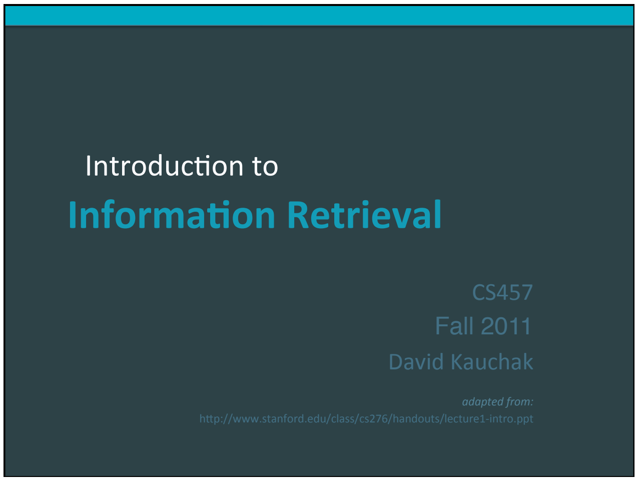


Introduction to

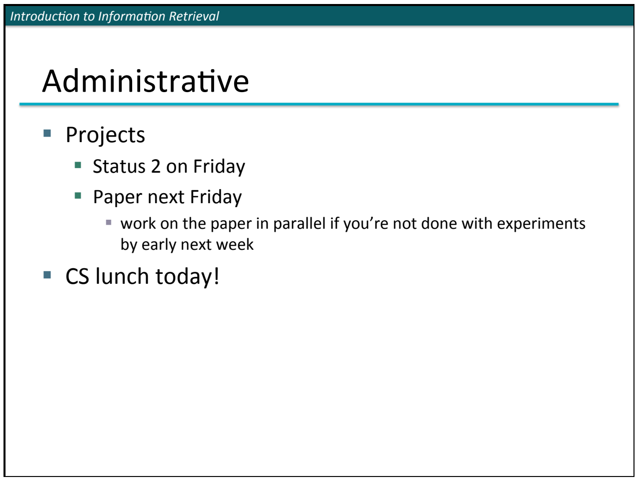
Information Retrieval

CS457

Fall 2011

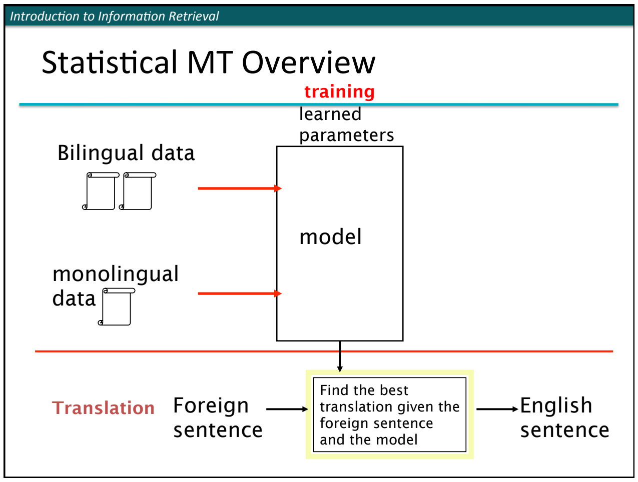
David Kauchak

adapted from:
http://www.stanford.edu/class/cs276/handouts/lecture1-intro.ppt



Administrative

- Projects
  - Status 2 on Friday
  - Paper next Friday
    - work on the paper in parallel if you're not done with experiments by early next week
- CS lunch today!



Statistical MT Overview

training

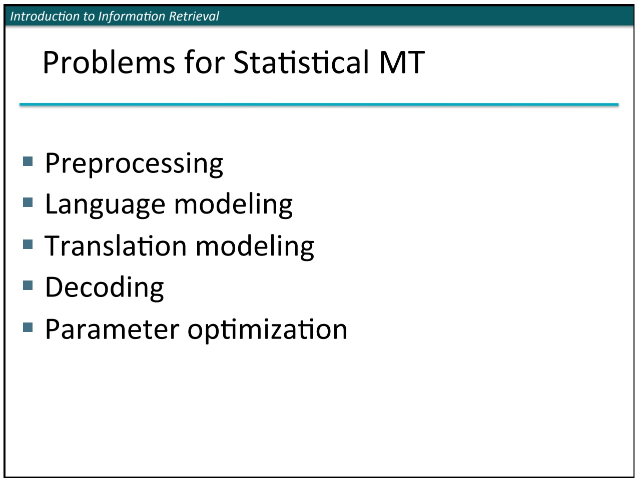
Bilingual data

monolingual data

learned parameters

model

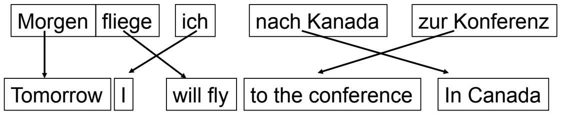
Translation Foreign sentence → Find the best translation given the foreign sentence and the model → English sentence



Problems for Statistical MT

- Preprocessing
- Language modeling
- Translation modeling
- Decoding
- Parameter optimization

Phrase-Based Statistical MT



Morgen fliege ich nach Kanada zur Konferenz

Tomorrow I will fly to the conference In Canada

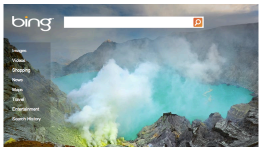
- Foreign input segmented in to phrases
  - "phrase" is any sequence of words
- Each phrase is probabilistically translated into English
  - P(to the conference | zur Konferenz)
  - P(into the meeting | zur Konferenz)
- Phrases are probabilistically re-ordered
- See [Koehn et al, 2003] for an intro.

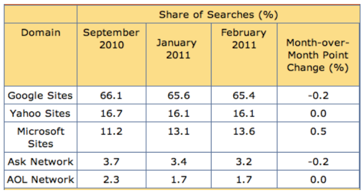
Information retrieval (IR)

- What comes to mind when I say "information retrieval"?

- Where have you seen IR? What are some real-world examples/uses?
  - Search engines
  - File search (e.g. OS X Spotlight, Windows Instant Search, Google Desktop)
  - Databases?
  - Catalog search (e.g. library)
  - Intranet search (i.e. corporate networks)

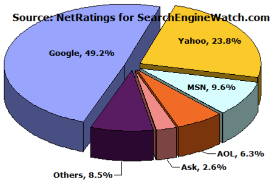
Web search



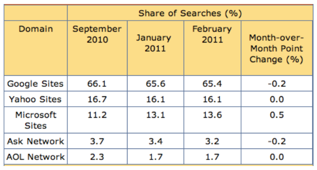


	Share of Searches (%)			
Domain	September 2010	January 2011	February 2011	Month-over-Month Point Change (%)
Google Sites	66.1	65.6	65.4	-0.2
Yahoo Sites	16.7	16.1	16.1	0.0
Microsoft Sites	11.2	13.1	13.6	0.5
Ask Network	3.7	3.4	3.2	-0.2
AOL Network	2.3	1.7	1.7	0.0

Web search


Source: NetRatings for SearchEngineWatch.com

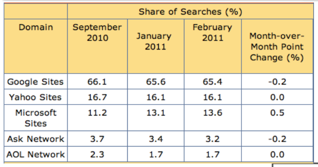
July 2006



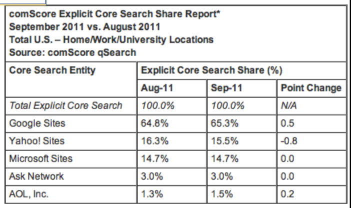
	Share of Searches (%)			
Domain	September 2010	January 2011	February 2011	Month-over-Month Point Change (%)
Google Sites	66.1	65.6	65.4	-0.2
Yahoo Sites	16.7	16.1	16.1	0.0
Microsoft Sites	11.2	13.1	13.6	0.5
Ask Network	3.7	3.4	3.2	-0.2
AOL Network	2.3	1.7	1.7	0.0

Feb 2011

Web search



	Share of Searches (%)			
Domain	September 2010	January 2011	February 2011	Month-over-Month Point Change (%)
Google Sites	66.1	65.6	65.4	-0.2
Yahoo Sites	16.7	16.1	16.1	0.0
Microsoft Sites	11.2	13.1	13.6	0.5
Ask Network	3.7	3.4	3.2	-0.2
AOL Network	2.3	1.7	1.7	0.0



comScore Explicit Core Search Share Report*
September 2011 vs. August 2011
Total U.S. – Home/Work/University Locations
Source: comScore qSearch

Core Search Entity	Explicit Core Search Share (%)		
	Aug-11	Sep-11	Point Change
Total Explicit Core Search	*100.0%*	*100.0%*	*N/A*
Google Sites	64.8%	65.3%	0.5
Yahoo! Sites	16.3%	15.5%	-0.8
Microsoft Sites	14.7%	14.7%	0.0
Ask Network	3.0%	3.0%	0.0
AOL, Inc.	1.3%	1.5%	0.2

Challenges

- Why is information retrieval hard?
  - Lots and lots of data
    - efficiency
    - storage
    - discovery (web)
  - Data is unstructured
  - Querying/Understanding user intent
  - SPAM
  - Data quality

web search engine traffic

About 92,200,000 results (0.13 seconds)

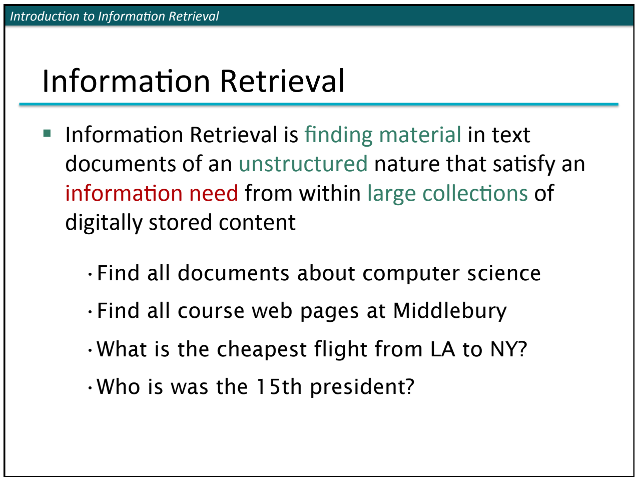
Information Retrieval

- Information Retrieval is finding material in documents of an unstructured nature that satisfy an information need from within large collections of digitally stored content

Information Retrieval

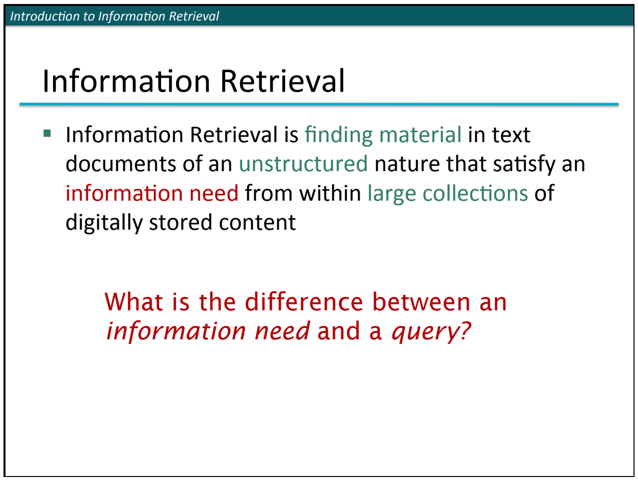
- Information Retrieval is finding material in documents of an unstructured nature that satisfy an information need from within large collections of digitally stored content

?



Information Retrieval

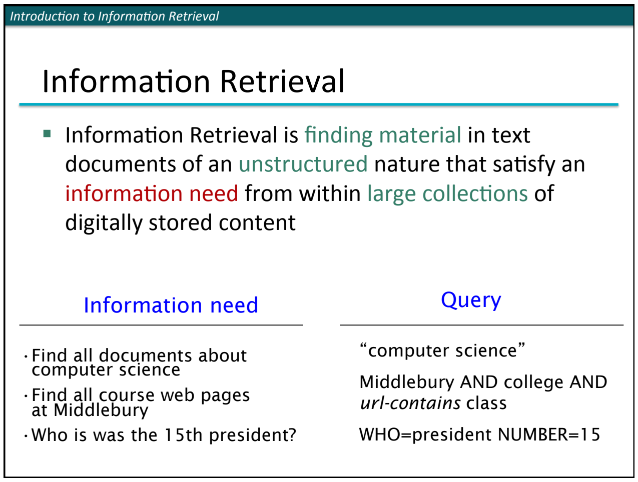
- Information Retrieval is finding material in text documents of an unstructured nature that satisfy an information need from within large collections of digitally stored content
  - Find all documents about computer science
  - Find all course web pages at Middlebury
  - What is the cheapest flight from LA to NY?
  - Who is was the 15th president?



Information Retrieval

- Information Retrieval is finding material in text documents of an unstructured nature that satisfy an information need from within large collections of digitally stored content

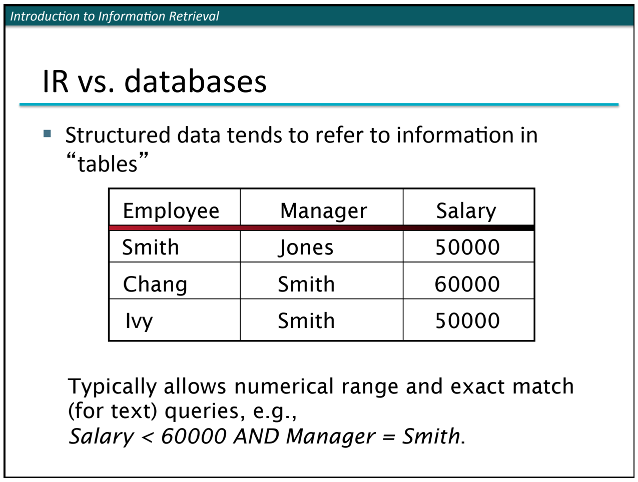
What is the difference between an *information need* and a *query?*



Information Retrieval

- Information Retrieval is finding material in text documents of an unstructured nature that satisfy an information need from within large collections of digitally stored content

Information need	Query
· Find all documents about computer science	“computer science”
· Find all course web pages at Middlebury	Middlebury AND college AND *url-contains* class
· Who is was the 15th president?	WHO=president NUMBER=15

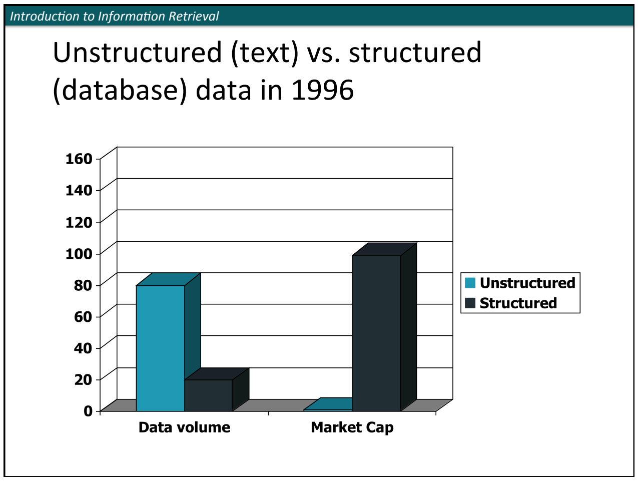


IR vs. databases

- Structured data tends to refer to information in “tables”

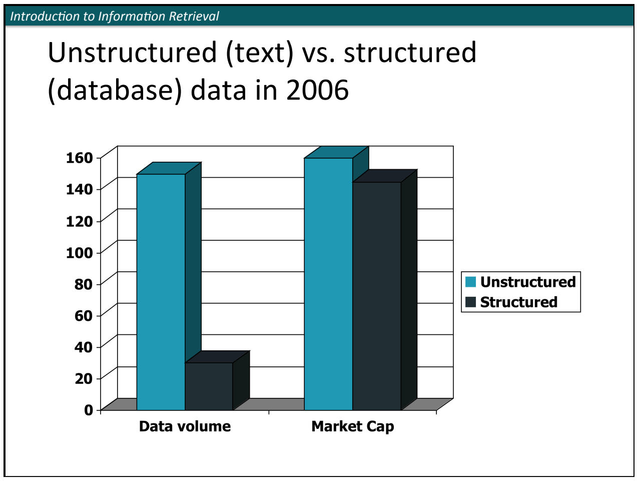
Employee	Manager	Salary
Smith	Jones	50000
Chang	Smith	60000
Ivy	Smith	50000

Typically allows numerical range and exact match (for text) queries, e.g.,
Salary < 60000 AND Manager = Smith.



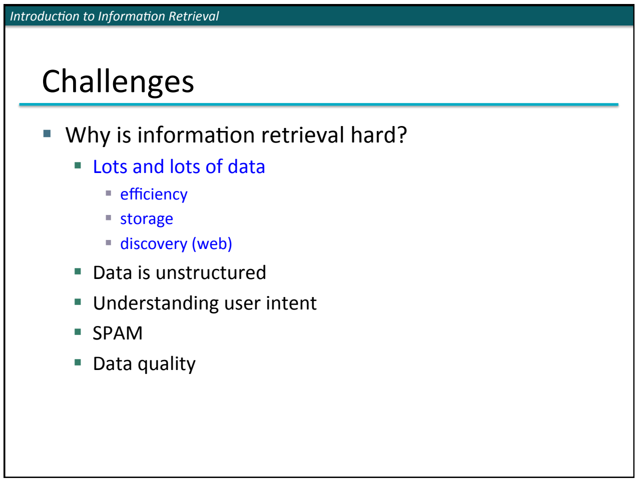
Introduction to Information Retrieval

Unstructured (text) vs. structured (database) data in 1996



Introduction to Information Retrieval

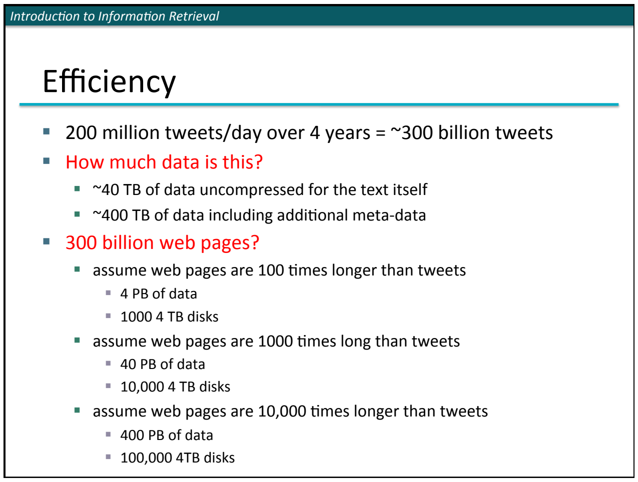
Unstructured (text) vs. structured (database) data in 2006



Introduction to Information Retrieval

Challenges

- Why is information retrieval hard?
  - Lots and lots of data
    - efficiency
    - storage
    - discovery (web)
  - Data is unstructured
  - Understanding user intent
  - SPAM
  - Data quality



Introduction to Information Retrieval

Efficiency

- 200 million tweets/day over 4 years = ~300 billion tweets
- How much data is this?
  - ~40 TB of data uncompressed for the text itself
  - ~400 TB of data including additional meta-data
- 300 billion web pages?
  - assume web pages are 100 times longer than tweets
    - 4 PB of data
    - 1000 4 TB disks
  - assume web pages are 1000 times long than tweets
    - 40 PB of data
    - 10,000 4 TB disks
  - assume web pages are 10,000 times longer than tweets
    - 400 PB of data
    - 100,000 4TB disks

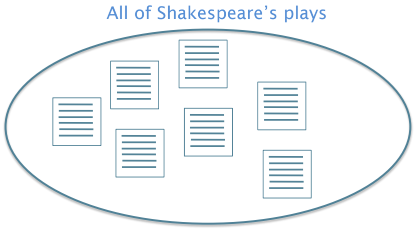
Efficiency

- Can we store all of the documents in memory?
- How long will it take to do a naïve search of the data?

- To search over a small data collection, almost any approach will work (e.g. grep)
- At web scale, there are many challenges:
  - queries need to be really fast!
  - massive parallelization
  - redundancy (hard-drives fail, networks fail, …)

Unstructured data in 1680

- Which plays of Shakespeare contain the words ***Brutus*** *AND* ***Caesar*** but *NOT* ***Calpurnia***?



How can we answer this query quickly?

Unstructured data in 1680

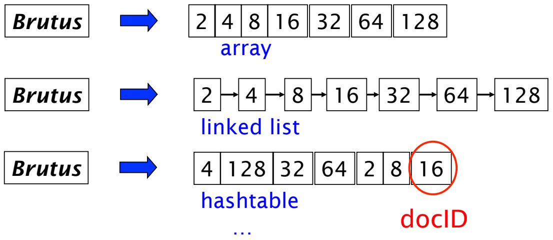
- Which plays of Shakespeare contain the words ***Brutus*** *AND* ***Caesar*** but *NOT* ***Calpurnia***?

- **Key idea:** we can pre-compute some information about the plays/documents that will make queries much faster
- What information do we need?

- Indexing: for each word, keep track of which documents it occurs in

Inverted index

- For each term/word, store a list of all documents that contain it
- What data structures might we use for this?

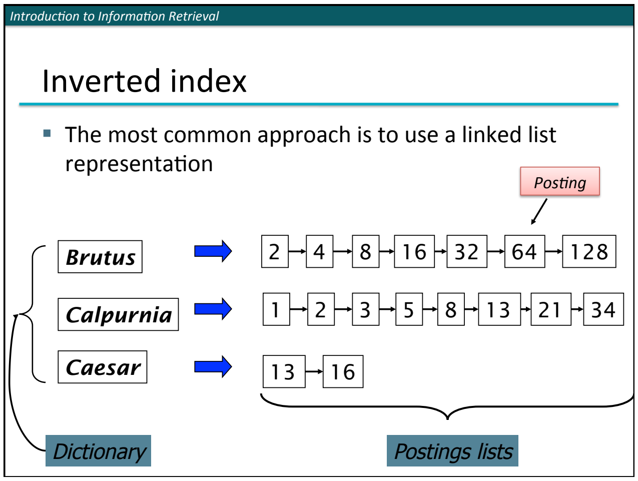

Brutus → 2 4 8 16 32 64 128
array

Brutus → 2 → 4 → 8 → 16 → 32 → 64 → 128
linked list

Brutus → 4 128 32 64 2 8 16
hashtable

…

docID



Inverted index

- The most common approach is to use a linked list representation

Posting

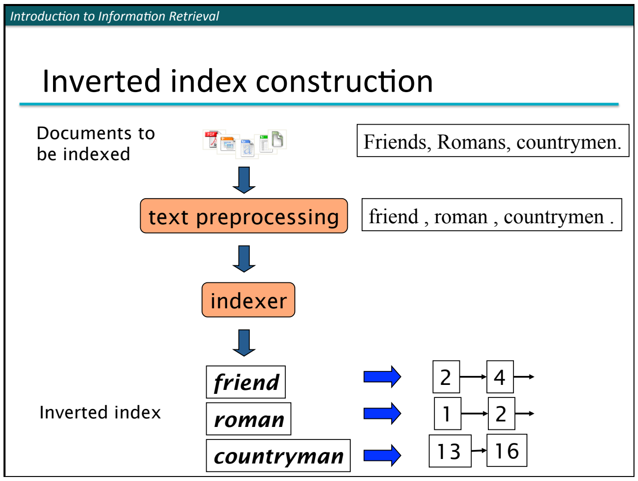
Brutus → 2 → 4 → 8 → 16 → 32 → 64 → 128

Calpurnia → 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

Caesar → 13 → 16

Dictionary

Postings lists



Inverted index construction

Documents to be indexed

Friends, Romans, countrymen.

text preprocessing

friend , roman , countrymen .

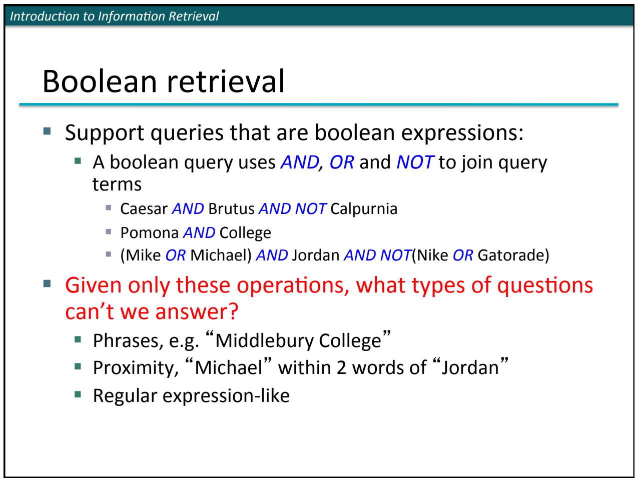
indexer

Inverted index

friend → 2 → 4 →

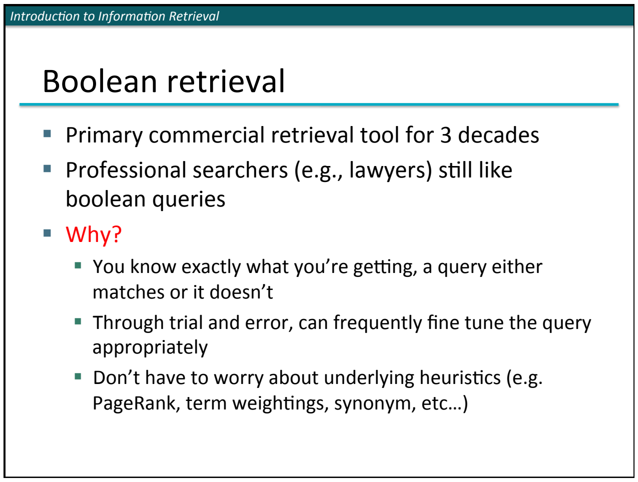
roman → 1 → 2 →

countryman → 13 → 16



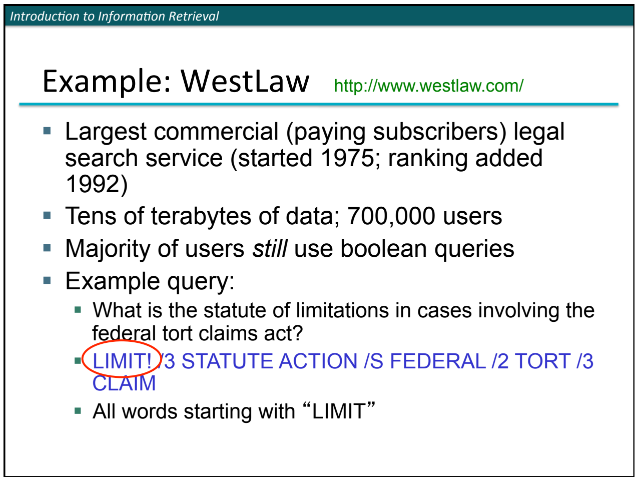
Boolean retrieval

- Support queries that are boolean expressions:
  - A boolean query uses *AND, OR* and *NOT* to join query terms
    - Caesar *AND* Brutus *AND NOT* Calpurnia
    - Pomona *AND* College
    - (Mike *OR* Michael) *AND* Jordan *AND NOT*(Nike *OR* Gatorade)
- Given only these operations, what types of questions can't we answer?
  - Phrases, e.g. "Middlebury College"
  - Proximity, "Michael" within 2 words of "Jordan"
  - Regular expression-like



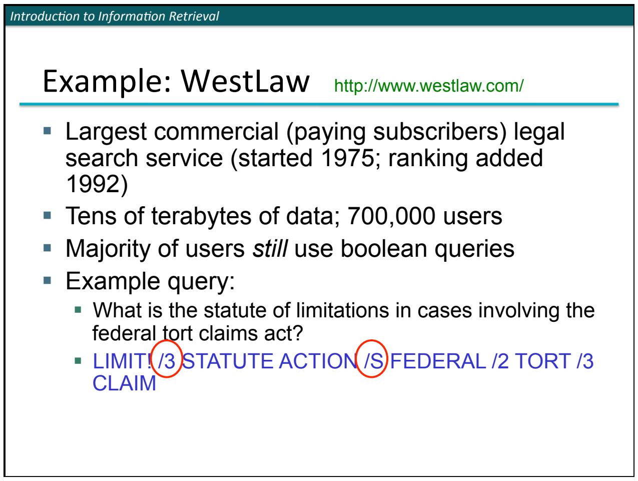
Boolean retrieval

- Primary commercial retrieval tool for 3 decades
- Professional searchers (e.g., lawyers) still like boolean queries
- Why?
  - You know exactly what you're getting, a query either matches or it doesn't
  - Through trial and error, can frequently fine tune the query appropriately
  - Don't have to worry about underlying heuristics (e.g. PageRank, term weightings, synonym, etc…)



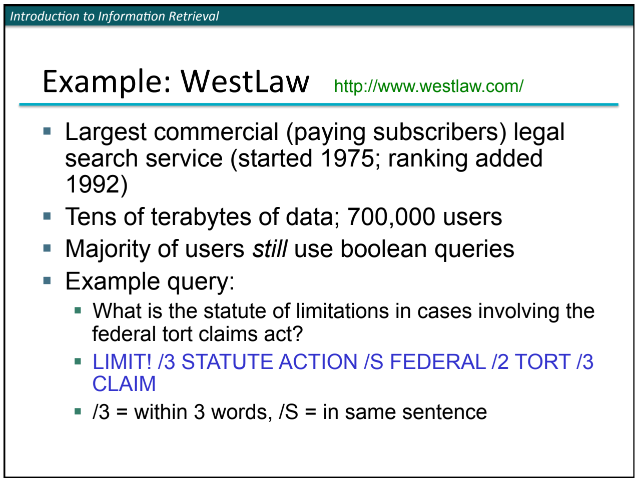
Example: WestLaw http://www.westlaw.com/

- Largest commercial (paying subscribers) legal search service (started 1975; ranking added 1992)
- Tens of terabytes of data; 700,000 users
- Majority of users *still* use boolean queries
- Example query:
  - What is the statute of limitations in cases involving the federal tort claims act?
  - LIMIT! /3 STATUTE ACTION /S FEDERAL /2 TORT /3 CLAIM
  - All words starting with “LIMIT”



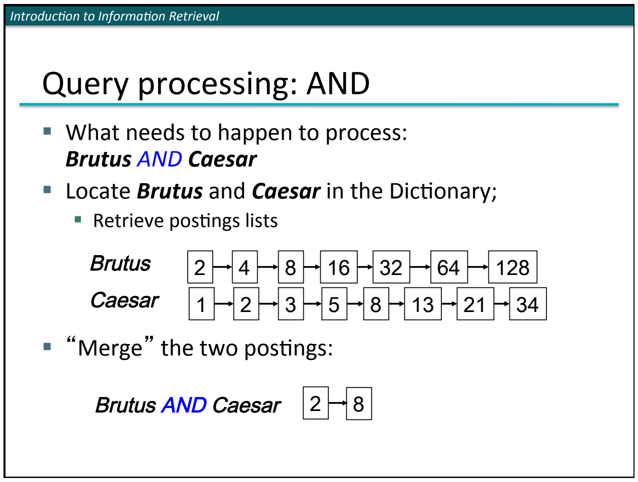
Example: WestLaw http://www.westlaw.com/

- Largest commercial (paying subscribers) legal search service (started 1975; ranking added 1992)
- Tens of terabytes of data; 700,000 users
- Majority of users *still* use boolean queries
- Example query:
  - What is the statute of limitations in cases involving the federal tort claims act?
  - LIMIT! /3 STATUTE ACTION /S FEDERAL /2 TORT /3 CLAIM



Example: WestLaw http://www.westlaw.com/

- Largest commercial (paying subscribers) legal search service (started 1975; ranking added 1992)
- Tens of terabytes of data; 700,000 users
- Majority of users *still* use boolean queries
- Example query:
  - What is the statute of limitations in cases involving the federal tort claims act?
  - LIMIT! /3 STATUTE ACTION /S FEDERAL /2 TORT /3 CLAIM
  - /3 = within 3 words, /S = in same sentence



Query processing: AND

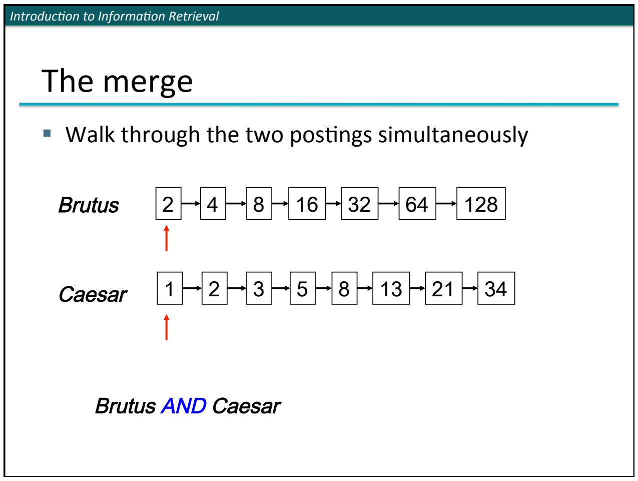
- What needs to happen to process: ***Brutus AND Caesar***
- Locate ***Brutus*** and ***Caesar*** in the Dictionary;
  - Retrieve postings lists

Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

- “Merge” the two postings:

Brutus AND Caesar 2 → 8



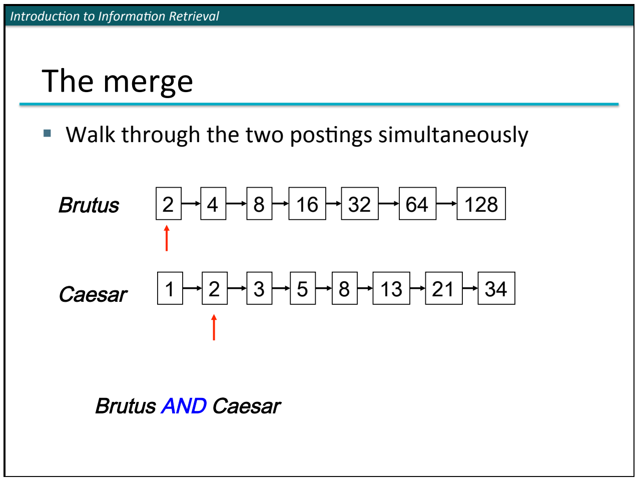
The merge

- Walk through the two postings simultaneously

Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

Brutus AND Caesar



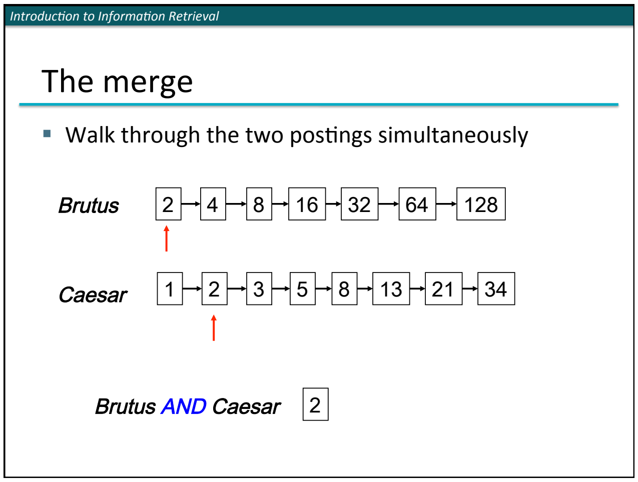
The merge

- Walk through the two postings simultaneously

Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

Brutus AND Caesar



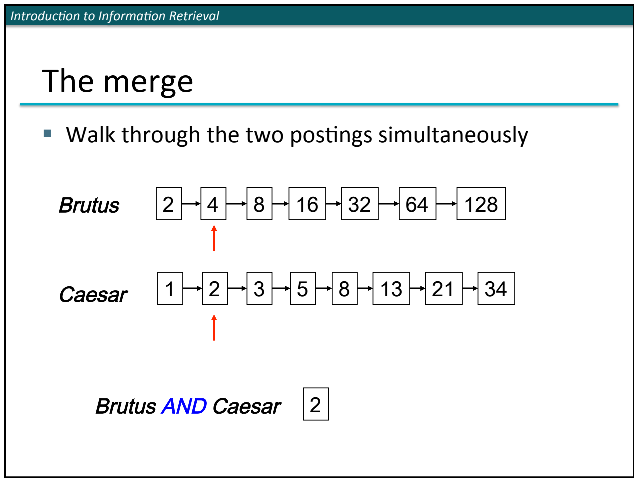
The merge

- Walk through the two postings simultaneously

Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

Brutus AND Caesar 2



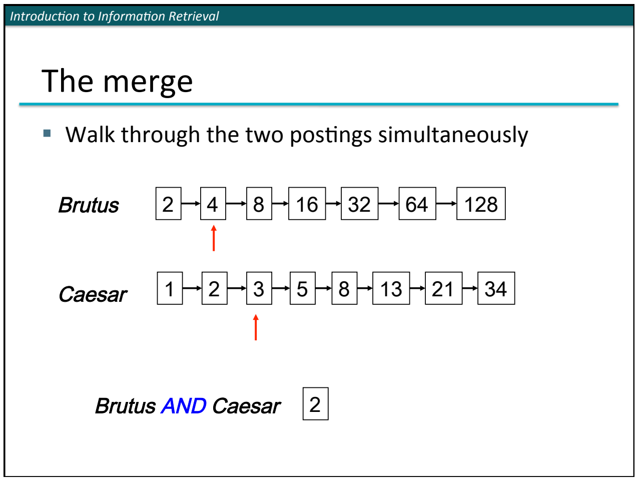
The merge

- Walk through the two postings simultaneously

Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

Brutus AND Caesar 2



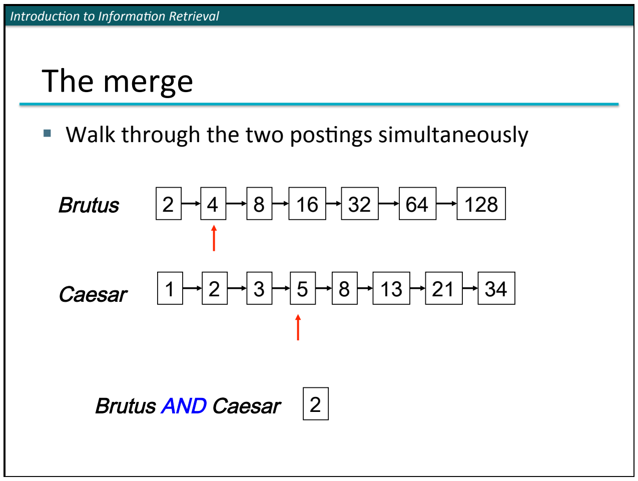
The merge

- Walk through the two postings simultaneously

Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

Brutus AND Caesar 2



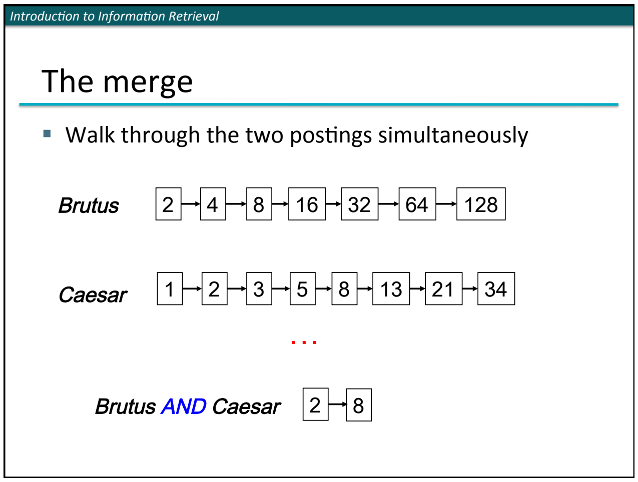
The merge

- Walk through the two postings simultaneously

Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

Brutus AND Caesar 2



The merge

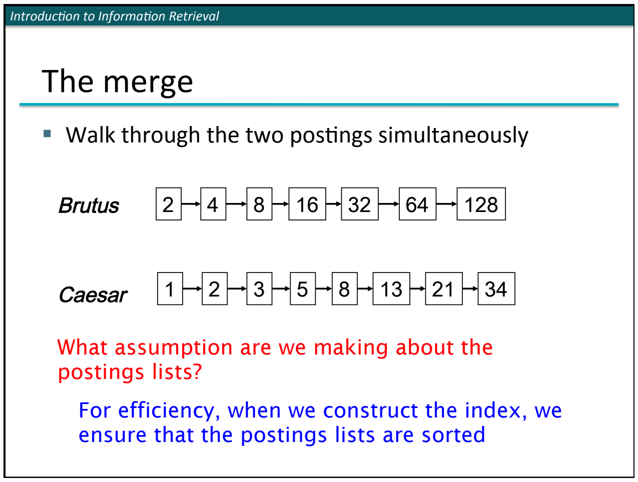
- Walk through the two postings simultaneously

Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

. . .

Brutus AND Caesar 2 → 8



The merge

- Walk through the two postings simultaneously

Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

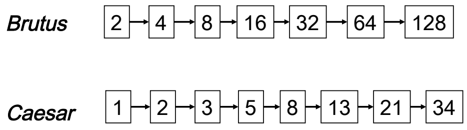
Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

What assumption are we making about the postings lists?

For efficiency, when we construct the index, we ensure that the postings lists are sorted

The merge

- Walk through the two postings simultaneously



Brutus 2 → 4 → 8 → 16 → 32 → 64 → 128

Caesar 1 → 2 → 3 → 5 → 8 → 13 → 21 → 34

What is the running time?

O(length1 + length2)

Boolean queries:
More general merges

- Which of the following queries can we still do in time O(length1+length2)?

Brutus *AND NOT* ***Caesar***

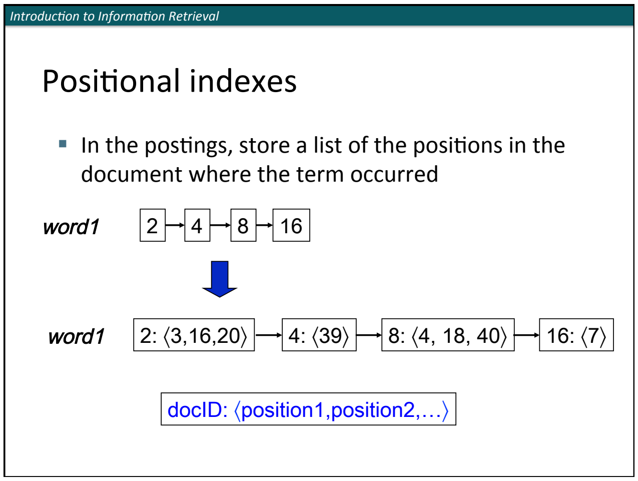
Brutus *OR NOT* ***Caesar***

From boolean to Google…

- What are we missing?
  - Phrases
    - ***Middlebury College***
  - Proximity: Find ***Gates*** *NEAR* ***Microsoft***.
  - Ranking search results
  - Incorporate link structure
  - document importance

From boolean to Google…

- Phrases
  - ***Middlebury College***
- Proximity: Find ***Gates*** *NEAR* ***Microsoft***
- Ranking search results
- Incorporate link structure
- document importance



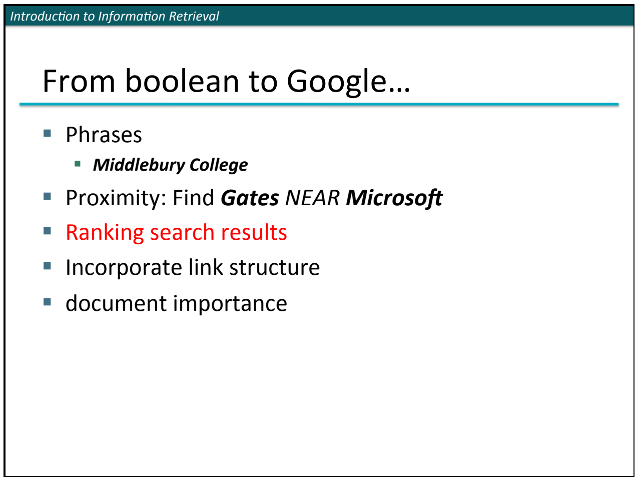
Positional indexes

- In the postings, store a list of the positions in the document where the term occurred

word1 2 → 4 → 8 → 16

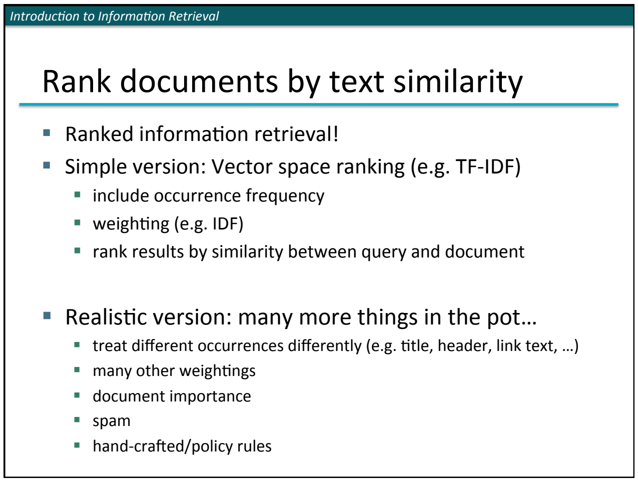
word1 2: ⟨3,16,20⟩ → 4: ⟨39⟩ → 8: ⟨4, 18, 40⟩ → 16: ⟨7⟩

docID: ⟨position1,position2,…⟩



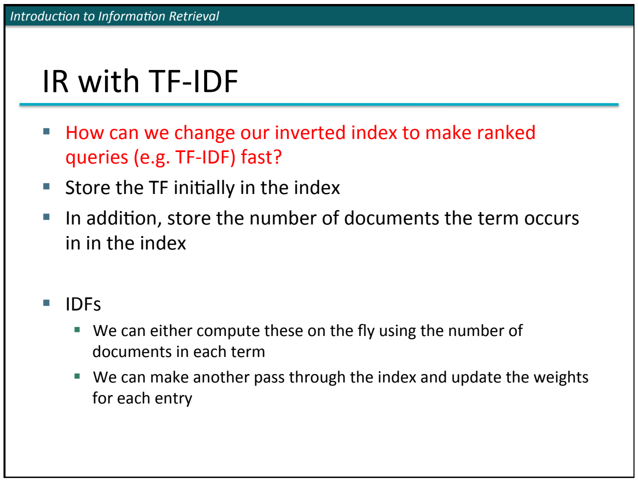
From boolean to Google…

- Phrases
  - ***Middlebury College***
- Proximity: Find ***Gates*** *NEAR* ***Microsoft***
- Ranking search results
- Incorporate link structure
- document importance



Rank documents by text similarity

- Ranked information retrieval!
- Simple version: Vector space ranking (e.g. TF-IDF)
  - include occurrence frequency
  - weighting (e.g. IDF)
  - rank results by similarity between query and document
- Realistic version: many more things in the pot…
  - treat different occurrences differently (e.g. title, header, link text, …)
  - many other weightings
  - document importance
  - spam
  - hand-crafted/policy rules



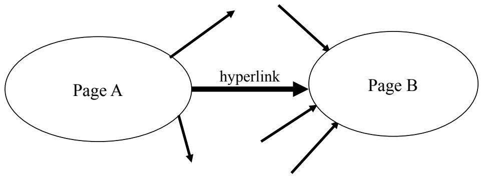
IR with TF-IDF

- How can we change our inverted index to make ranked queries (e.g. TF-IDF) fast?
- Store the TF initially in the index
- In addition, store the number of documents the term occurs in in the index
- IDFs
  - We can either compute these on the fly using the number of documents in each term
  - We can make another pass through the index and update the weights for each entry

From boolean to Google…

- Phrases
  - ***Middlebury College***
- Proximity: Find ***Gates*** *NEAR* ***Microsoft***
- Ranking search results
  - include occurrence frequency
  - weighting
  - treat different occurrences differently (e.g. title, header, link text, …)
- Incorporate link structure
- document importance

The Web as a Directed Graph

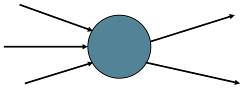


A hyperlink between pages denotes author perceived relevance AND importance

How can we use this information?

Query-independent ordering

- First generation: using link counts as simple measures of popularity
- Two basic suggestions:
  - Undirected popularity:
    - Each page gets a score = the number of in-links plus the number of out-links (3+2=5)
  - Directed popularity:
    - Score of a page = number of its in-links (3)

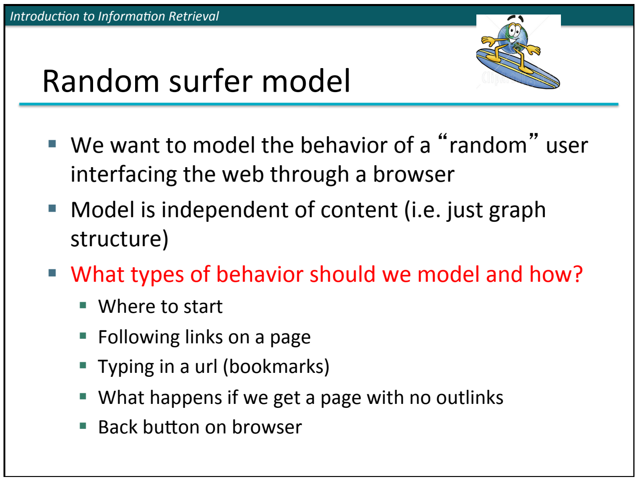


problems?

What is pagerank?

- The random surfer model
- Imagine a user surfing the web randomly using a web browser
- The pagerank score of a page is the probability that that user will visit a given page

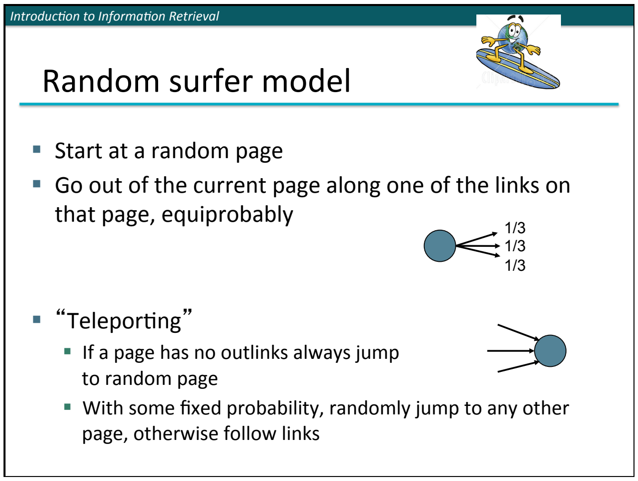
http://images.clipartof.com/small/7872-Clipart-Picture-Of-A-World-Earth-Globe-Mascot-Cartoon-Character-Surfing-On-A-Blue-And-Yellow-Surfboard.jpg



Introduction to Information Retrieval

Random surfer model

- We want to model the behavior of a "random" user interfacing the web through a browser
- Model is independent of content (i.e. just graph structure)
- What types of behavior should we model and how?
  - Where to start
  - Following links on a page
  - Typing in a url (bookmarks)
  - What happens if we get a page with no outlinks
  - Back button on browser

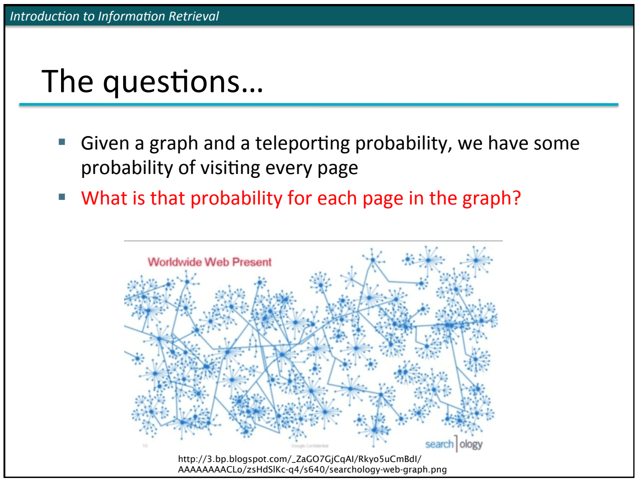


Introduction to Information Retrieval

Random surfer model

- Start at a random page
- Go out of the current page along one of the links on that page, equiprobably

- "Teleporting"
  - If a page has no outlinks always jump to random page
  - With some fixed probability, randomly jump to any other page, otherwise follow links

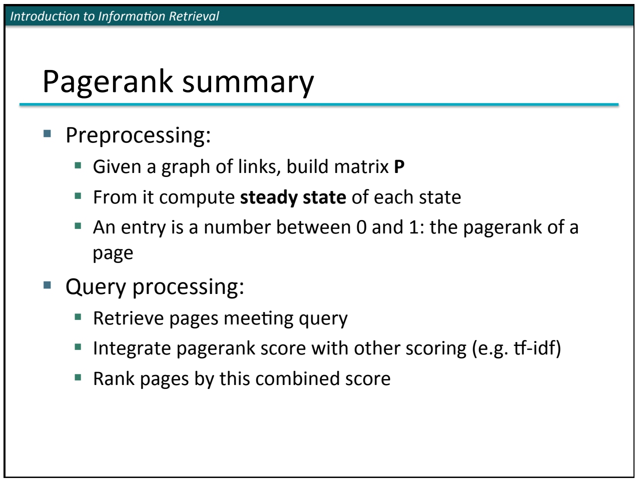


Introduction to Information Retrieval

The questions…

- Given a graph and a teleporting probability, we have some probability of visiting every page
- What is that probability for each page in the graph?

http://3.bp.blogspot.com/_ZaGO7GjCqAI/Rkyo5uCmBdI/AAAAAAAACLo/zsHdSlKc-q4/s640/searchology-web-graph.png



Introduction to Information Retrieval

Pagerank summary

- Preprocessing:
  - Given a graph of links, build matrix **P**
  - From it compute **steady state** of each state
  - An entry is a number between 0 and 1: the pagerank of a page
- Query processing:
  - Retrieve pages meeting query
  - Integrate pagerank score with other scoring (e.g. tf-idf)
  - Rank pages by this combined score

Pagerank problems?

- Can still fool pagerank
  - link farms
    - Create a bunch of pages that are tightly linked and on topic, then link a few pages to off-topic pages
  - link exchanges
    - I'll pay you to link to me
    - I'll link to you if you'll link to me
  - buy old URLs
  - post on blogs, etc. with URLs
  - Create crappy content (but still may seem relevant)

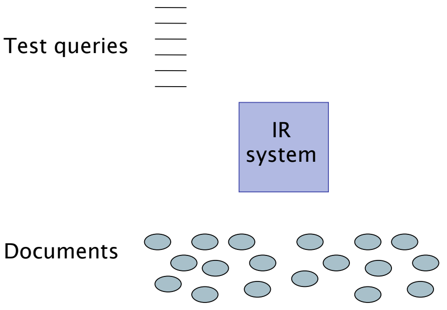
IR Evaluation

- Like any research area, an important component is how to evaluate a system
- What are important features for an IR system?
- How might we automatically evaluate the performance of a system? Compare two systems?
- What data might be useful?

Measures for a search engine

- How fast does it index (how frequently can we update the index)
- How fast does it search
- How big is the index
- Expressiveness of query language
- UI
- Is it free?
- Quality of the search results

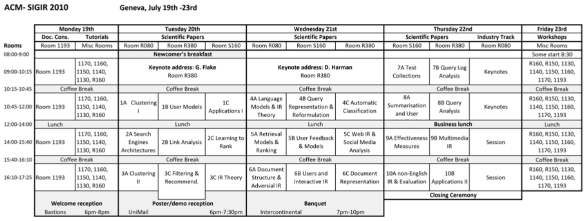
Data for evaluation



Many other evaluation measures…

- F1
- Precision at K
- 11-point average precision
- mean average precision (MAP) score
- normalized discounted cumulative gain (NDGC)
- …

IR Research



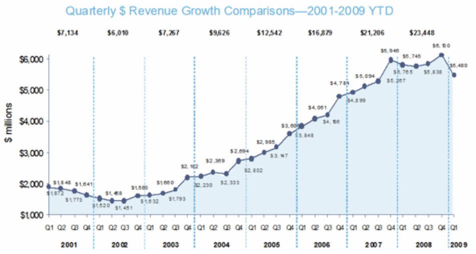
ACM- SIGIR 2010 **Geneva, July 19th -23rd**

	Monday 19th		**Tuesday 20th**			**Wednesday 21st**			**Thursday 22nd**			**Friday 23rd**
	Doc. Cons.	**Tutorials**	**Scientific Papers**			**Scientific Papers**			**Scientific Papers**		**Industry Track**	**Workshops**
Rooms	Room 1193	Misc Rooms	Room R080	Room R380	Room S160	Room R080	Room S160	Room R380	Room S160	Room R380	Room R080	Misc Rooms
08:00-9:00			**Newcomer's breakfast**									Some start 8:30
09:00-10:15	Room 1193	1170, 1160, 1150, 1140, 1130, R160	**Keynote address: G. Flake** Room R380			**Keynote address: D. Harman** Room R380			7A Test Collections	7B Query Log Analysis	Keynotes	R160, R150, 1130, 1140, 1150, 1160, 1170, 1193
10:15-10:45	Coffee Break		Coffee Break			Coffee Break			Coffee Break			Coffee Break
10:45-12:00	Room 1193	1170, 1160, 1150, 1140, 1130, R160	1A Clustering I	1B User Models	1C Applications I	4A Language Models & IR Theory	4B Query Representation & Reformulation	4C Automatic Classification	8A Summarisation and User	8B Query Analysis	Keynotes	R160, R150, 1130, 1140, 1150, 1160, 1170, 1193
12:00-14:00	Lunch		Lunch			Lunch			**Business lunch**			Lunch
14:00-15:40	Room 1193	1170, 1160, 1150, 1140, 1130, R160	2A Search Engines Architectures	2B Link Analysis	2C Learning to Rank	5A Retrieval Models & Ranking	5B User Feedback & Models	5C Web IR & Social Media Analysis	9A Effectiveness Measures	9B Multimedia IR	Session	R160, R150, 1130, 1140, 1150, 1160, 1170, 1193
15:40-16:10	Coffee Break		Coffee Break			Coffee Break			Coffee Break			Coffee Break
16:10-17:25	Room 1193	1170, 1160, 1150, 1140, 1130, R160	3A Clustering II	3C Filtering & Recommend.	3C IR Theory	6A Document Structure & Adversial IR	6B Users and Interactive IR	6C Document Representation	10A non-English IR & Evaluation	10B Applications II	Session	R160, R150, 1130, 1140, 1150, 1160, 1170, 1193
	Welcome reception Bastions 6pm-8pm		**Poster/demo reception** UniMail 6pm-7:30pm			**Banquet** Intercontinental 7pm-10pm			**Closing Ceremony**			

$$$$

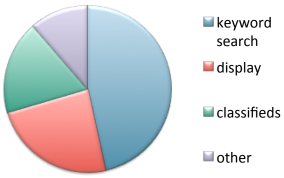
- How do search engines make money?
- How much money do they make?

Online advertising $



http://www.iab.net/about_the_iab/recent_press_releases/press_release_archive/press_release/pr-060509

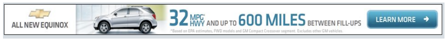
Where the $ comes from



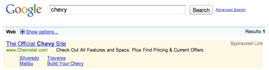
http://www.informationweek.com/news/internet/reporting/showArticle.jhtml?articleID=207800456

3 major types of online ads

- Banner ads

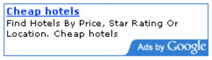

ALL NEW EQUINOX
32 MPG HWY AND UP TO 600 MILES BETWEEN FILL-UPS
LEARN MORE
*Based on EPA estimates, FWD models and GM Compact Crossover segment. Excludes other GM vehicles.

- Keyword linked ads



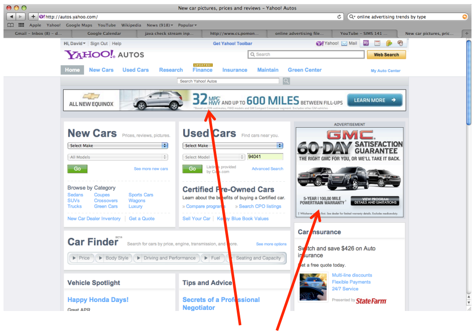
The Official Chevy Site
www.Chevrolet.com Check Out All Features and Specs. Plus Find Pricing & Current Offers.
Silverado Traverse
Malibu Build Your Chevy

- Context linked ads



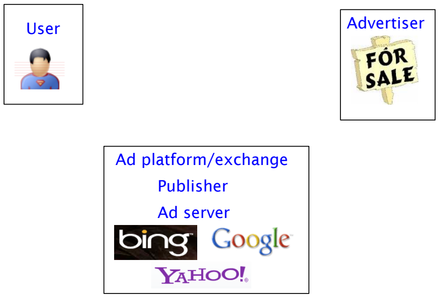
Cheap hotels
Find Hotels By Price, Star Rating Or Location. Cheap hotels

Banner ads



standardized set of sizes

Paid search components

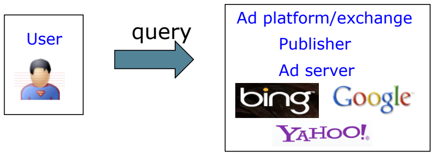


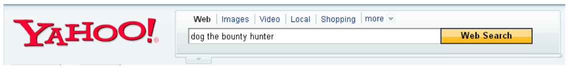
User

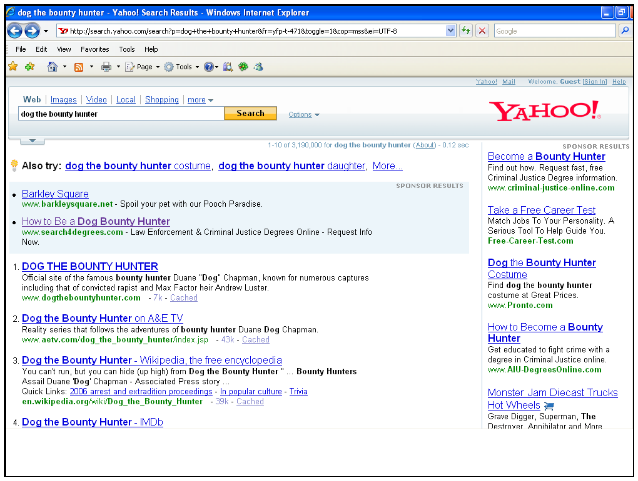
Advertiser

Ad platform/exchange
Publisher
Ad server

Paid search query






dog the bounty hunter - Yahoo! Search Results - Windows Internet Explorer
http://search.yahoo.com/search?p=dog+the+bounty+hunter&fr=yfp-t-471&toggle=1&cop=mss&ei=UTF-8
Google
File
Edit
View
Favorites
Tools
Help
Page
Tools
Yahoo!
Mail
Welcome, Guest [Sign In]
Help
Web
Images
Video
Local
Shopping
more
dog the bounty hunter
Search
Options
YAHOO!

1-10 of 3,190,000 for **dog the bounty hunter** (About) - 0.12 sec

Also try: **dog the bounty hunter** costume, **dog the bounty hunter** daughter, More...

SPONSOR RESULTS

- Barkley Square
  www.**barkleysquare.net** - Spoil your pet with our Pooch Paradise.
- How to Be a **Dog Bounty Hunter**
  www.**search4degrees.com** - Law Enforcement & Criminal Justice Degrees Online - Request Info Now.

1. **DOG THE BOUNTY HUNTER**
   Official site of the famous **bounty hunter** Duane "**Dog**" Chapman, known for numerous captures including that of convicted rapist and Max Factor heir Andrew Luster.
   www.**dogthebountyhunter.com** - 7k - Cached
2. **Dog the Bounty Hunter** on A&E TV
   Reality series that follows the adventures of **bounty hunter** Duane **Dog** Chapman.
   www.**aetv.com/dog_the_bounty_hunter**/index.jsp - 43k - Cached
3. **Dog the Bounty Hunter** - Wikipedia, the free encyclopedia
   You can't run, but you can hide (up high) from **Dog the Bounty Hunter** " ... **Bounty Hunters** Assail Duane '**Dog**' Chapman - Associated Press story ...
   Quick Links: 2006 arrest and extradition proceedings - In popular culture - Trivia
   en.wikipedia.org/wiki/**Dog_the_Bounty_Hunter** - 39k - Cached
4. **Dog the Bounty Hunter** - IMDb

SPONSOR RESULTS

Become a **Bounty Hunter**
Find out how. Request fast, free Criminal Justice Degree information.
www.**criminal-justice-online.com**

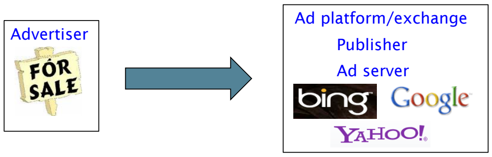
Take a Free Career Test
Match Jobs To Your Personality. A Serious Tool To Help Guide You.
Free-Career-Test.com

Dog the **Bounty Hunter** Costume
Find **dog** the **bounty hunter** costume at Great Prices.
www.**Pronto.com**

How to Become a **Bounty Hunter**
Get educated to fight crime with a degree in Criminal Justice online.
www.**AIU-DegreesOnline.com**

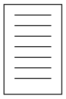
Monster Jam Diecast Trucks Hot Wheels
Grave Digger, Superman, **The** Destroyer, Annihilator and More

What is required of the advertiser?



Advertiser

set of keywords

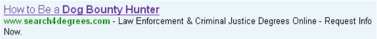


landing page

bids

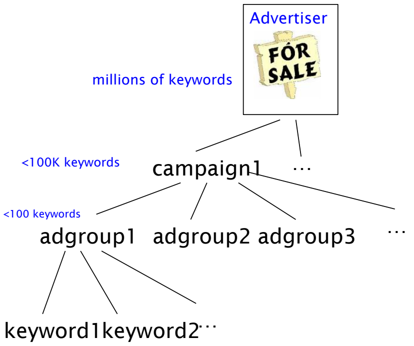
$

ad copy



How to Be a **Dog Bounty Hunter**
www.**search4degrees.com** - Law Enforcement & Criminal Justice Degrees Online - Request Info Now.

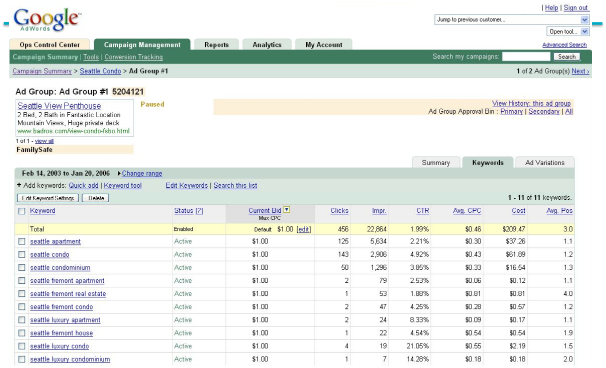
A bit more structure than this…



Adgroups

- Adgroups are the key structure
- Adcopy and landing pages are associated at the adcopy level
- Keywords should be tightly themed
  - promotes targeting
  - makes google, yahoo, etc. happy

Creating an AdWords Ad



Campaign Summary > Seattle Condo > Ad Group #1

1 of 2 Ad Group(s) Next ›

Ad Group: Ad Group #1 5204121

Seattle View Penthouse
2 Bed, 2 Bath in Fantastic Location
Mountain Views, Huge private deck
www.badros.com/view-condo-fsbo.html

Paused

View History: this ad group
Ad Group Approval Bin : Primary | Secondary | All

1 of 1 - view all

FamilySafe

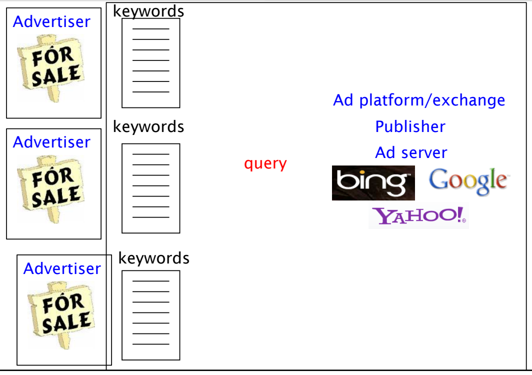
Feb 14, 2003 to Jan 20, 2006 ▸ Change range

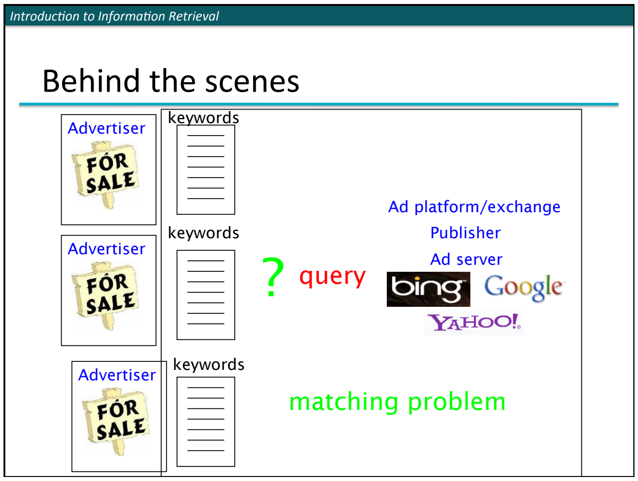
+ Add keywords: Quick add | Keyword tool   Edit Keywords | Search this list

1 - 11 of 11 keywords.

□	Keyword	Status [?]	Current Bid ▼ Max CPC	Clicks	Impr.	CTR	Avg. CPC	Cost	Avg. Pos
	Total	Enabled	Default $1.00 [edit]	456	22,864	1.99%	$0.46	$209.47	3.0
□	seattle apartment	Active	$1.00	125	5,634	2.21%	$0.30	$37.26	1.1
□	seattle condo	Active	$1.00	143	2,906	4.92%	$0.43	$61.89	1.2
□	seattle condominium	Active	$1.00	50	1,296	3.85%	$0.33	$16.54	1.3
□	seattle fremont apartment	Active	$1.00	2	79	2.53%	$0.06	$0.12	1.1
□	seattle fremont real estate	Active	$1.00	1	53	1.88%	$0.81	$0.81	4.0
□	seattle fremont condo	Active	$1.00	2	47	4.25%	$0.28	$0.57	1.2
□	seattle luxury apartment	Active	$1.00	2	24	8.33%	$0.09	$0.17	1.1
□	seattle fremont house	Active	$1.00	1	22	4.54%	$0.54	$0.54	1.9
□	seattle luxury condo	Active	$1.00	4	19	21.05%	$0.55	$2.19	1.5
□	seattle luxury condominium	Active	$1.00	1	7	14.28%	$0.18	$0.18	2.0

Behind the scenes





Behind the scenes

Advertiser

keywords

Advertiser

keywords

Advertiser

keywords

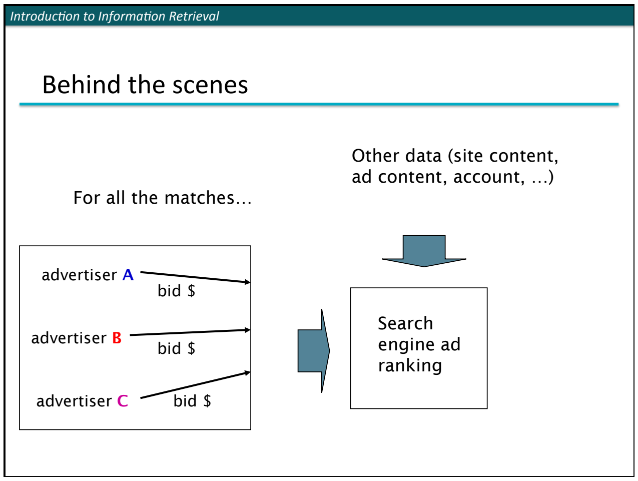
? query

Ad platform/exchange

Publisher

Ad server

matching problem



Behind the scenes

For all the matches…

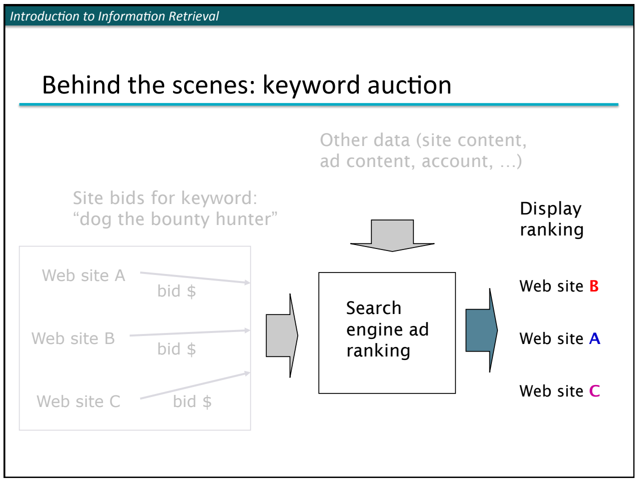
Other data (site content,
ad content, account, …)

advertiser A bid $

advertiser B bid $

advertiser C bid $

Search engine ad ranking



Behind the scenes: keyword auction

Site bids for keyword:
“dog the bounty hunter”

Other data (site content,
ad content, account, …)

Web site A bid $

Web site B bid $

Web site C bid $

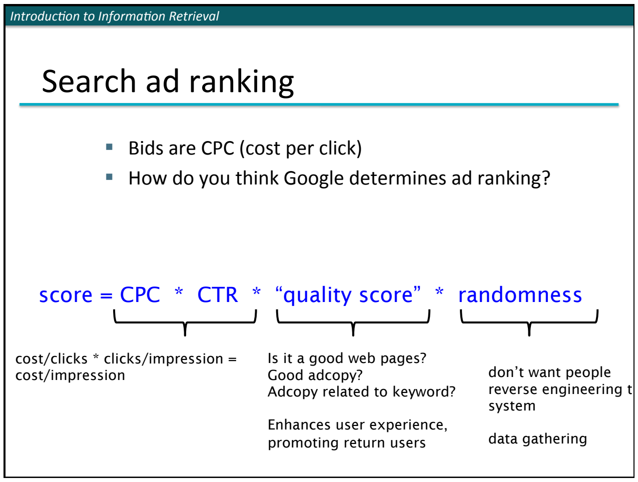
Search engine ad ranking

Display ranking

Web site B

Web site A

Web site C



Search ad ranking

- Bids are CPC (cost per click)
- How do you think Google determines ad ranking?

$$\text{score} = \underbrace{\text{CPC} * \text{CTR}}_{\substack{\text{cost/clicks * clicks/impression =}\\ \text{cost/impression}}} * \underbrace{\text{“quality score”}}_{\substack{\text{Is it a good web pages?}\\ \text{Good adcopy?}\\ \text{Adcopy related to keyword?}\\ \text{Enhances user experience,}\\ \text{promoting return users}}} * \underbrace{\text{randomness}}_{\substack{\text{don't want people}\\ \text{reverse engineering t}\\ \text{system}\\ \text{data gathering}}}$$