-IDF, no VSM, no LM, no BM25 anymore…

No more CSI4107 course!!
Just need $$$$!!!

Split "paid sites" / "natural sites"

Advertisers bid for “keywords”. Ads for highest bidders displayed when user query contains a purchased keyword.

Leads to scams... Company buying keywords and reselling them at a higher price!

Expensive keywords...

Side banners

Advertisers pay for banner ads on the site that do not depend on a user's query.

- CPM: Cost Per Mille (thousand impressions). Pay for each ad display.

Side banners

Advertisers pay for banner ads on the site that do not depend on a user's query.

- CPM: Cost Per Mille (thousand impressions). Pay for each ad display.
- CPC: Cost Per Click. Pay only when user clicks on ad.

Side banners

Advertisers pay for banner ads on the site that do not depend on a user's query.

- CPM: Cost Per Mille (thousand impressions). Pay for each ad display.
- CPC: Cost Per Click. Pay only when user clicks on ad.
- CTR: Click Through Rate. Fraction of ad impressions that result in clicks throughs.

Side banners

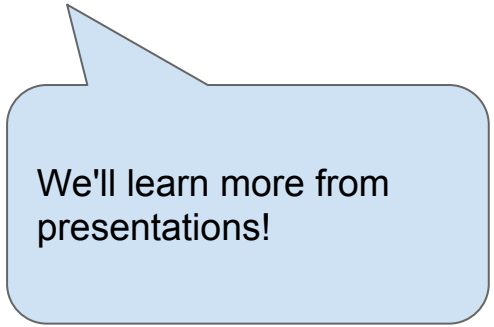
Advertisers pay for banner ads on the site that do not depend on a user's query.

- CPM: Cost Per Mille (thousand impressions). Pay for each ad display.
- CPC: Cost Per Click. Pay only when user clicks on ad.
- CTR: Click Through Rate. Fraction of ad impressions that result in clicks throughs.
- CPA: Cost Per Action (Acquisition). Pay only when user actually makes a purchase on target site.

Today's advertisement: Bidding

Advertisers BID for banner ads, with complex bidding algorithms:

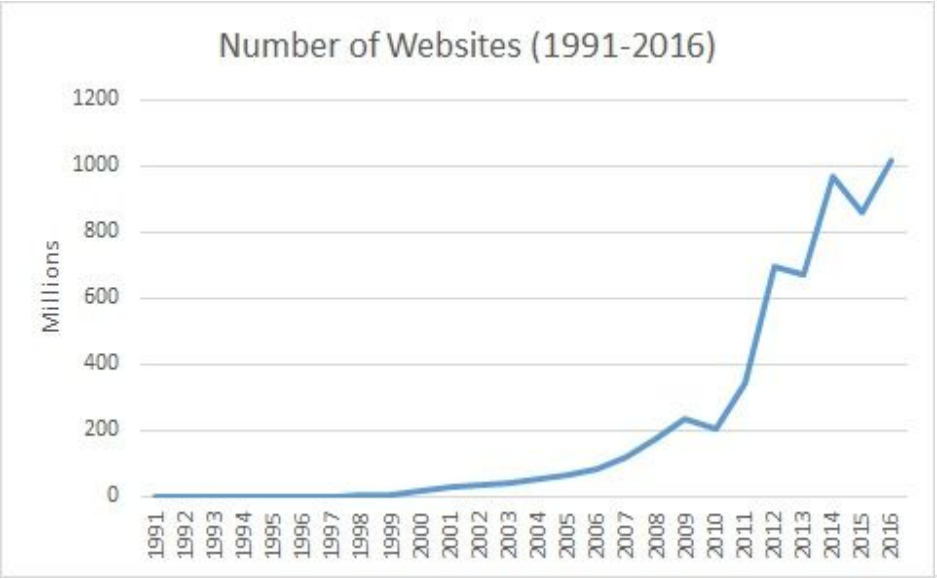
- behavioral advertisement
- semantic advertisement

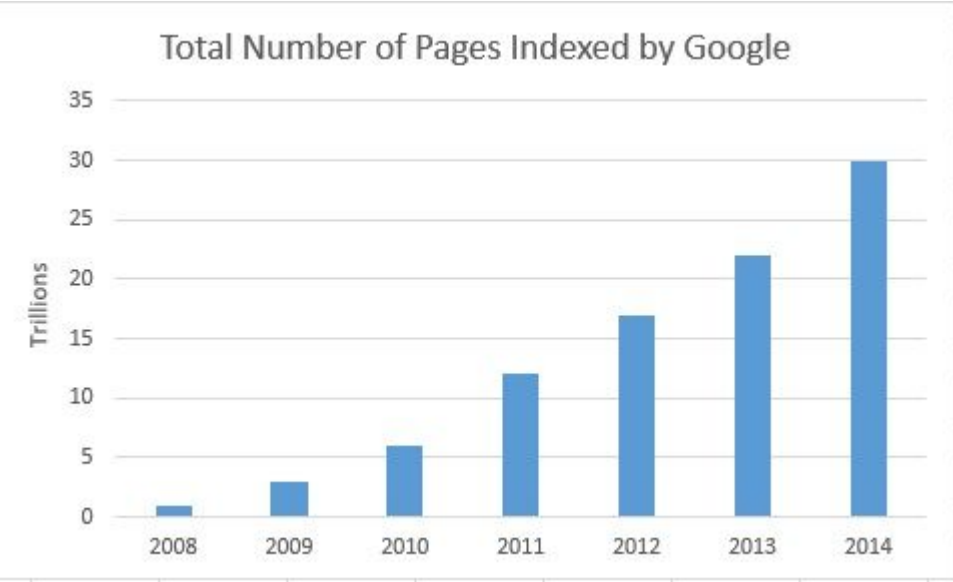


- Search engine company gets revenue every time somebody clicks on an ad (>90% Google's revenue)

Web size... growth...

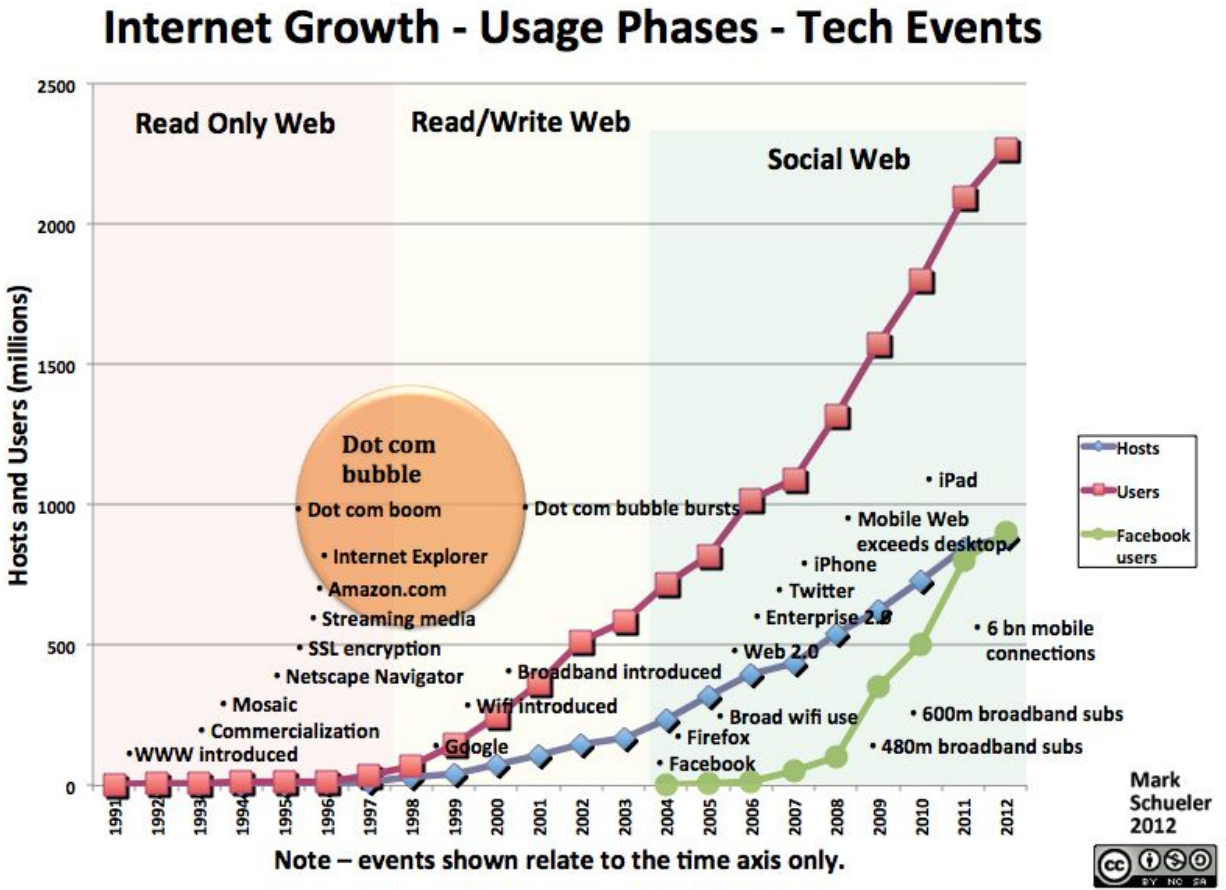
Growing Web





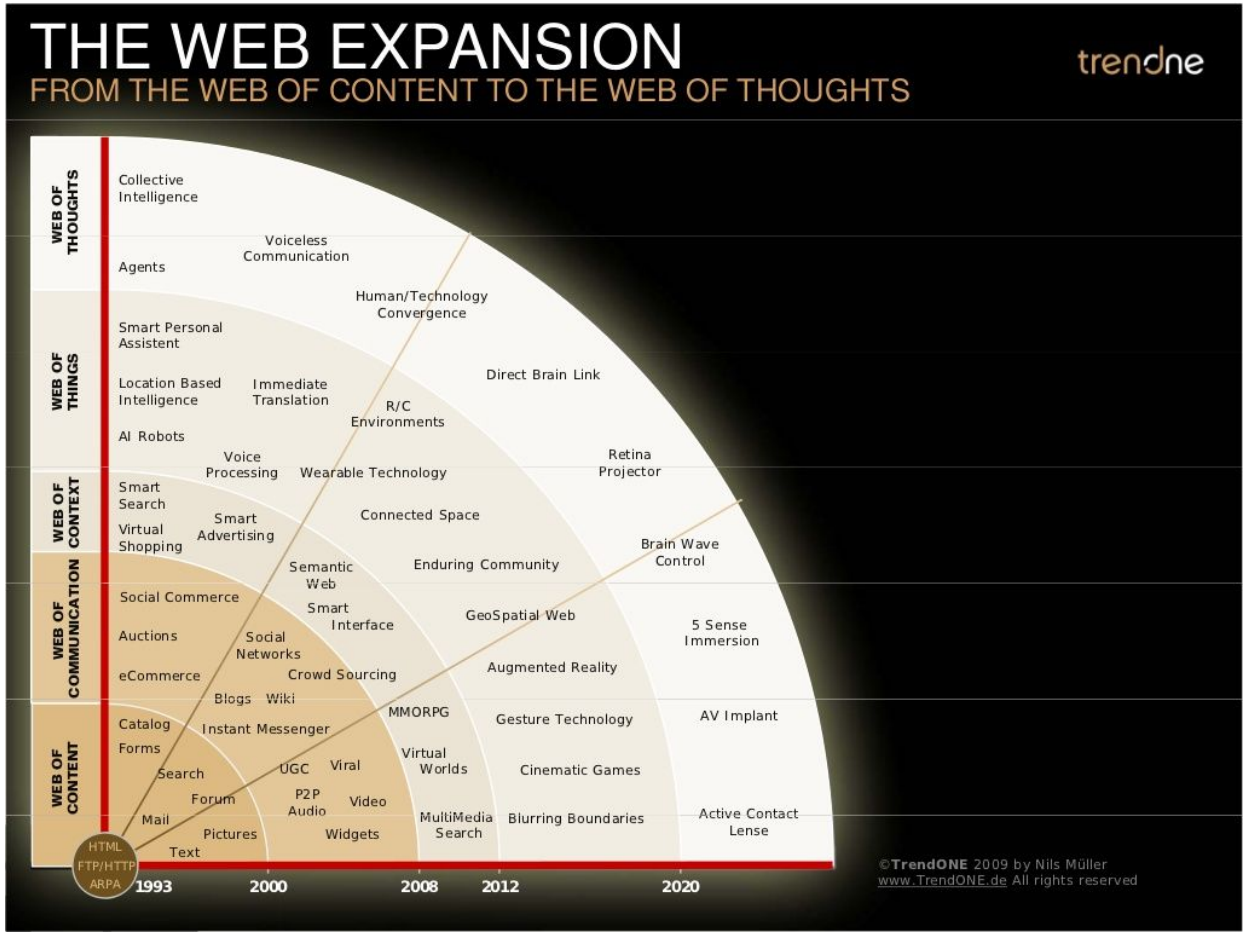
Source: https://www.scribblrs.com/increase-number-pages-indexed-google-years/

Growing Web



Source: http://wstweb1.ecs.soton.ac.uk/web-observatory/about/tracking-explosive-growth/

View of the future



https://www.slideshare.net/vladsitnikov/trend-one-web-expansion-grape-online-strategies-2009-by-nick-sohnemann/27-THE_WEB_EXPANSIONFROM_THE_WEB

Web Challenges for IR

Web Challenges for IR

- Large Volume:

  Billions of separate documents.
- Distributed Data:

  Documents spread over millions of different web servers.
- Volatile Data:

  Many documents change or disappear rapidly (e.g. dead links).

Web Challenges for IR

- Unstructured and Redundant Data:

  No uniform structure, HTML errors, up to 30% (near) duplicate documents.
- Quality of Data:

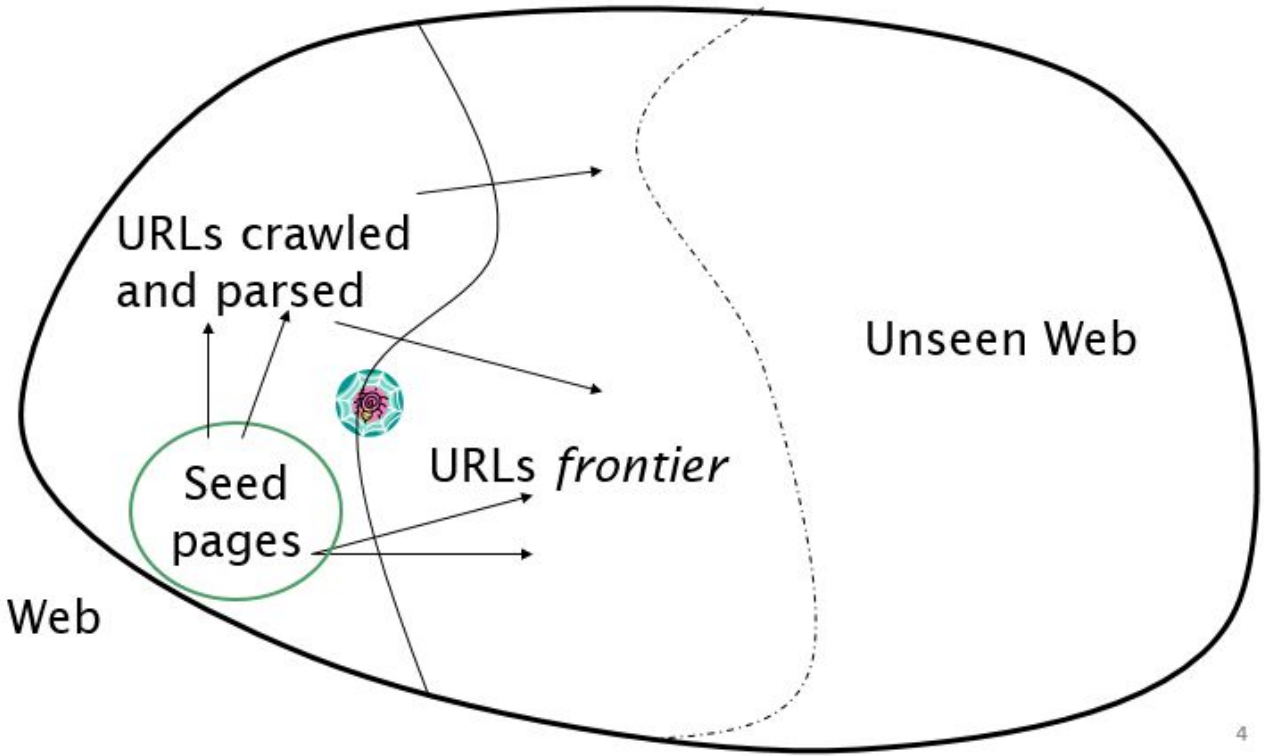
  No editorial control, false information, poor quality writing, typos, etc.
- Heterogeneous Data:

  Multiple media types (images, video, VRML), languages, character sets, etc.

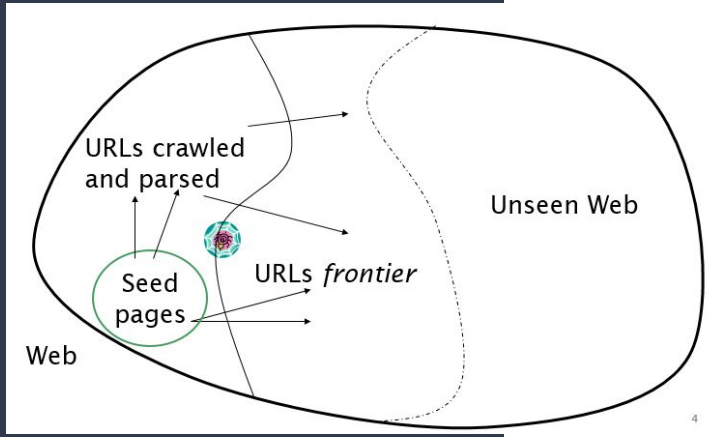
Web crawling

Crawler overview

Crawling picture



Spiders / Crawlers



- Start with a comprehensive set of root URLs from which to start the search.
- Follow all links on these pages recursively to find additional pages.
- Index all novel found pages in an inverted index as they are encountered.

Link Extraction + IP search

1) Find all links in a page and extract URLs.

<a href="http://www.cs.utexas.edu/users/mooney/ir-course">

2) Complete relative URL's using current page URL:

<a href="proj3"> to http://www.cs.utexas.edu/users/mooney/ir-course/proj3

<a href="../cs343/syllabus.html"> to http://www.cs.utexas.edu/users/mooney/cs343/syllabus.html

3) Use DNS (Domain Name Server) to locate IP address

Crawlers Design Rules

- Be Robust: Be immune to spider traps and other malicious behavior from web servers
- Be Polite: Respect implicit and explicit politeness considerations

Crawlers politeness

- Explicit politeness: specifications from webmasters on what portions of site can be crawled
  - robots.txt
- Implicit politeness: even with no specification, avoid hitting any site too often

Robots Exclusion Protocol

Site administrator puts a “robots.txt” file at the root of the host’s web directory.

http://www.ebay.com/robots.txt

http://www.cnn.com/robots.txt

File is a list of excluded directories for a given robot (user-agent).

Exclude all robots from the entire site:

```
User-agent: *
Disallow: /
```

Examples of Robot Exclusion Protocol

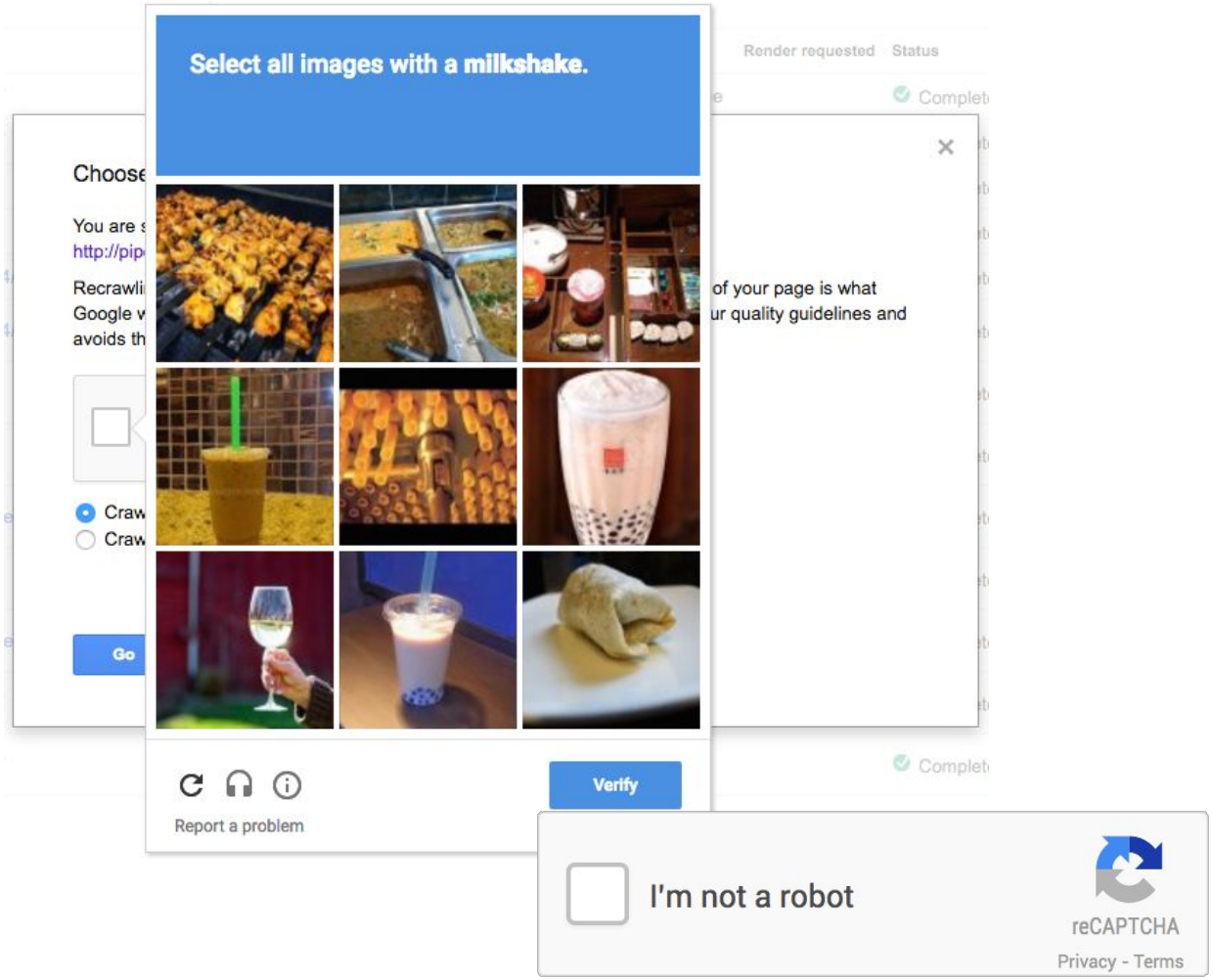
Exclude specific directories:

```
User-agent: *
Disallow: /tmp/
Disallow: /cgi-bin/
Disallow: /users/paranoid/
```

Allow/Exclude a specific robot:

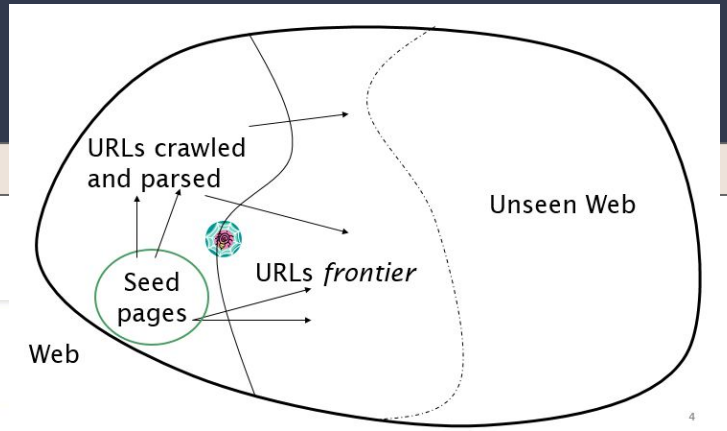
```
User-agent: KnownBot
Disallow:
User-agent: OtherBot
Disallow: /
```

Robot or human search



Crawling strategy

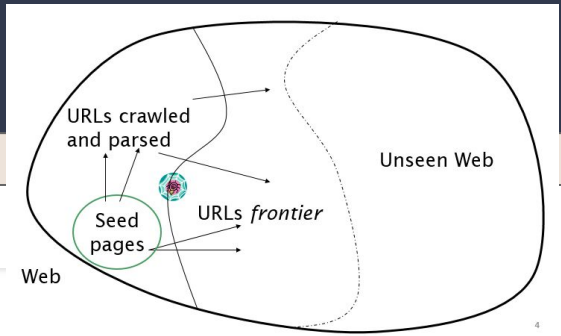
Spidering Algorithm



- Pick a URL from the frontier
- Fetch the document at the URL
- Parse the URL
  - Extract links from it to other docs (URLs)
- Check if URL has content already seen
  - If not, add to indexes
- For each extracted URL
  - Ensure it passes certain URL filter tests

E.g., only crawl .edu, obey robots.txt, etc.

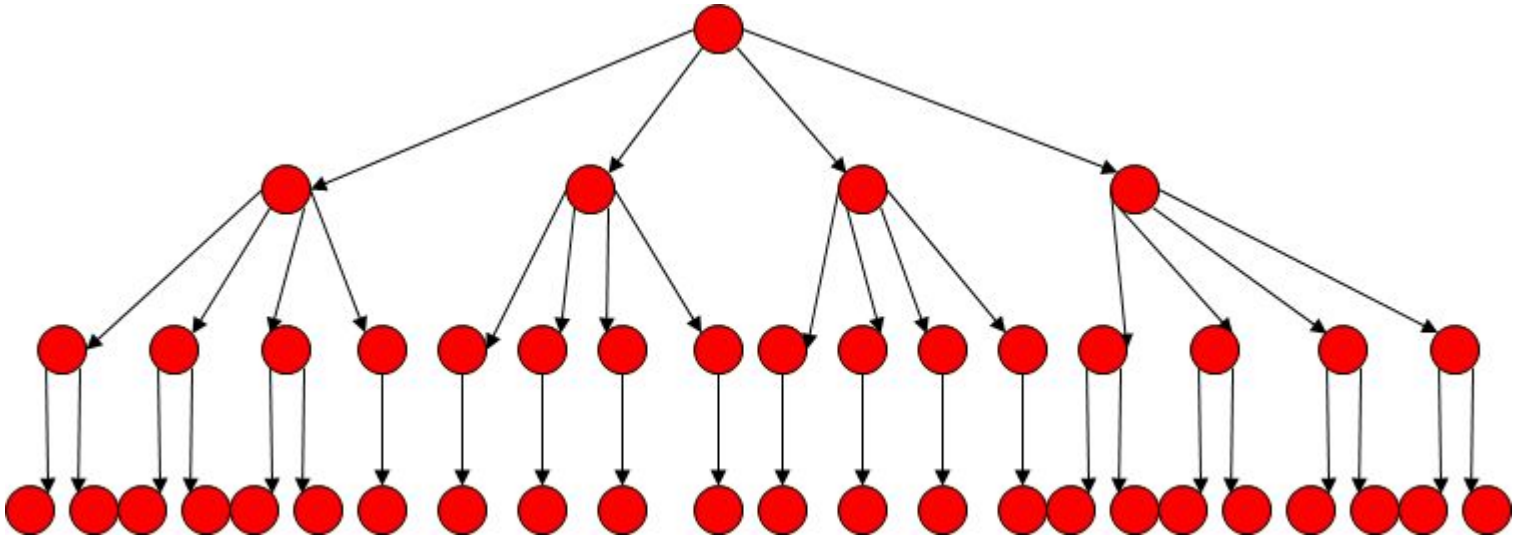
Spidering Algorithm



- Pick a URL from the frontier ← Which one?
- Fetch the document at the URL
- Parse the URL
  - Extract links from it to other docs (URLs)
- Check if URL has content already seen
  - If not, add to indexes
- For each extracted URL
  - Ensure it passes certain URL filter tests

E.g., only crawl .edu, obey robots.txt, etc.

Breadth-first Search // Depth-first Search



Search Strategy Trade-Offs

- Breadth-first explores uniformly outward from the root page but requires memory of all nodes on the previous level (exponential in depth). Standard spidering method.

- Depth-first requires memory of only depth times branching-factor (linear in depth) but gets “lost” pursuing a single thread.

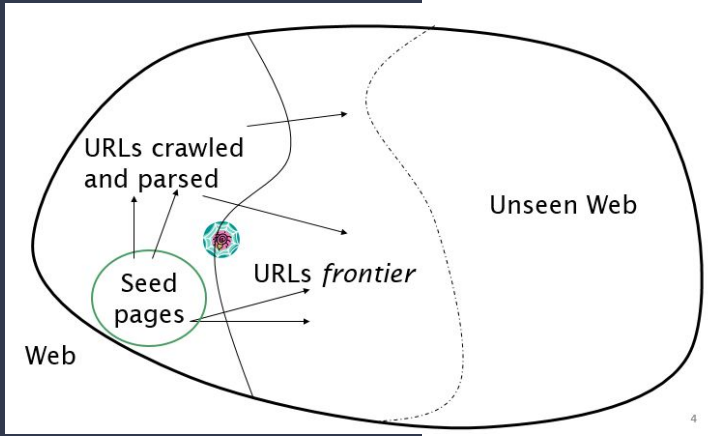
- Both strategies implementable using a queue of links (URL’s).

Multi-Threaded Spidering

- Bottleneck is network delay in downloading individual pages.
- Best to have multiple threads running in parallel each requesting a page from a different host.
- Distribute URLs to threads to guarantee equitable distribution of requests across different hosts to maximize throughput and avoid overloading any single server.
- Early Google spider had multiple coordinated crawlers with about 300 threads each, together able to download over 100 pages per second.

Web is dynamic

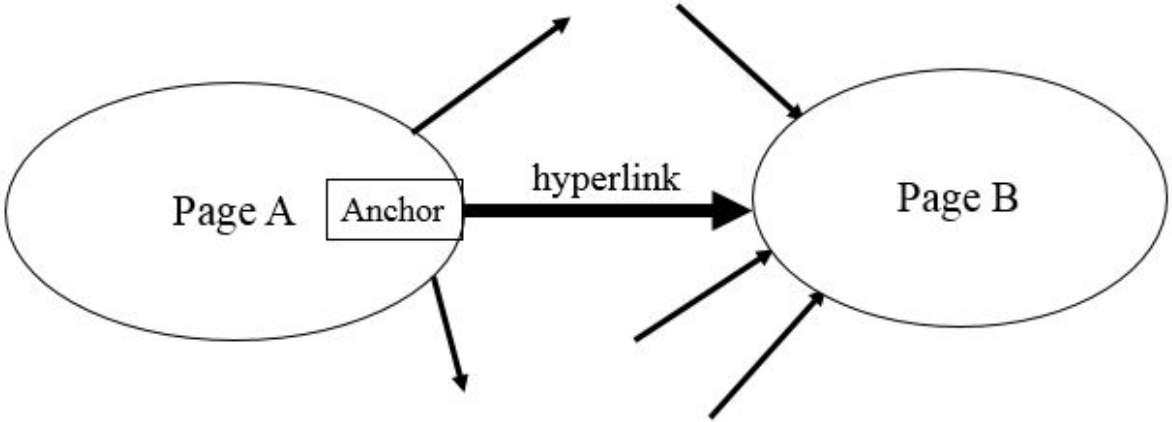
- Freshness: crawl some pages more often than others
  - E.g., pages (such as News sites) whose content changes often



But... this might conflict with being polite! Many sites point to themselves...

Link Analysis

The Web as a Directed Graph



Hypothesis 1: A hyperlink between pages denotes a conferral of authority (quality signal)

Hypothesis 2: The text in the anchor of the hyperlink on page A describes the target page B

Anchor texts

Can the anchor text be used?

- Journal of the ACM
- Good site to buy computers and related stuff
- We all fear the Big Blue
- I just found this amazing Internet Portal

Anchor text indexing

- Journal of the ACM
- Good site to buy computers and related stuff
- We all fear the Big Blue
- I just found this amazing Internet Portal

"computer" ⋯⋯▶ "ibm.com"

"big" ⋯⋯▶ "ibm.com"

"blue" ⋯⋯▶ "ibm.com"

Anchor text indexing

Not all anchor text is useful...

There is a good discussion of traffic rules in Qatar <a>here.

IDF can be used to decrease the importance of common words, such as "here", "click", "link", "site"

AnchorText
click here
read more
more info here
about the author
additional info
check this out
this website
the page here
over here
over there
here
this page

Generic Anchor texts

Anchor text indexing observations

- IBM can be found searching for its nickname "Big Blue", even though there is no single mention of "big blue" in its website
- Anchor text is often a better description of a page' s content than the page itself
- Allows orchestrated campaigns against specific sites:
  - Google Bomb – Queries like "evil empire", "who is a failure"

Anchor text indexing observations

Search Engines do put a significant weight on the anchor text, in conjunction with the document terms.

Source: http://cultofweb.com/blog/seo-friendly-anchor-text/

PageRank

PageRank

PageRank

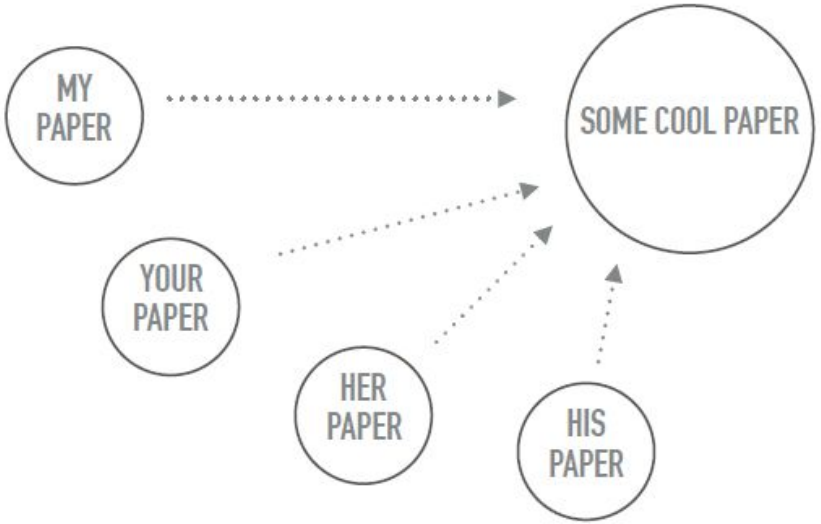
From Wikipedia, the free encyclopedia

PageRank (**PR**) is an algorithm used by Google Search to rank web pages in their search engine results. PageRank was named after Larry Page,[1] one of the founders of Google. PageRank is a way of measuring the importance of website pages. According to Google:

> PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites.[2]

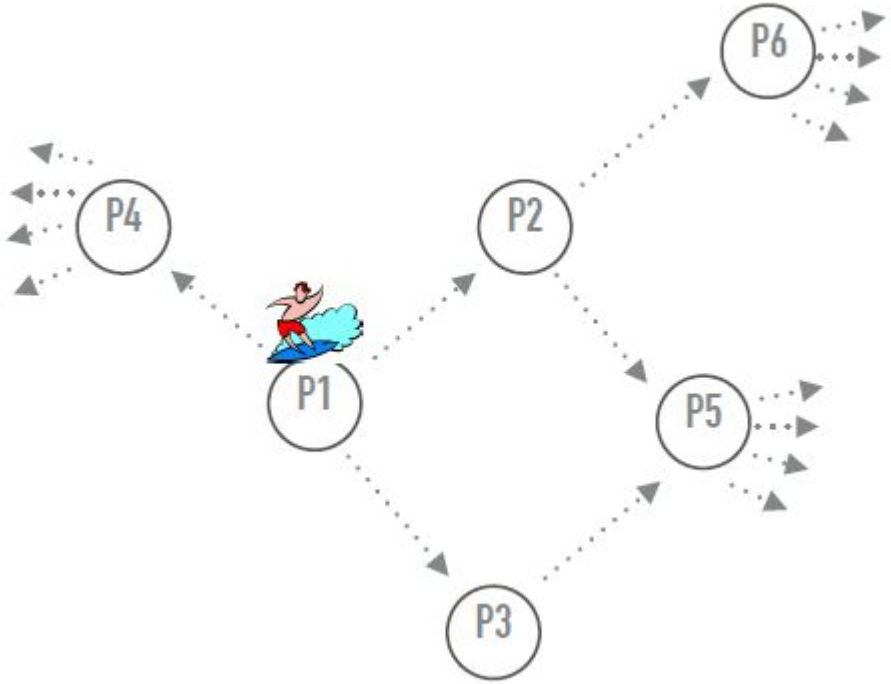
Currently, PageRank is not the only algorithm used by Google to order search results, but it is the first algorithm that was used by the company, and it is the best known.[3][4]

Inspiration from citation analysis



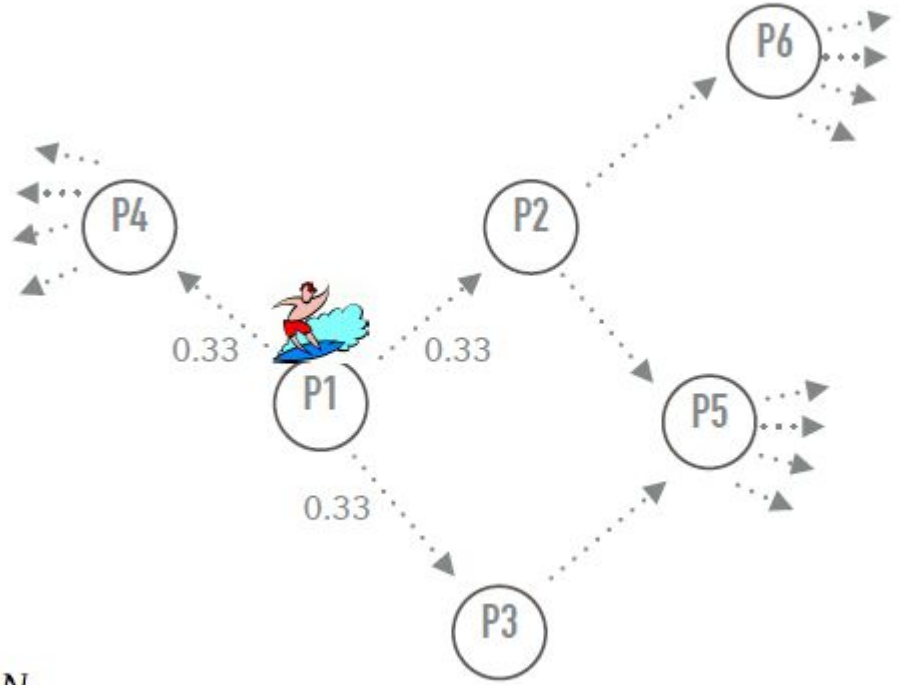
Web surfer

Random Walking on the Web



Web surfer

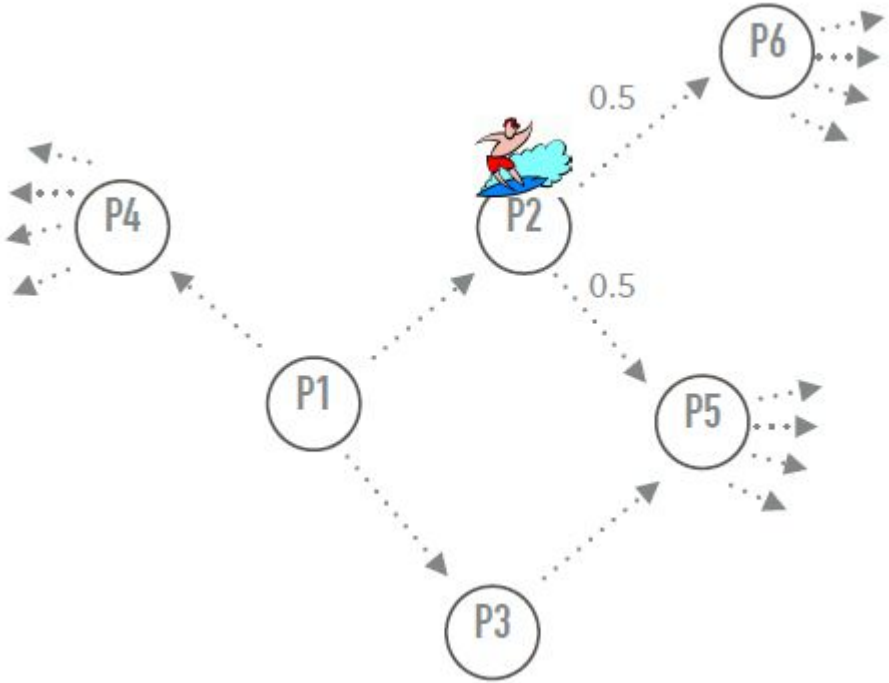
Random Walking on the Web



$$\forall i, \sum_{j=1}^{N} P_{ij} = 1.$$

Web surfer

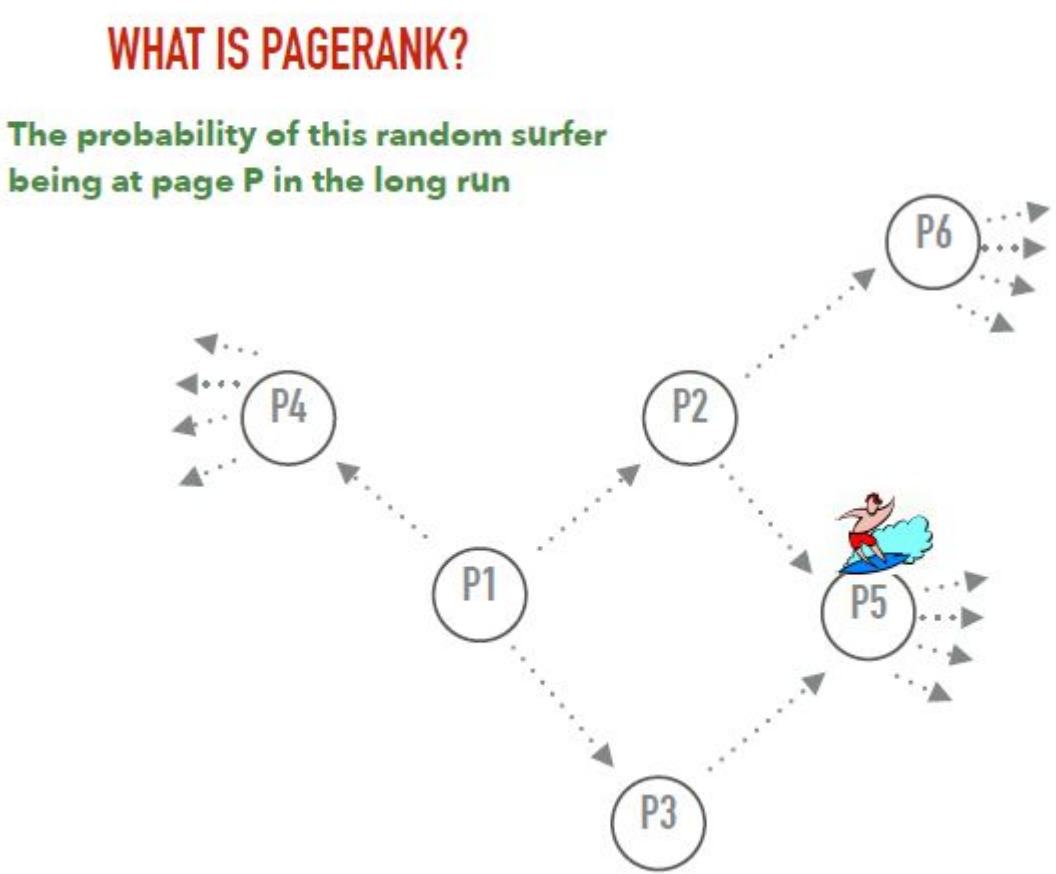
Random Walking on the Web



$$\forall i, \sum_{j=1}^{N} P_{ij} = 1.$$

Web surfer

Random Walking on the Web



PageRank

- Considerations:
  - A page that has many in-links has a higher probability of being visited
  - Pages linked by popular pages have higher probability of being visited
  - What if a page is a dead end?

PageRank

- Considerations:
  - A page that has many in-links has a higher probability of being visited
  - Pages linked by popular pages have higher probability of being visited
  - What if a page is a dead end?

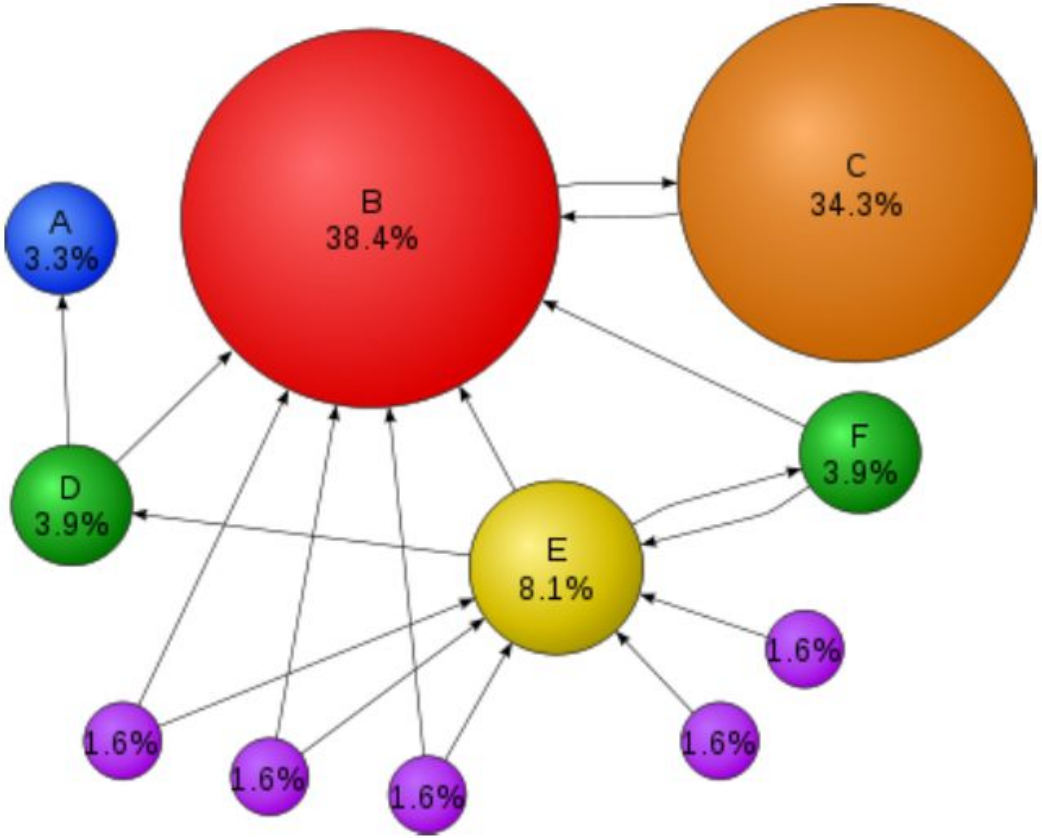
TELEPORTING!

PageRank - Example: teleportation rate of 10%

- At a dead end, surfer jumps to a random page with probability 1/N
- At a non-dead end:
  - with probability 10%, jumps to a random web page
  - with probability 90%, go out on a random regular link

Example of PageRank result

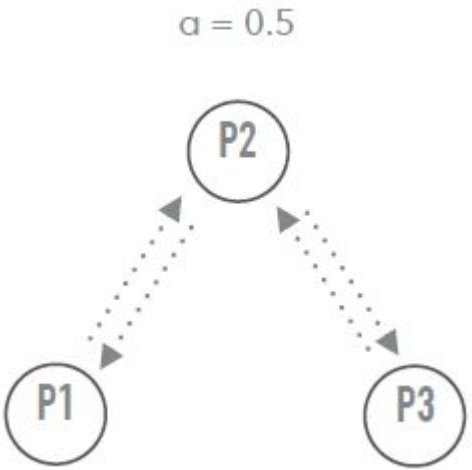
But.... how do we get there???



PageRank Algorithm - Matrix initialization

1. If a row of A has no 1's, then replace each element by 1/N. For all other rows proceed as follows.
2. Divide each 1 in A by the number of 1's in its row.
3. Multiply the resulting matrix by (1-α)
4. Add α/N to every entry of the resulting matrix, to obtain P.

Let's do another example in class.



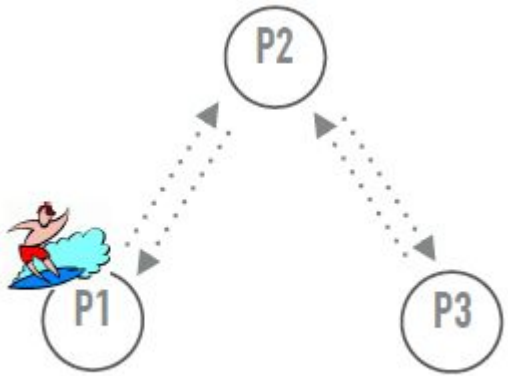
PageRank Algorithm - Markov Chain Property

PROPERTY OF A MARVOK CHAIN

$$PR_t(d) = PR_{t-1}(d) * M$$

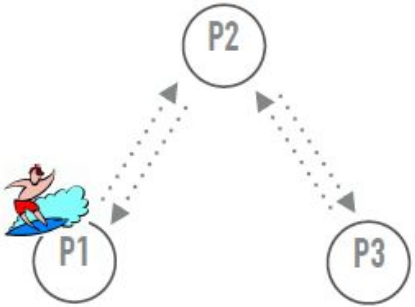
$$PR = \begin{bmatrix} X & Y & Z \end{bmatrix}$$

Page Rank Algorithm - Iterative Process Initialization



$$PR = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$ Arbitrarily chosen. Does not matter!

Page Rank Algorithm - Iterative Process



This was the result of the initialization step.

$$PR_t(d) = PR_{t-1}(d) * M$$

$$PR(1) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1/6 & 2/3 & 1/6 \\ 5/12 & 1/6 & 5/12 \\ 1/6 & 2/3 & 1/6 \end{bmatrix} = \begin{bmatrix} 1/6 & 2/3 & 1/6 \end{bmatrix}$$

Page Rank Algorithm - Iterative Process

$$PR_t(d) = PR_{t-1}(d) * M$$

$$PR(1) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1/6 & 2/3 & 1/6 \\ 5/12 & 1/6 & 5/12 \\ 1/6 & 2/3 & 1/6 \end{bmatrix} = \begin{bmatrix} 1/6 & 2/3 & 1/6 \end{bmatrix}$$

$$PR_t(d) = PR_{t-1}(d) * M$$

$$PR(2) = \begin{bmatrix} 1/6 & 2/3 & 1/6 \end{bmatrix} \begin{bmatrix} 1/6 & 2/3 & 1/6 \\ 5/12 & 1/6 & 5/12 \\ 1/6 & 2/3 & 1/6 \end{bmatrix} = \begin{bmatrix} 1/3 & 1/3 & 1/3 \end{bmatrix}$$

Page Rank Algorithm - Iterative Process

$$PR_t(d) = PR_{t-1}(d) * M$$

$$PR(2) = \begin{bmatrix} 1/6 & 2/3 & 1/6 \end{bmatrix} \begin{bmatrix} 1/6 & 2/3 & 1/6 \\ 5/12 & 1/6 & 5/12 \\ 1/6 & 2/3 & 1/6 \end{bmatrix} = \begin{bmatrix} 1/3 & 1/3 & 1/3 \end{bmatrix}$$

$\vec{x}_0$	1	0	0
$\vec{x}_1$	1/6	2/3	1/6
$\vec{x}_2$	1/3	1/3	1/3
$\vec{x}_3$	1/4	1/2	1/4
$\vec{x}_4$	7/24	5/12	7/24
...	...	...	...
$\vec{x}$	5/18	4/9	5/18

Using PageRank

Remember Query likelihood Model

Retrieval question:

WHAT IS THE MOST LIKELY DOCUMENT THAT GENERATED THIS QUERY?

Query

"capital arabic countries"

Query Likelihood Model

Ranking of documents

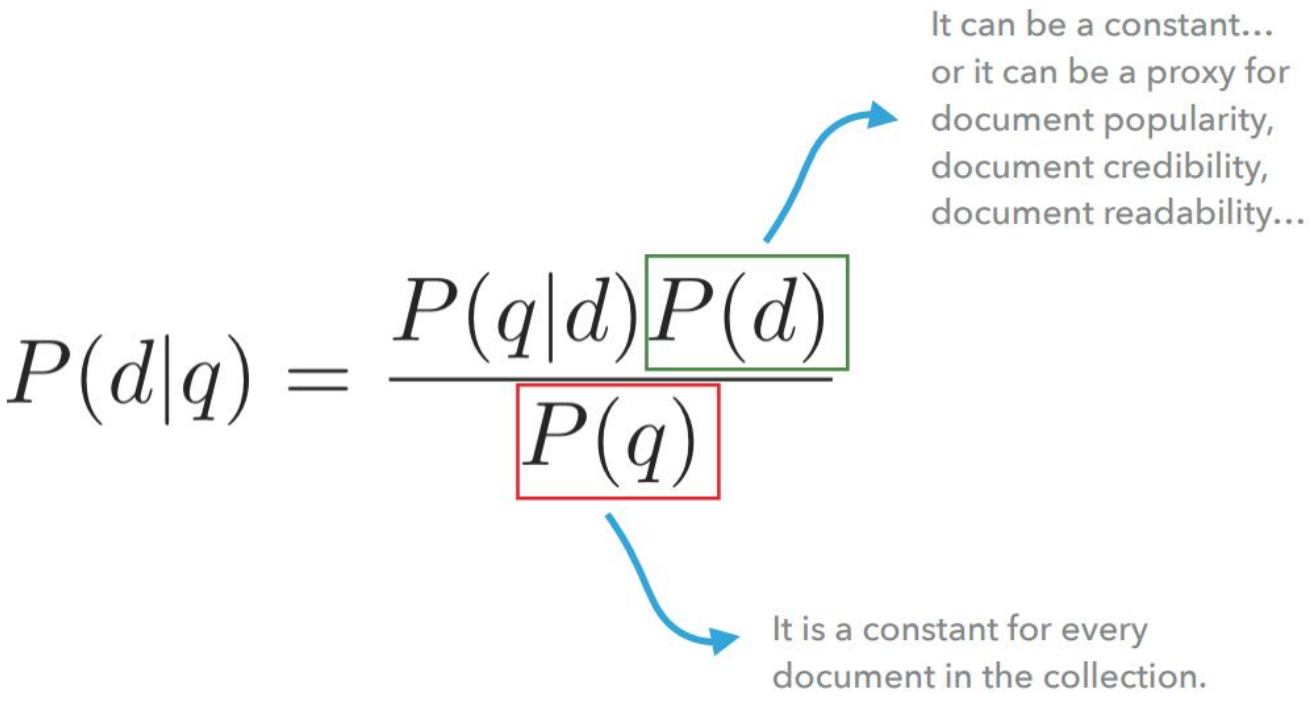
Probability of a document given a query

Bayes Rule

$$P(d|q) = \frac{P(q|d)P(d)}{P(q)}$$

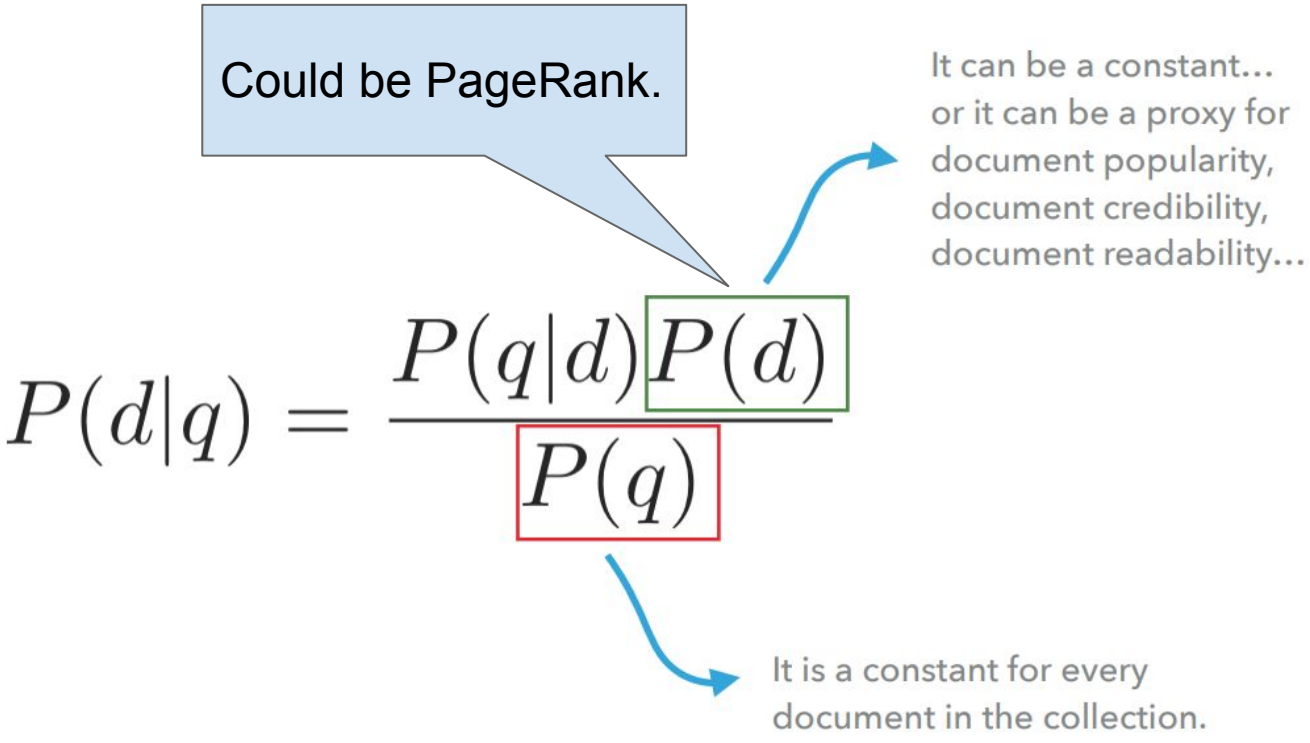
Query Likelihood Model

Bayes Rule



Query Likelihood Model

Bayes Rule



Query Likelihood Model

Linear Interpolation Smoothing

- Estimate conditional probabilities $\mathrm{P}(X_i \mid Y)$ as a mixture of conditioned and unconditioned estimates:

$$P(X_i \mid Y) = \lambda \hat{P}(X_i \mid Y) + (1 - \lambda)\hat{P}(X_i)$$

Query Likelihood Model

Calculate with smoothing

λ = 0.8

ϕ_1 = (red = 4, green = 6, automobile = 2, flower = 2, transit = 2, house = 2, tulip = 1, rose = 1)

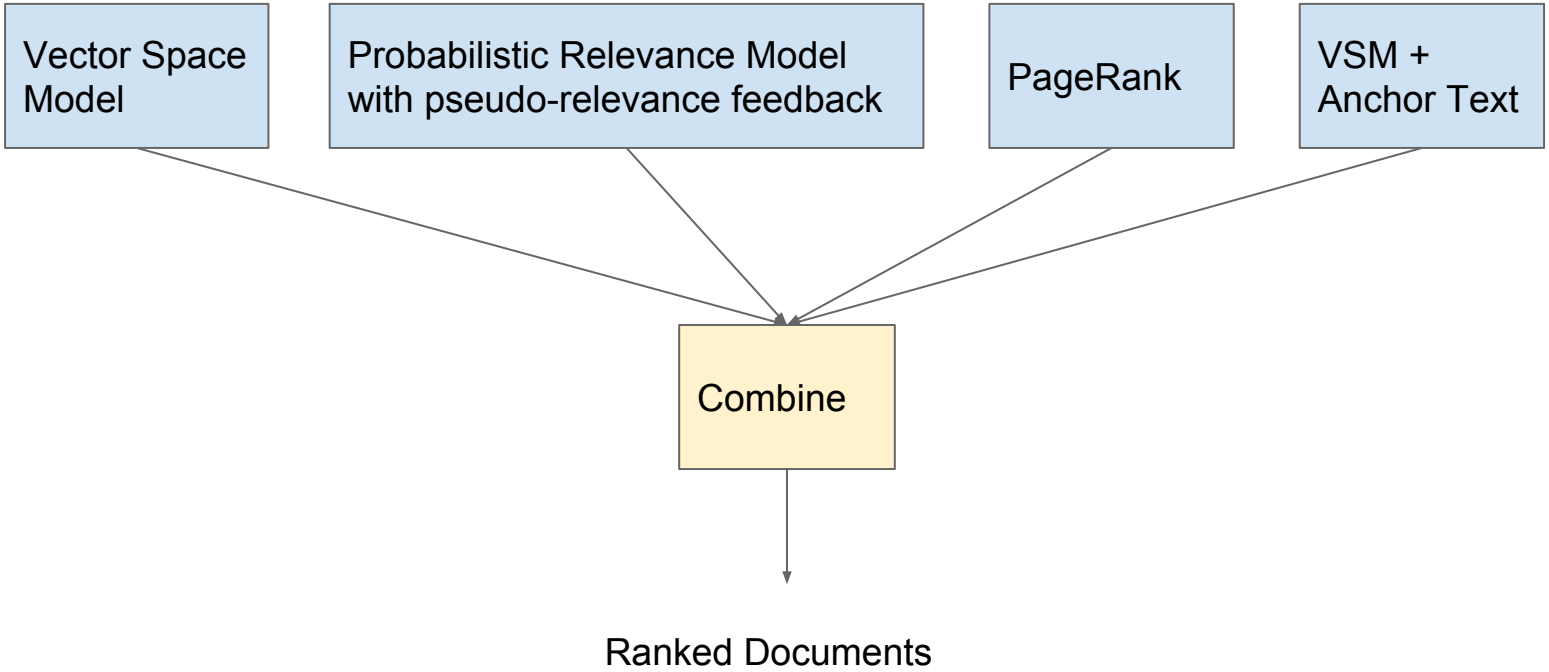
ϕ_2 = (red = 4, green = 4, automobile = 2, flower = 12, transit = 2, house = 8, tulip = 4, rose = 4)

ϕ_3 = (apple = 4, orange = 2, fruit = 2, red = 1, green = 1)

$P(\phi_3 \mid \text{red rose}) \propto P(\text{red rose} \mid \phi_3) * P(\phi_3)$

Could be PageRank.

What about a joint model search engine?



Google Ranking

Google ranking includes (based on university publications prior to commercialization).

- Vector-space similarity component
- Keyword proximity component
- HTML-tag weight component (e.g. title preference)
- PageRank component

Details of current commercial ranking functions are trade secrets.

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 was one of the primary reasons for Google’s initial success.

In summary….

What have you learned ?

Vocabulary

- Write down all the vocabulary words learned

Descriptive Information

- Advertisement Business Model
- Web challenges for IR

Comparative information

- Web crawling approaches (breadth-first / depth-first)
- Early web taxonomy based search versus full-text search

Algorithms

- PageRank
- Query Likelihood with PageRank as Prior estimator

Quiz 4

What to expect for Quiz 4 ?

Closed book. Calculator necessary.
True/False. Word/Definition associations. Short answers.
Algorithm Demonstration.

Research topics
- *Vocabulary words*

The WWW *(Short answers and T/F)*
- Crawling approaches
- Issues of WWW for IR

Link Analysis
- Anchor Text Indexing
- PageRank *(Algorithm Demonstration)*

Retrieval Model
- Query Likelihood with PageRank