CS347

Lecture 12

May 21, 2001

Topics

- Web characterization
- Research Problems

The Web: A directed graph

- Nodes = static web pages (1+ billion)
- Edges = static hyperlinks (~10 billion)
- Web graph = Snapshot of web pages and hyperlinks
- Sparse graph: ~7 links/page on average
- Focus on graph structure, ignore content

Questions about the web graph

- How big is the graph? How many links on a page (outdegree)? How many links to a page (indegree)?
- Can one browse from any web page to any other? How many clicks?
- Can we pick a random page on the web?
  - Search engine measurement.

Questions about the web graph

- Can we exploit the structure of the web graph for searching and mining?
- What does the web graph reveal about social processes which result in its creation and dynamics?
- How different is browsing from a “random walk”?

Why?

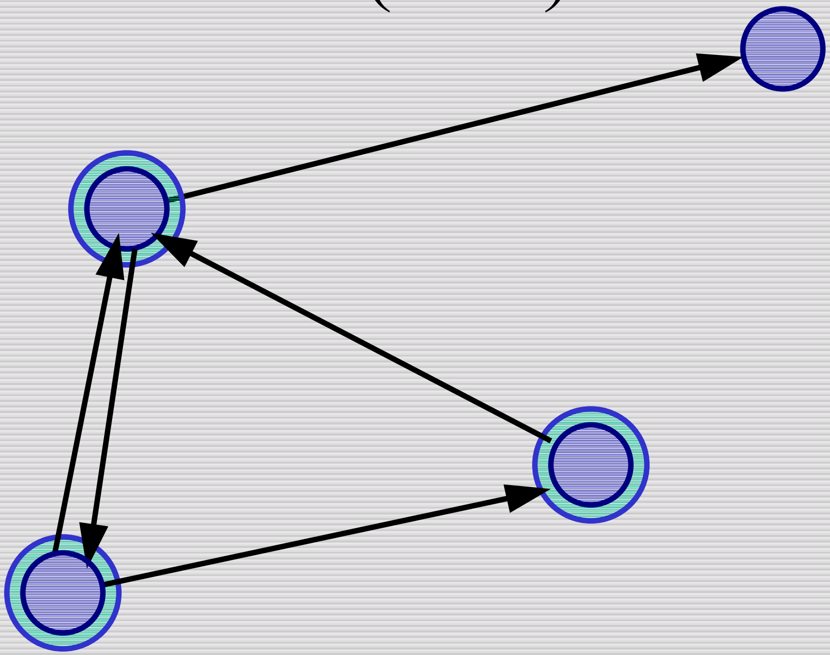
- Exploit structure for Web algorithms
  - Crawl strategies
  - Search
  - Mining communities
- Classification/organization
- Web anthropology
  - Prediction, discovery of structures
  - Sociological understanding

Web snapshots

- Altavista crawls (May 99/Oct 99/Feb 00)
- 220/317/500M pages
- 1.5/2.1B/5B hyperlinks
- Compaq CS2 connectivity server
  - back-link information
  - 10bytes/url, 3.4bytes/link, 0.15µs/access
  - given pages, return their in/out neighborhood

Algorithms

- Weakly connected components (WCC)
- Strongly connected components (SCC)
- Breadth-first search (BFS)
- Diameter



Challenges from scale

- Typical diameter algorithm:
  - number of steps ~ pages × links.
  - For 500 million pages, 5 billion links, even at a *very* optimistic 0.15μs/step, we need

    ~4 billion seconds.

  Hopeless.
- Will estimate diameter/distance metrics.

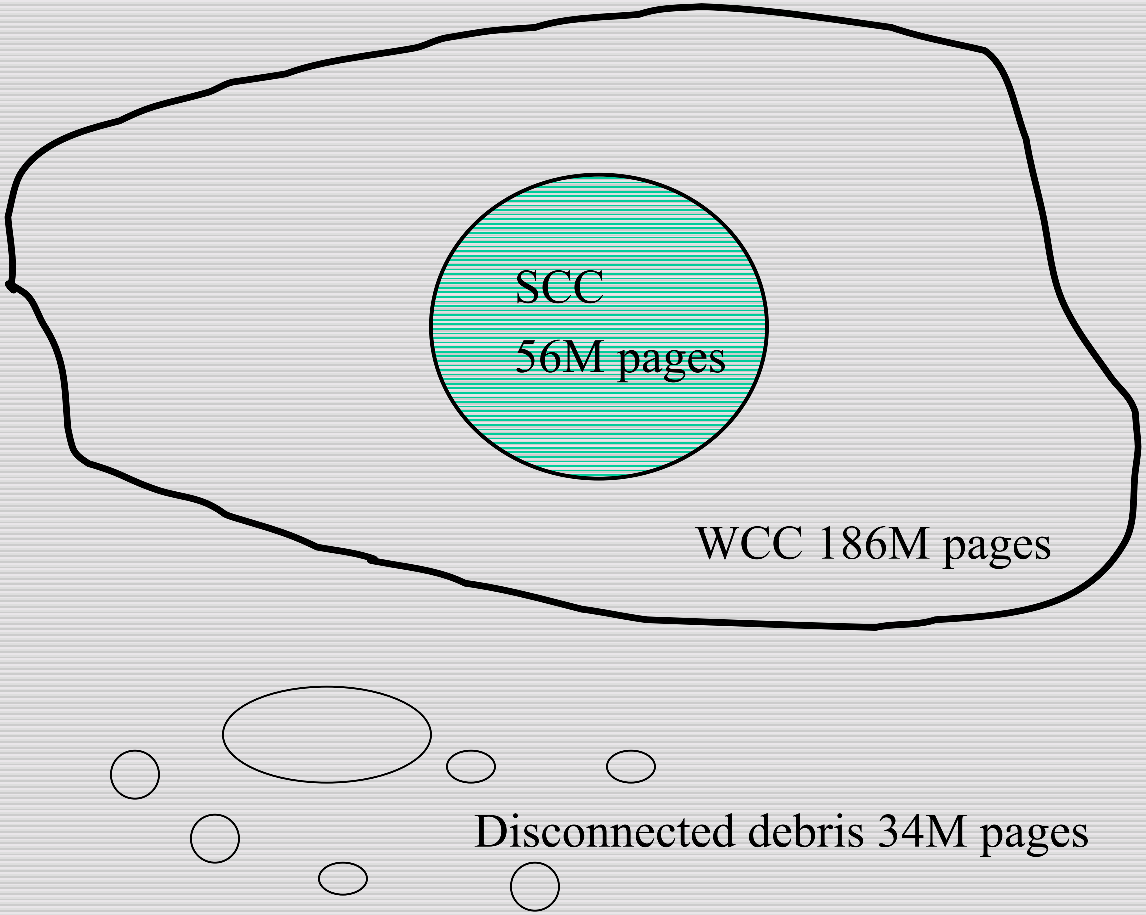
Scale

- On the other hand, can handle tasks linear in the links (5 billion) at a μs/step.
  - E.g., breadth-first search
- First eliminate duplicate pages/mirrors.
- Linear-time implementations for WCC and SCC.

May 1999 crawl

- 220 million pages after duplicate elimination.
- Giant WCC has ~186 million pages.
- Giant SCC has ~56 million pages.
  - Cannot browse your way from any page to any other
  - Next biggest SCC ~150K pages
- Fractions roughly the same in other crawls.

Tentative picture



SCC

56M pages

WCC 186M pages

Disconnected debris 34M pages

Breadth-first search (BFS)

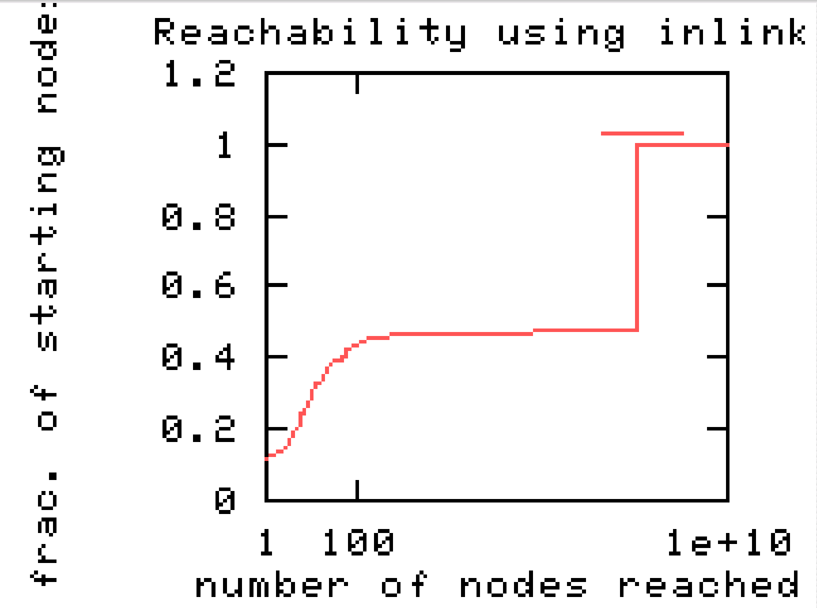
- Start at a page *p*
  - get its neighbors;
  - their neighbors, etc.
- Get profile of the number of pages reached by crawling out of *p*, as a function of distance *d*
- Can do this following links forwards as well as backwards

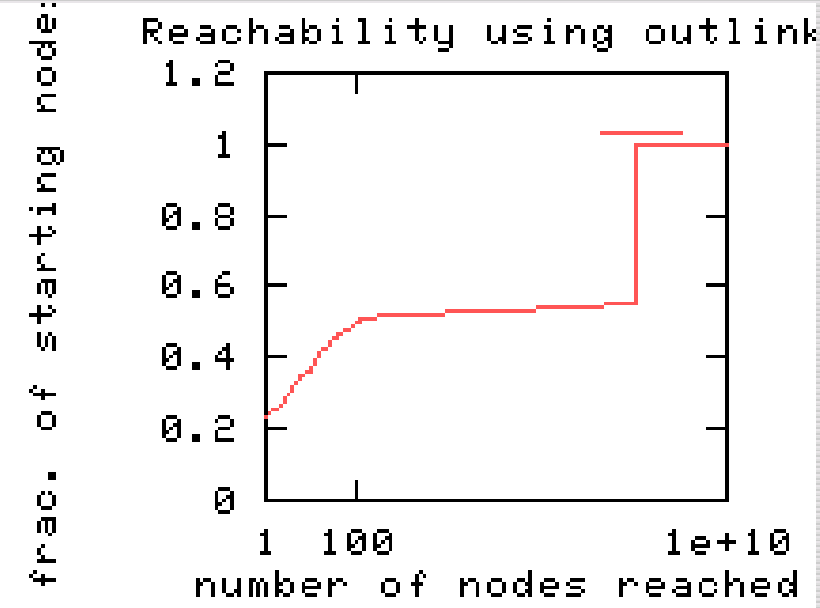
BFS experiment

- Start at 1000+ random pages
- For each start page, build BFS (reachability vs. distance) profiles going forwards, and backwards

Reachability

How many pages are reachable from a random page?





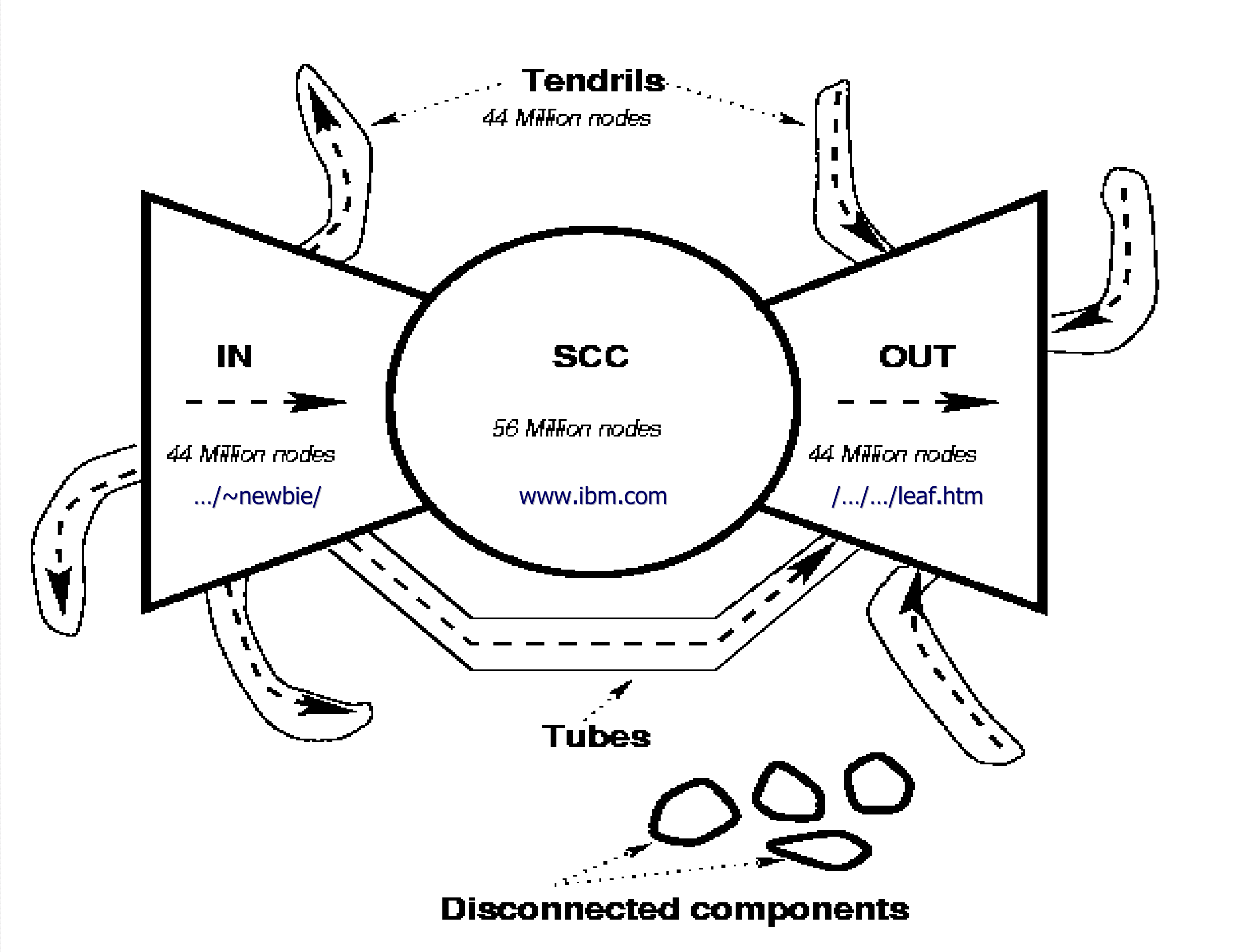
Net of BFS experiments

- BFS out of a page
  - – either dies quickly (~100 pages reached)
  - – “explodes” and reaches ~100 million pages
    - somewhat over 50% of starting pages
  - – SCC pages ~25% of total, reach >56M pages
- Qualitatively the same following in- or out-links

Interpreting BFS expts

- Need another 100-56 = 44M pages reachable from SCC
  - gives us 100M pages reachable from SCC
- Likewise, need another ~44M pages reachable from SCC going backwards
- These together don't account for all 186M pages in giant WCC.

Web anatomy


Tendrils
44 Million nodes
IN
44 Million nodes
.../~newbie/
SCC
56 Million nodes
www.ibm.com
OUT
44 Million nodes
/.../.../leaf.htm
Tubes
Disconnected components

Distance measurements

- For random pages *p1*,*p2*:

  Pr[*p1* reachable from *p2*] ~ 1/4
- Maximum directed distance between 2 SCC nodes: >28
- Maximum directed distance between 2 nodes, given there is a path: > 900
- Average directed distance between 2 SCC nodes: ~16
- Average undirected distance: ~7

Exercise

- Given the BFS and component size measurements, how can we infer all of the above measurements?

Power laws on the Web

- Inverse polynomial distributions:

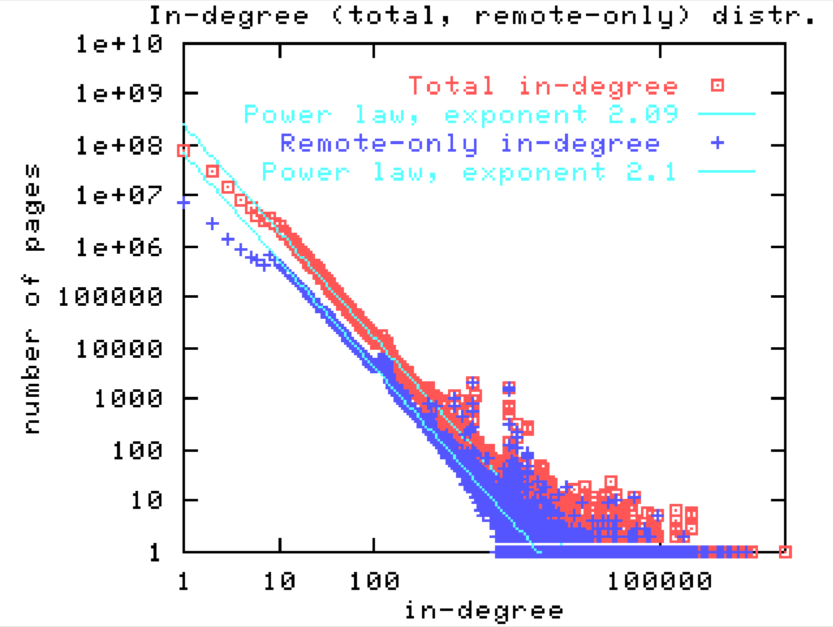
  $\Pr[k] \sim c/k^{\alpha}$ for a constant c.

  $\Leftrightarrow \log \Pr[k] \sim c - \alpha \log k$

- Thus plotting $\log \Pr[k]$ against $\log k$ should give a straight line (of negative slope).

In-degree distribution

Probability that a random page has *k* other pages pointing to it is $\sim k^{-2.1}$ (Power law)

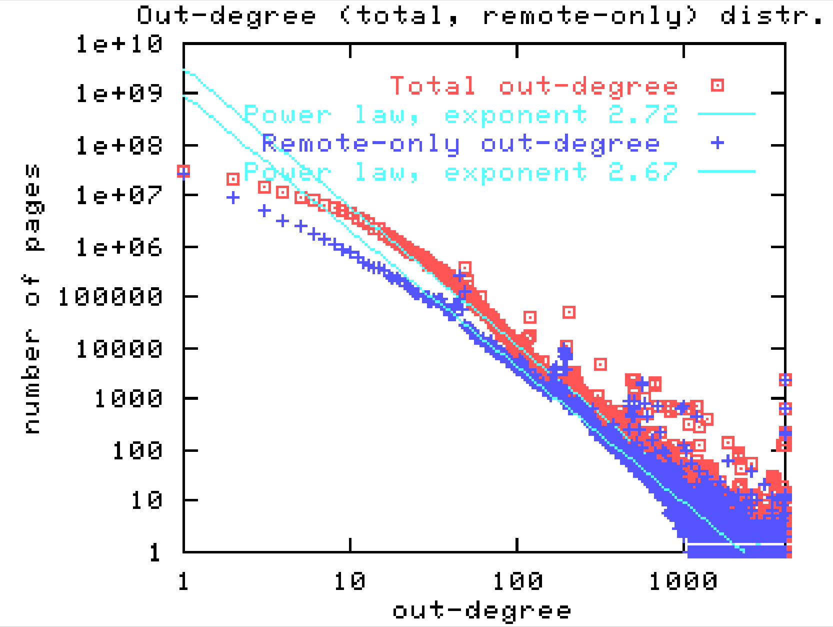


Slope = -2.1

Out-degree distribution

Probability that a random page points to *k* other pages is

$\sim k^{-2.7}$

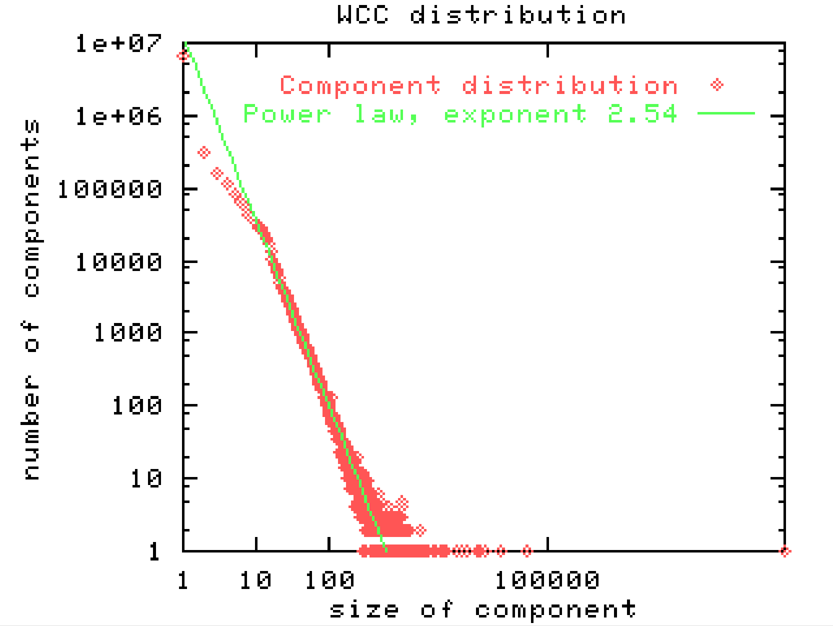


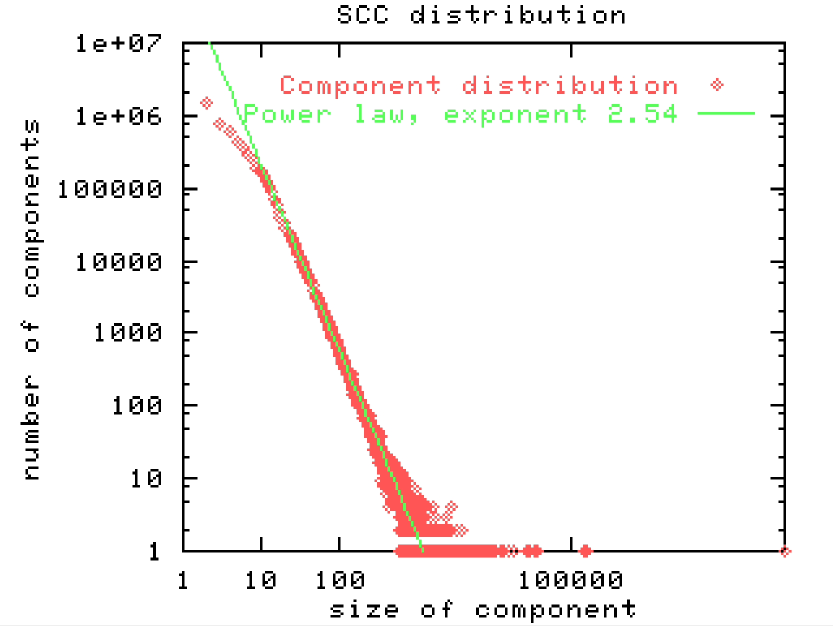
Slope = -2.7

Connected components

Largest WCC = 186M, SCC = 56M

Connected component sizes:





Other Web/internet power laws

- Rates of visits to sites
- Degrees of nodes in physical network

Resources

- Broder et al. Graph structure in the Web. WWW9, 2000. *www.almaden.ibm.com/cs/k53/www9.final/*
- Albert, R., Jeong, H., & Barabasi, A.L. (1999). Diameter of the world wide web, Nature, 401, 130-131. *http://citeseer.nj.nec.com/context/938378/0*
- M. Faloutsos, P. Faloutsos, and C. Faloutsos, On Power-Law Relationships of the Internet Topology. SIGCOMM '99, pp. 251-262, Aug. 1999. *http://citeseer.nj.nec.com/context/973789/208125*

Open Problems

P Papers/prizes

$ Money

? Difficulty

Computational bottlenecks

- If computation were not a limit, could we get better ranking in search results?
- Better classification?
- Better clustering?
- What does "better" mean?

P P P P $? ? ? ? ?

Set intersection in search

- For query w/AND of two terms, we retrieve and intersect their postings' sets
  - Can do work disproportionately large compared to the size of the output.
- Is there a data structure that does better than this - without keeping a postings entry for each pair of terms?

P $ $? ?

Text query optimization

- Recommended query processing order in early lectures - simple heuristics
  - infamous true/false question from midterm
- What can we do that’s more sophisticated but still fast in practice?

P $ $? ?

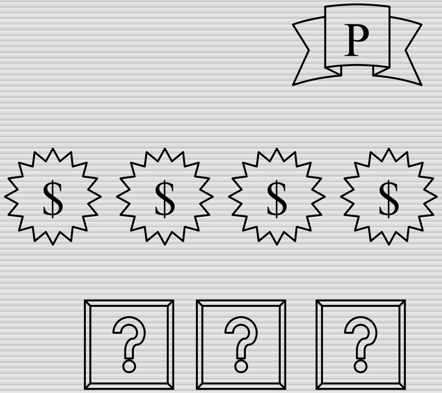
Practical nearest-neighbor search

- In high-dimensional vector spaces
  - moderate preprocessing
  - fast query processing
  - nearly accurate nearest neighbors

P P P $ $ $ $? ? ?

Classification

- Saw several schemes (Bayes, SVM) for classifying based on exemplary docs.
- Can also automatically classify based on persistent queries.
- How can we combine the two?
- Issues:
  - Combined representation of topic.
  - UI design vs.representation.



Benchmarks

- Web IR - search/classification benchmarks.
- Benchmarks for measuring recommendation systems.

P P P P $? ? ?

Taxonomy construction

- Metrics of human effort
  - how much human effort vs. accuracy
    - training by exemplary docs vs. persistent queries
- UI effects
  - what is the ideal user environment for building taxonomies
- What does it take to get to 98+ % accuracy?
  - Combination of UI, algorithms, best practices

P P P $ $?

Summarization

- How do you summarize a set of docs?
  - Results of clustering/trawling/ …
  - Visual vs. textual vs. combinations
- Measuring quality of summarization.

P P P P $ $? ? ?

Corpus analysis

- Given a corpus, extract its significant themes
  - organize into a navigation structure
- Visualization of themes in corpus
- Power set: all subsets of docs in a corpus
  - some subsets are interesting - which ones?
  - how do you organize them for human consumption?

P P P P P $ $ $? ? ? ?

Intranets vs. internet

- How do intranet structures differ from internet structures?
  - Influence of policy on content creators.

P P P $?

Recurring themes

- Not an exact science
- Focus on end-user
  - who? why? how?
- Bend rules for progress
  - ignore performance to start with
  - think huge - power sets