An Introduction to Web Mining

Ricardo Baeza-Yates, Aristides Gionis
Yahoo! Labs
Barcelona, Spain

RuSSIR
Russian Summer School in Information Retrieval

Contents of the tutorial

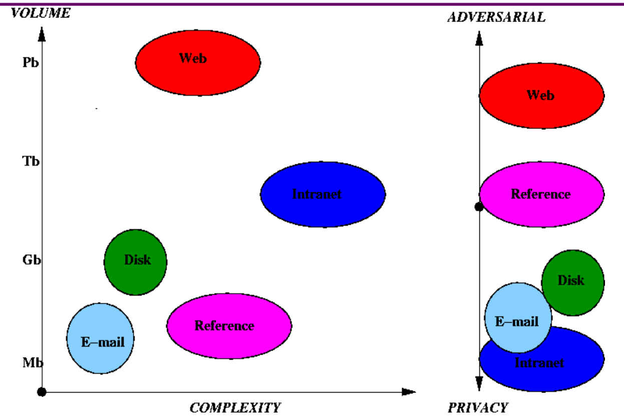
1. Motivations for Web mining
   - The Web, definitions, wisdom of crowds, the long tail, search, Web spam, advertising and social media
2. The mining process
   – Crawling, data cleaning and data anonymization
3. The basic concepts
   – Data statistics, usage mining, link mining, graph mining, finding communities
4. Detailed examples
   – Size of the web, near-duplicate detection, spam detection based on content and links, social media mining, query mining
5. Final remarks

(1) Motivations

Internet and the Web Today

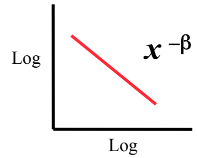
- **Between 1 and 2.5 billion people connected**
  – 5 billion estimated for 2015
- **1.8 billion mobile phones today**
  – 500 million expected to have mobile broadband during 2010
- **Internet traffic has increased 20 times in the last 5 years**
- **Today there are more than 170 million Web servers**
- **The Web is in practice unbounded**
  – Dynamic pages are unbounded
  – Static pages over 20 billion?

Different Views on Data



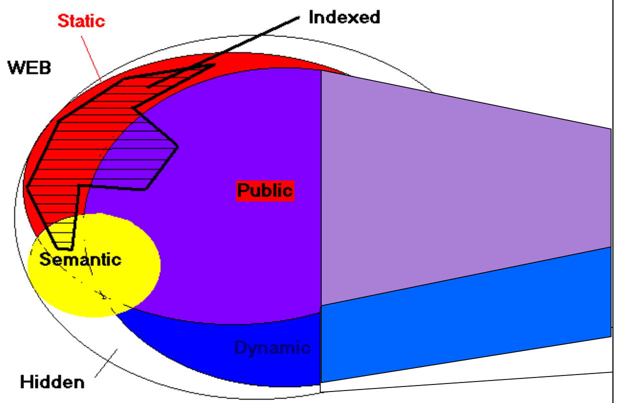
The Web

- Largest public repository of data
- Today, there are more than 200 million Web servers (Jun 2010) and more than 750 million hosts (Apr 2010)
- Well connected graph with out-link and in-link power law distributions

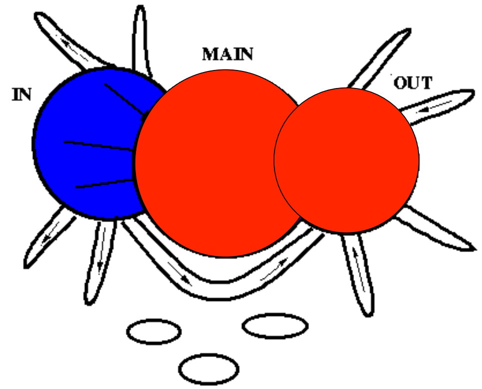


Self-similar &
Self-organizing

The Different Facets of the Web



The Structure of the Web



Web Mining

- **Content:** text & multimedia mining
- **Structure:** link analysis, graph mining
- **Usage:** log analysis, query mining
- **Relate all of the above**
  - Web characterization
  - Particular applications

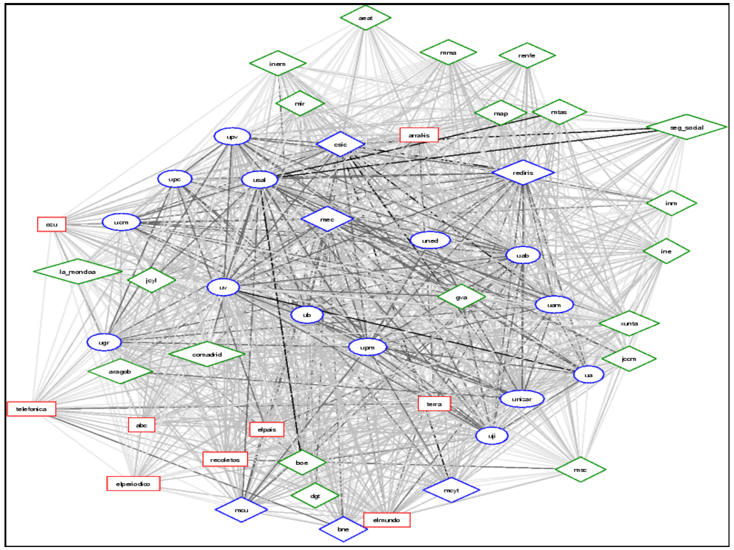
Dynamic

What for?

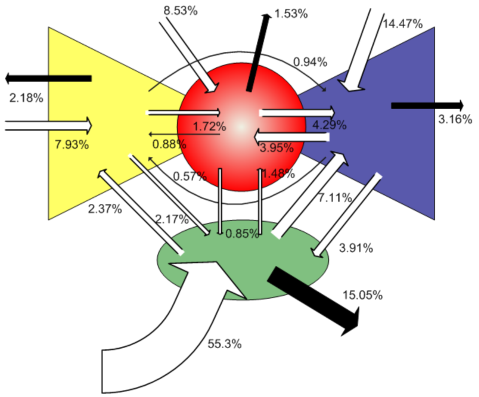
- The Web as an object
- User-driven Web design
- Improving Web applications
- Social mining
- .....

A Few Examples

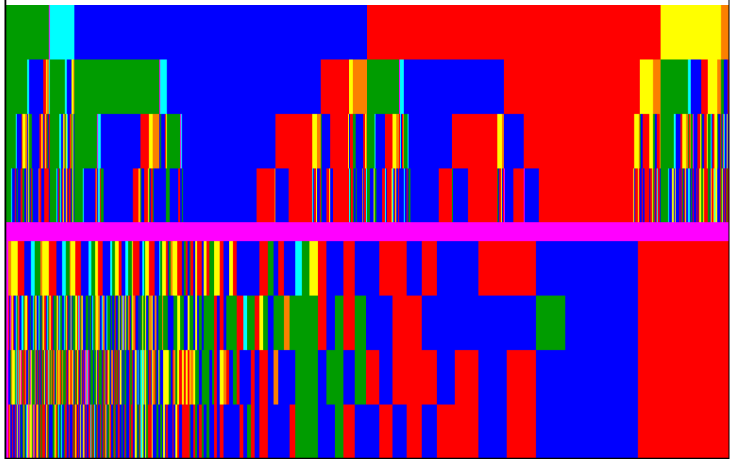
- Web Characterization of Spain
- Link Analysis
- Log Analysis
- Web Dynamics
- Social Mining



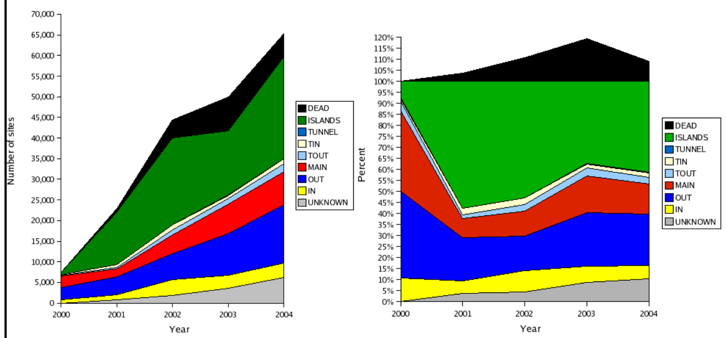
Structure Macro Dynamics



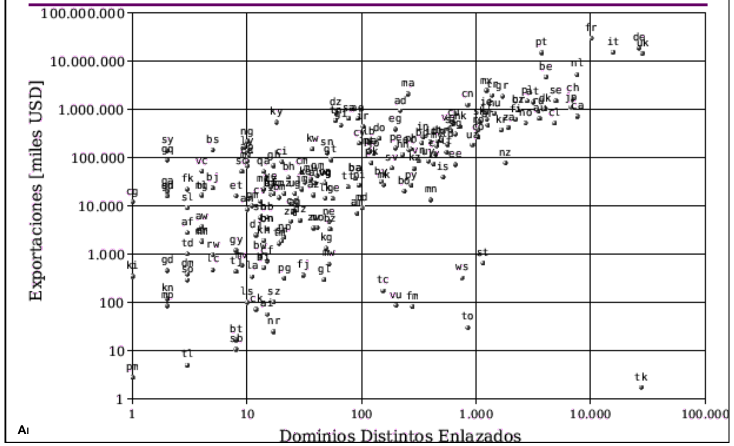
Structure Micro Dynamics



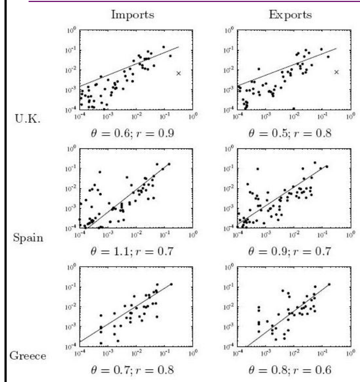
Size Evolution

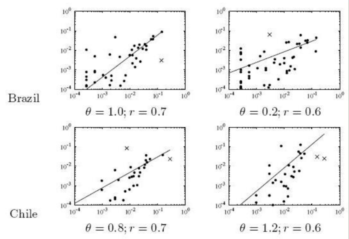


Mirror of the Society



Exports/Imports vs. Domain Links





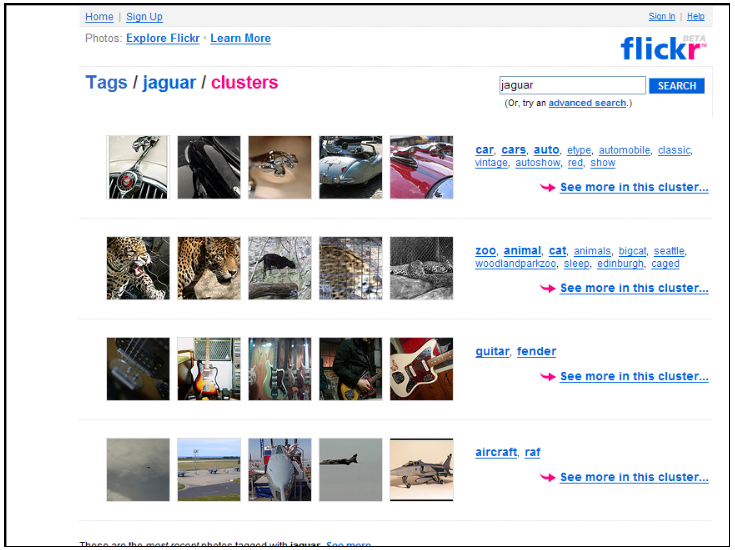
Baeza-Yates & Castillo, WWW2006

The Wisdom of Crowds

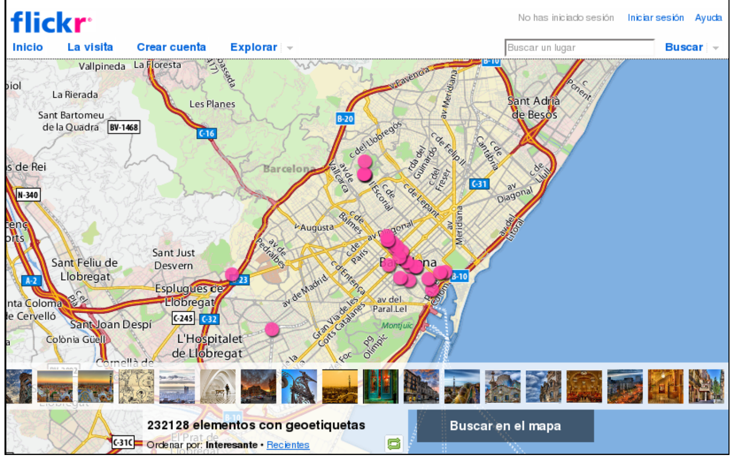
- James Surowiecki, a *New Yorker* columnist, published this book in 2004
  - "Under the right circumstances, groups are remarkably intelligent"
- Importance of diversity, independence and decentralization

Aggregating data

"large groups of people are smarter than an elite few, no matter how brilliant—they are better at solving problems, fostering innovation, coming to wise decisions, even predicting the future".



Flickr: Geo-tagged pictures



The Wisdom of Crowds

- Popularity
- Diversity
- Quality
- Coverage

Long tail

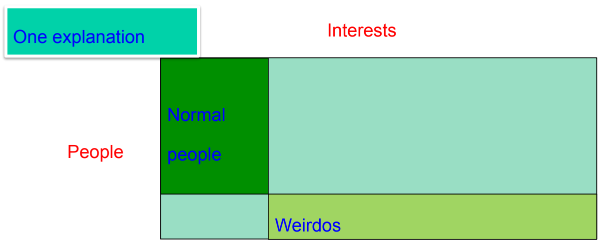
The Long Tail

Explore Flickr through tags

architecture art australia beach birthday blue bw california canada canon china christmas city concert england europe family festival flower flowers food france friends fun germany green italy japan london music nature new newyork night nikon nyc paris park party people portrait red sanfrancisco sky snow spain street summer sunset taiwan travel trip uk usa vacation water wedding white winter

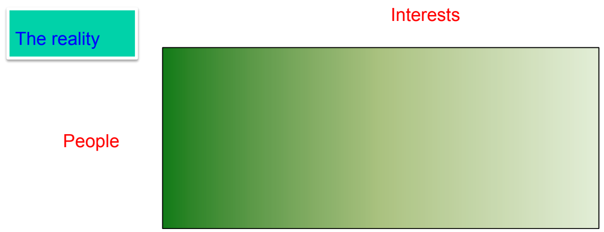
Heavy tail of user interests

- **Many queries, each asked very few times, make up a large fraction of all queries**
  - Movies watched, blogs read, words used …



Heavy tail of user interests

- **Many queries, each asked very few times, make up a large fraction of all queries**
- **Applies to word usage, web page access …**
- **We are all partially eclectic**



Broder, Gabrilovich, Goel, Pang; WSDM 2009

Why the heavy tail matters

- **Not because the worst-sellers make a lot of money**
- **But because they matter to a lot of people**

The Big Challenge for Search

Meet the diverse user needs
given
their poorly made queries
and
the size and heterogeneity of the Web corpus

The Wisdom of Crowds

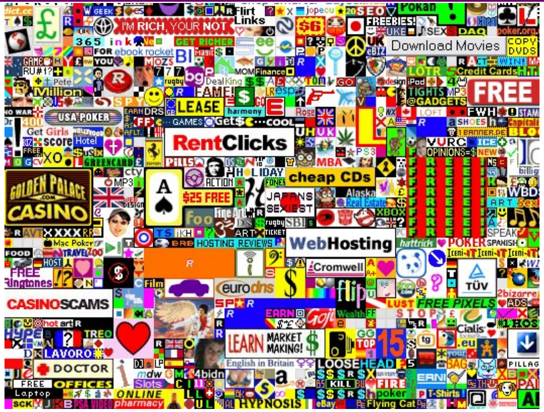
- Crucial for Search Ranking
- Text: Web Writers & Editors
  - not only for the Web!
- Links: Web Publishers
- Tags: Web Taggers
- Queries: All Web Users!
  - Queries and actions (or no action!)

What is in the Web?

- Information
- Porn

- + On-line casinos + Free movies + Cheap software + Buy a MBA diploma + Prescription - free drugs + V!-4-gra + Get rich now now now!!!

What is in the Web?



Spam is an Economic Activity

- Depending on the goal and the data spam is easier to generate
- Depending on the type & target data spam is easier to fight
- Disincentives for spammers?
  - Social
  - Economical
- Exploit the power of social networks and their work

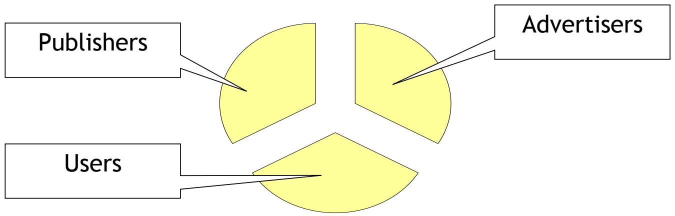
Current challenges (1)

- **Scraper spam**
  - Copies good content from other sites, adds monetization (most often Google AdSense)
  - Hard to identify at the page level (indistinguishable from original source), monetization not reliable clue (there is actually good content on the web that uses AdSense/YPN!)
- **Synthetic text**
  - Boilerplate text, randomized, built around key phrases
  - Avoids duplicate detection
- **Query-targeted spam**
  - Each page targets a single tail query (anchortext, title, body, URL). Often in large auto-constructed hosts, host-level analysis most helpful
- **DNS spam**

Current challenges (2)

- **Blog spam**
  - Continued trend toward blog “ownership” rather than comment spam
  - Orthogonal to other categories (scrapers, synthesizers). Just a hosting technique, plus exploiting blog interest
- **Example:**
  - 68,000 blogspot.com hosts all generated by the same spammer
    - 1) nursingschoolresources.blogspot.com
      2) transplantresources.blogspot.com
      ..
      67,798) beachesresourcesforyou.blogspot.com
      67,799) startrekresourcesforyou.blogspot.com

Content match = meeting of Publishers, Advertisers, Users



and Spammers! Grrr...

Contextual ads

Contextual ads

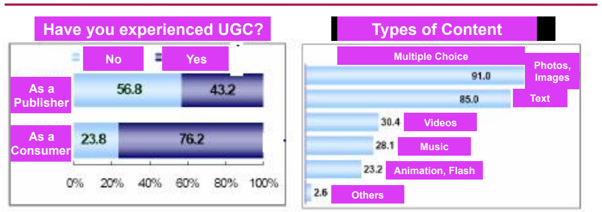
Click spam

- **Rival click fraud:** Rival of advertising company employs clickers for clicking through ads to exhaust budget
- ~~**Publisher click fraud:** Publisher employs clickers to reap per-click revenue from ads shown by search firm~~
- **Bidder click fraud:** Keyword bidders employ clickers to raise rate used in (click-thru-rate * bid) ranking used to allocate ad space in search engines (or to pay less!)

Other Possible Ad Spam

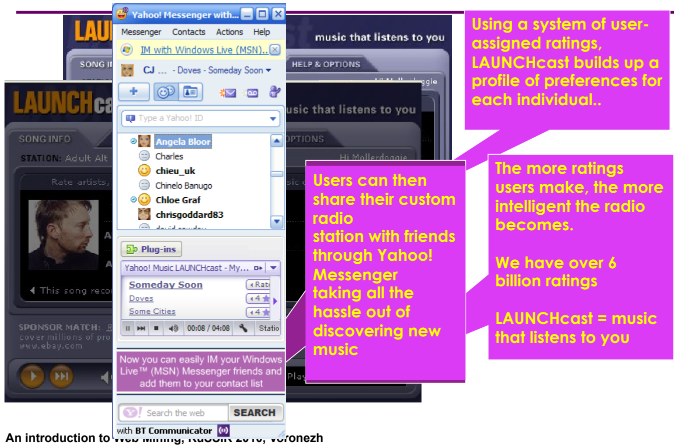
- **Rival buys misleading or fraudulent ads**
  - Queries
  - Bids
  - Ads
- **Rival submits queries that brings up competitor ad but without clicking on it**
  - *Reduces* rival's CTR and hence its ranking for ad space

Internet UGC (User Generated Content)

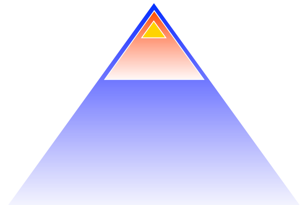


Source National Internet Development Agency Report in June, 2006 (South Korea)

Simple acts create value and opportunity



Community Dynamics



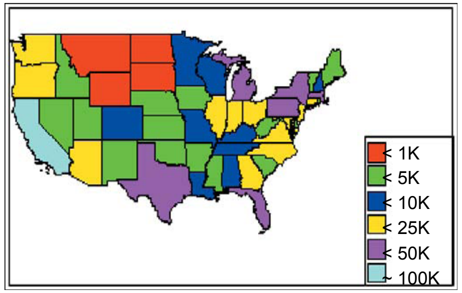
1 creators

10 synthesizers

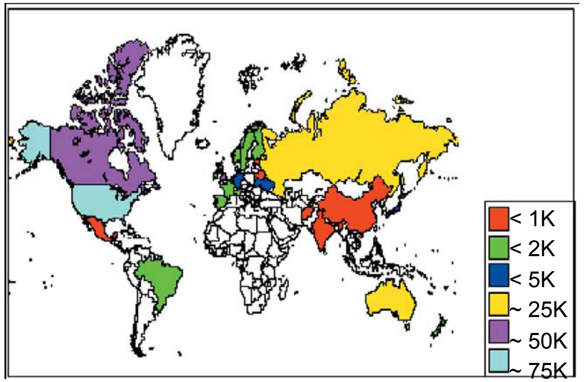
100 consumers

Next generation products will blur distinctions between Creators, Synthesizers, and Consumers
Example: Launchcast
Every act of consumption is an implicit act of production that requires no incremental effort…
Listening itself implicitly creates a radio station…

Community Geography: Live Journal bloggers in US



LJ bloggers world-wide



Who are they?

Age	%	Representative interests
1 to 3	0.5	treats, catnips, daddy, mommy, purring, mice, playing, napping, scratching, milk
13 to 15	3.5	webdesigning, Jeremy Sumpter, Chris Wilson, Emma Watson, T. V., Tom Felton, FUSE, Adam Carson, Guyz, Pac Sun, mall, going online
16 to 18	25.2	198{6,7,8}, class of 200{4,5}, dream street, drama club, band trips, 16, Brave New Girl, drum major, talkin on the phone, highschool, JROTC
19 to 21	32.8	198{3,5}, class of 2003, dorm life, frat parties, college life, my tattoo, pre-med
22 to 24	18.7	198{1,2}, Dumbledore's army, Midori sours, Long island iced tea, Liquid Television, bar hopping, disco house, Sam Adams, fraternity, He-Man, She-Ra
25 to 27	8.4	1979, Catherine Wheel, dive bars, grad school, preacher, Garth Ennis, good beer, public radio
28 to 30	4.4	Hal Hartley, geocaching, Camarilla, Amtgard, Tivo, Concrete Blonde, motherhood, SQL, TRON
31 to 33	2.4	my kids, parenting, my daughter, my wife, Bloom County, Doctor Who, geocaching, the prisoner, good eats, herbalism
34 to 36	1.5	Cross Stitch, Thelema, Tivo, parenting, cubs, role-playing games, bicycling, shamanism, Burning Man
37 to 45	1.6	SCA, Babylon 5, pagan, gardening, Star Trek, Hogwarts, Macintosh, Kate Bush, Zen, tarot
46 to 57	0.5	science fiction, wine, walking, travel, cooking, politics, history, poetry, jazz, writing, reading, hiking
> 57	0.2	death, cheese, photography, cats, poetry

(2) The Mining Process

Yahoo! Research

The Process

- Data recollection: crawling, log keeping
- Data cleaning and anonymization
- Data statistics and data modeling

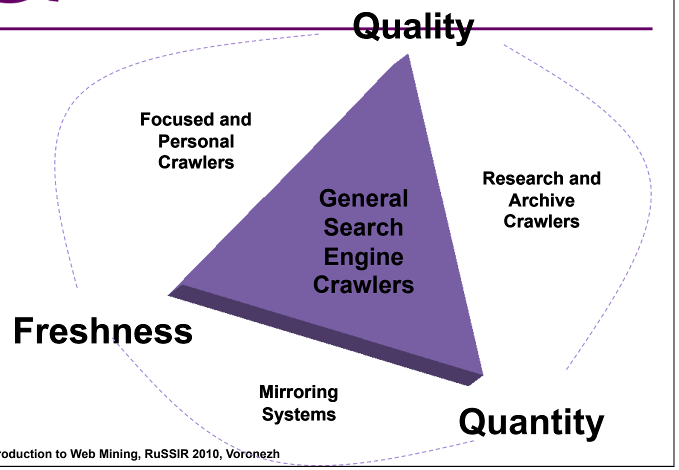
Data Recollection

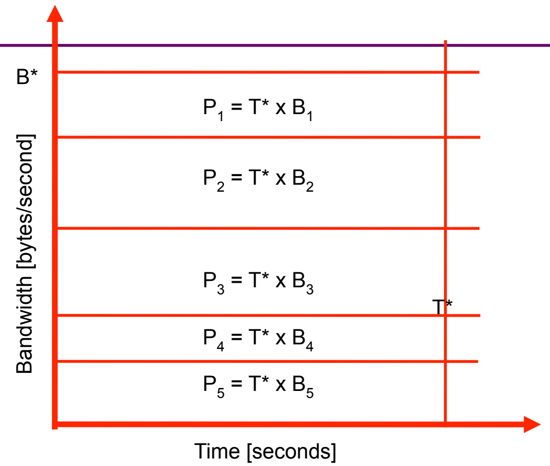
- Content and structure: Crawling
- Usage: Logs
  - Web Server logs
  - Specific Application logs

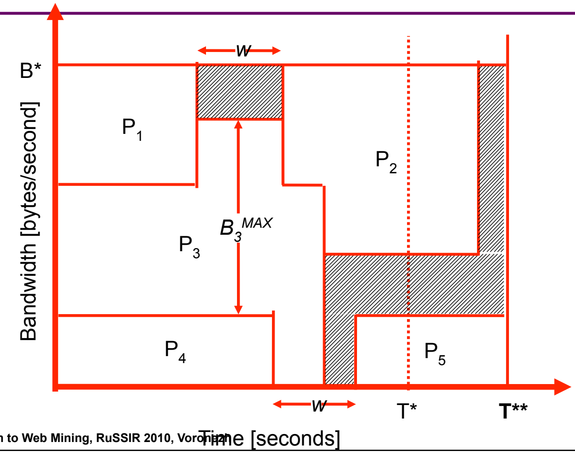
Crawling

- NP-Hard Scheduling Problem
- Different goals
- Many Restrictions
- Difficult to define optimality
- No standard benchmark

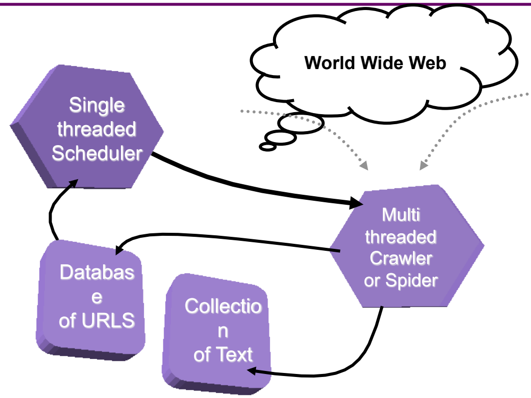
Crawling Goals

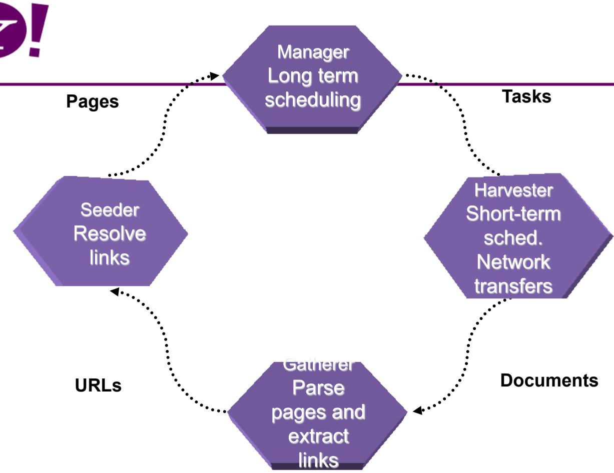


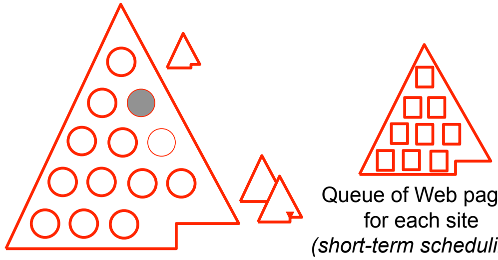




Software Architecture







Queue of Web pages for each site *(short-term scheduling)*

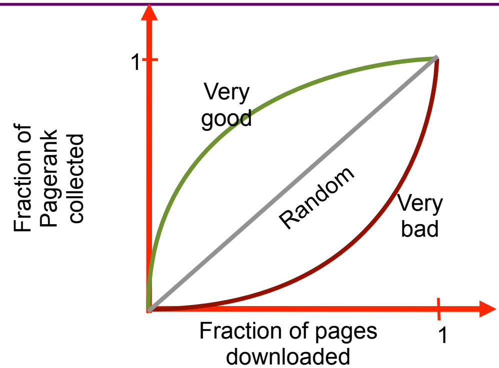
Queue of Web sites *(long-term scheduling)*

Formal Problem

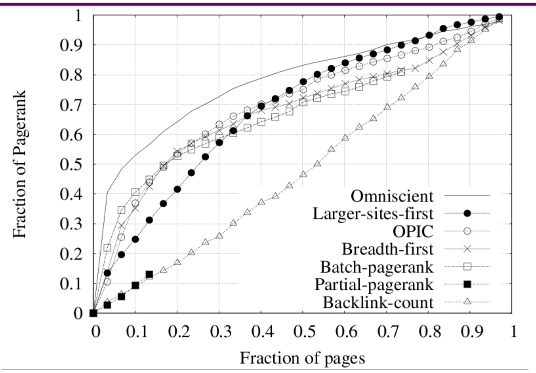
- **Find a sequence of page requests *(p,t)* that:**
  - Optimizes a function of the volume, quality and freshness of the pages
  - Has a bounded crawling time
  - Fulfils politeness
  - Maximizes the use of local bandwidth
- **Must be on-line: how much knowledge?**

Crawling Heuristics

- **Breadth-first**
- **Ranking-ordering**
  - PageRank
- **Largest Site-first**
- **Use of:**
  - Partial information
  - Historical information
- **No Benchmark for Evaluation**

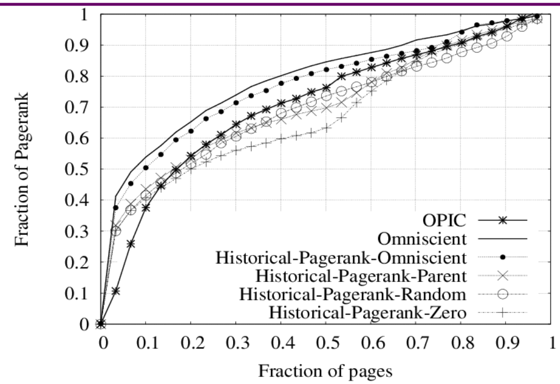


No Historical Information

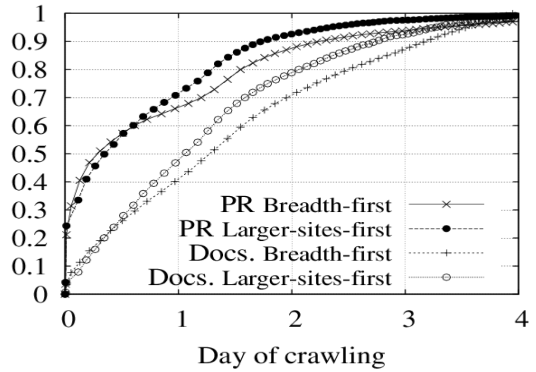


Baeza-Yates, Castillo, Marin & Rodriguez, WWW2005

Historical Information



Validation in the Greek domain



Data Cleaning

- **Problem Dependent**
- **Content: Duplicate and spam detection**
- **Links: Spam detection**
- **Logs: Spam detection**
  - Robots vs. persons

Data Processing

- **Structure: content, links and logs**
  - XML, relational database, etc.
- **Usage mining:**
  - Anonymize if needed
  - Define sessions

Data Characteristics

- **Yahoo! as a Case Study**
  - Data Volume
  - Data Types

Yahoo! World

- **Search**
  - Yahoo! Image,
  - Yahoo! Video,
  - Yahoo! Local,
  - Yahoo! News,
  - Yahoo! Shopping Search,
- **Communication**
  - Yahoo! Mail,
  - Yahoo! Messenger,
  - My Web,
  - Yahoo! Personals,
  - Yahoo! 360°,
  - Yahoo! Photos,
  - Flickr, Delicious,
  - Yahoo! Answers
- **Content:**
  - Yahoo! Sports,
  - Yahoo! Finance,
  - Yahoo! Music,
  - Yahoo! Movies,
  - Yahoo! News,
  - Yahoo! Games,
  - My Yahoo!
- **Mobile:**
  - Yahoo! Mobile
- **Commerce:**
  - Yahoo! Shopping,
  - Yahoo! Autos,
  - Yahoo! Auctions,
  - Yahoo! Travel,
- **Small Business:**
  - Yahoo! Small Business
  - Yahoo! Domains,
  - Yahoo! Web Hosting,
  - Yahoo! Merchant Solutions,
  - Yahoo! Business Email,
  - HotJobs
- **Advertising:**
  - Yahoo! Search Marketing
  - Yahoo! Publisher Network.

Yahoo! Numbers (2006)

24 languages, 20 countries

- **> 4 billion page views per day (largest in the world)**
- **> 500 million unique users each month (half the Internet users!)**
- **> 250 million mail users (1 million new accounts a day)**
- **95 million groups members**
- **7 million moderators**
- **4 billion music videos streamed in 2005**

- **20 Pb of storage (20M Gb)**
  - US Library of congress every day (28M books, 20TB)
- **12 Tb of data processed per day**
- **7 billion song ratings**
- **2 billion photos stored**
- **2 billion Mail+Messenger sent per day**

Crawled Data

- **WWW** heterogeneous, large, dangerous
  - Web Pages & Links
  - Blogs
  - Dynamic Sites
- **Sales Providers (Push)** very high quality & structure, expensive, sparse, safe
  - Advertising
  - Items for sale: Shopping, Travel, etc.
- **News Index** high quality, sparse, redundant
  - RSS Feeds
  - Contracted information

Produced data

- **Yahoo's Web** homogeneous, high quality, safer, highly structured
  - Ygroups
  - YCars, YHealth, Ytravel
- **Produced Content** Trusted, high quality, sparse
  - Edited (news)
  - Purchased (news)
- **Direct Interaction:** Ambiguous semantics? trust? quality?
  - Tagged Content
    - Object tagging (photos, pages, ?)
    - Social links
  - Question Answering

"Information Games" (e..g. www.espgame.org)

Observed Data

- **Query Logs**
  - spelling, synonyms, phrases (named entities), substitutions good quality, sparse, power law
- **Click-Thru**
  - relevance, intent, wording good quality, sparse, mostly safe
- **Advertising**
  - relevance, value, terminology Trusted, high quality, homogeneous, structured
- **Social**
  - links, communities, dialogues... trust? quality?

Data anonymization

- **American Online (AOL) query log released in August 2006**
- **Objective was to contribute to IR research**
- **Query log rough statistics**
  - 20 million queries
  - 650 K users
  - from over 3 months
- **Social security numbers, credit card numbers, driver license numbers, etc.**
- **Possible to uniquely identify many users by combining information from queries and yellow pages, etc.**
- **Big media scandal, big damage to AOL and the privacy of its users**

A typical query log

- Entries of the format:

<cookie, query, rank, clickURL, timeStamp, IP, country,...>

Anonymizing query logs

- [Adar 2007]
- Argue that anonymization is potentially possible
- Two main techniques:
  - Eliminate infrequent queries
  - Splitting personalities
- Additionally:
  - Eliminate identifying information (SSN, credit card numbers, etc.)

Anonymizing query logs

- Eliminate infrequent queries:
- Keep only queries generated by a large number of users
- Computationally possible using counters
- How to do it on-the-fly?
- Long tail disappears!

Online elimination of infrequent queries

- Background: How to split a secret among n people so that every coalition of *k* persons can access the secret?
- Answer: Let the secret be the coefficients of a *(k-1)*-degree polynomial $f(x) = a_{k-1}x^{k-1} + . . . + a_1x + a_0$
- For the *i*-th person, select a number x_i, and give to the person the pair $(x_i , f(x_i))$
- Any k persons can cooperate and recover the polynomial, while no *k-1* persons can recover it

Online elimination of infrequent queries

- Straightforward application in eliminating infrequent queries
- A query *q* is decoded as a *(k-1)*-degree polynomial f_q
- For a person u_i who makes the query *q*, print $(u_i, f_q(u_i))$
- If *k* or more people type the query *q*, it is possible to decrypt *q*!

Split personalities

- Split the queries of the same user into sessions
- E.g., queries about food recipes, sport results, buying books, music, etc.
- Assign each of those sessions to a di erent virtual user
- Released query log can be still useful for many applications
- More difficult to identify users by combining queries
- Finding similar queries and finding query sessions is quite hard problem

Anonymizing query logs: negative resuls

- [Kumar et al., 2007]
- Anonymization via token-based hashing:
- The query is split into terms and each term is hashed to a token
- Co-occurrence analysis and frequency analysis can be used to reveal the query terms
- Assume access to an unencrypted query log
- Query term statistics remain constant across different query logs
- Provide practical graph-matching algorithms and analysis of real query logs

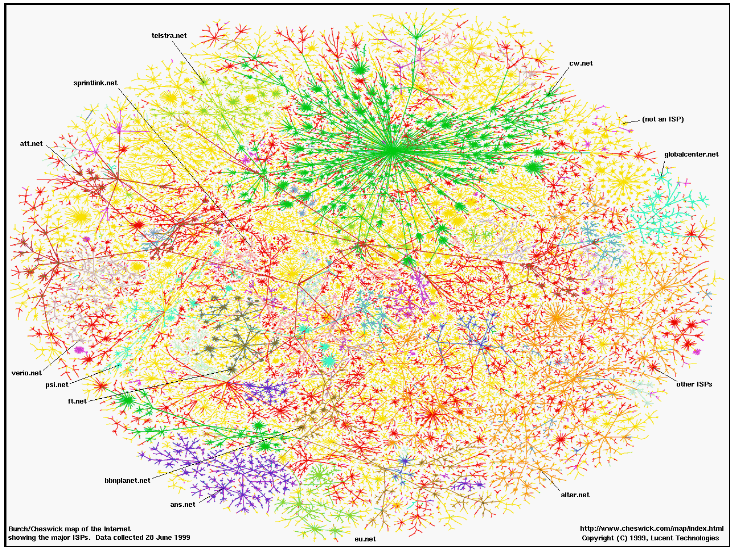
(3) Main Concepts

Topics

- Data statistics and data modelling
- Usage mining
- Link analysis
- Graph mining
- Finding communities

Data statistics and data modeling

- Graph structures
- Degree distribution
- Community structure
- Diameter and other properties



Degree distribution

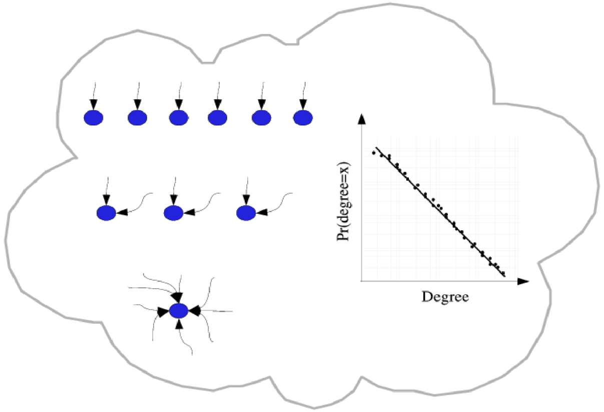
- Consider a graph $G=(V,E)$
- C_k the number of vertices u with degree d(u) = k

$$C_k = c\, k^{-\gamma} \quad \text{with} \quad \gamma>1,$$

$$\log(C_k) = \log(c) - \gamma \log(k)$$

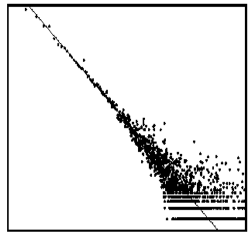
- So, plotting $\log(C_k)$ versus $\log(k)$ gives a straight line with slope $-\gamma$
- Heavy-tail distribution: there is a non-negligible fraction of nodes that has very high degree (hubs)
- Scale-free: no characteristic scale, average is not informative

Degree distribution

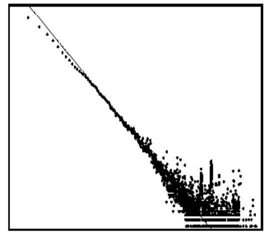


Degree distribution

In-degree distributions of web graphs within national domains



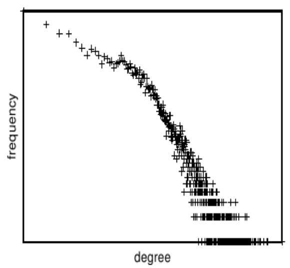
Greece



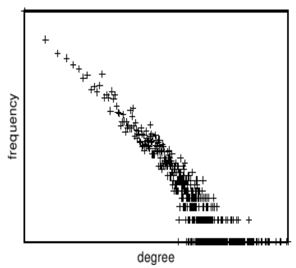
Spain

Degree distribution

...and more “straight” lines...



in-degrees of UK hostgraph



out-degrees of UK hostgraph

Community structure

- **Intuitively a subset of vertices that are more connected to each other than to other vertices in the graph**
- **A proposed measure is clustering coefficient**

$$C_1 = \frac{3 \times \text{ number of triangles in the network}}{\text{number of connected triples of vertices}}$$

- Captures “transitivity of clustering”
- If *u* is connected to *v* and *v* is connected to *w*, it is also likely that *u* is connected to *w*

Community structure

- **Alternative definition.**
- **Local clustering coefficient:**

$$C_i = \frac{\text{number of triangles connected to vertex } i}{\text{number of triples centered at vertex } i}$$

- Global clustering coefficient:

$$C_2 = 1/n \sum_i C_i$$

- Community structure is captured by large values of clustering coefficient

Small diameter

- **Diameter of many real graphs is small (e.g., *D* = *6* is famous)**
- **Proposed measures:**
  - Hop-plots: plot of $|N_h(u)|$, the number of neighbors of *u* at distance at most *h*, as a function of *h*
  - [M. Faloutsos, 1999] conjectured that it grows exponentially and considered hop exponent
  - Effective diameter: upper bound of the shortest path of 90% of the pairs of vertices
  - Average diameter: average of the shortest paths over all pairs of vertices
  - Characteristic path length: median of the shortest paths over all pairs of vertices

Other properties

- **Degree correlations**
- **Distribution of sizes of connected components**
- **Resilience**
- **Eigenvalues**
- **Distribution of motifs**
- **... all very different than predicted for random graphs**

- **Properties of evolving graphs [Leskovec et al., 05]**
  - Densification power law
  - Diameter is shrinking

Power-law distributions

- *"A brief history of generative models for power laws and log-normal distributions"* [Mitzenmacher, 04]
- A random variable *X* has power-law distribution, if

$$Pr[X>x] \propto cx^{-\alpha} \text{ for } c > 0 \text{ and } \alpha > 0$$

- A random variable *X* has Pareto distribution, if

$$Pr[X>x] = (x/k)^{-\alpha} \quad \text{for } k > 0,\ \alpha > 0, \text{ and } X > k$$

- On a log-log plot straight line with slope $-\alpha$

A process that generates a power-law

- Preferential attachment
- The main idea is that "the rich get richer"
  - First studied by [Yule, 1925] to suggest a model of why the number of species in genera follows a power-law
  - Generalized by [Simon, 1955]
    - applications in distribution of word frequencies, population of cities, income, etc.
  - Revisited in the 90s as a basis for Web-graph models [Barabasi and Albert, 1999, Broder et al., 2000, Kleinberg et al., 1999]

Preferential attachement

- The basic theme:
  - Start with a single vertex, with a link to itself
  - At each time step a new vertex *u* appears with out-degree 1 and gets connected to an existing vertex *v*
  - With probability $\alpha < 1$, vertex *v* is chosen uniformly at random
  - With probability $1-\alpha$, vertex *v* is chosen with probability proportional to its degree
  - Process leads to power law for the in-degree distribution, with exponent $(2-\alpha)/(1-\alpha)$

Log-normal distribution

- Random variable X has log-normal distribution, if Y=log(X) has normal distribution
- Always finite mean and variance
- But also appears as a straight line on a log-log plot
  (for small values of x)
- Multiplicative processes tend to give log-normal distributions:
  - The product of two log-normally distributed independent random variables follows a log-normal distribution

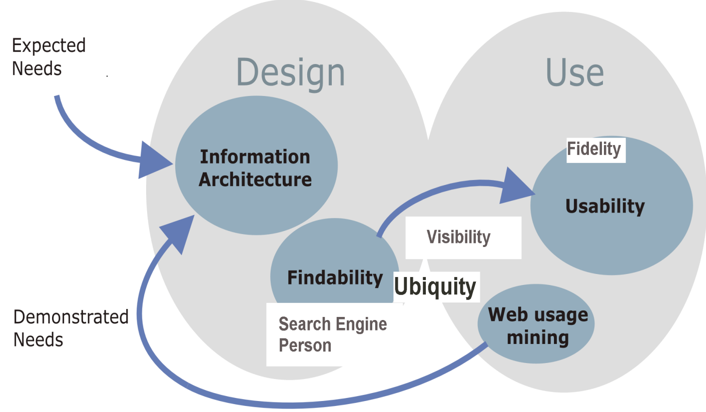
Power law or log-normal?

- Distribution of income
- Start with some income X_0
- At time t, with probability 1/3 double the income, with probability 2/3 cut income at half
- Then income distribution is log-normal
  (multiplicative process)
- But... assume a "reflective barrier":
  - At X_0 maintain same income with probability 2/3
- ... a power law!

Usage mining for...

- **User Driven Design**
  - Information Scent
  - The Web Site that the Users Want
  - The Web Site that You should Have
  - Improve content & structure
- **Improved Web Search: index layout, ranking**
- **Bootstrap of pseudo-semantic resources**

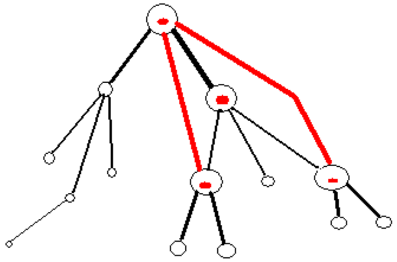
Web Design



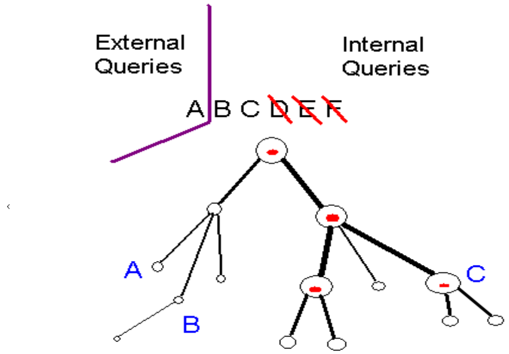
User-driven design

- ***User-driven design***
  - Best example: Yahoo!
- **Navigational log analysis**
  - Site reorganization
- **Query log analysis**
  - Information Scent
  - Content that is missing: market niches

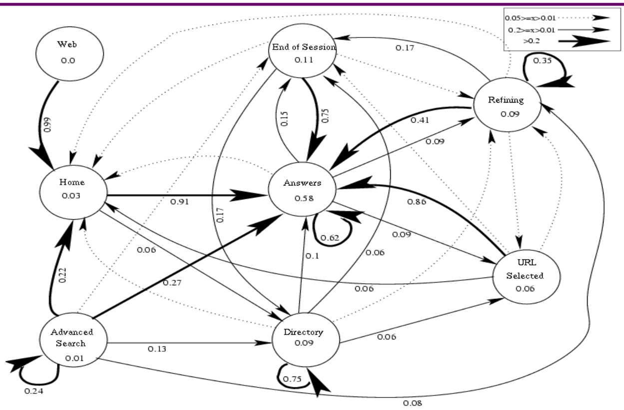
Navigation Mining



Web Site Query Mining



User Modeling



Link analysis

- Infer properties of Web entities based on their connectivity / link structure of graph structures they belong to
- Such properties can be importance of nodes or similarity between nodes
- Mostly focused on Web pages, but ideas apply to many domains: social networks, query logs, etc.
- Prestige, centrality, co-citation, PageRank, HITS

Social sciences and bibliometry

"...we are involved in an 'infinite regress': [an actor's status] is a function of the status of those who choose him; and their [status] is a function of those who choose them, and so ad infinitum"

[Seeley, 1949]

Prestige

- Consider a graph $G=(V,E)$
- $E[u,v] = 1$ if there is a link from u to v
- $E[u,v] = 0$ otherwise
- p a prestige vector: $p[u]$ the prestige score of node u

$$p' = E^T p$$

because

$$p[u] = \sum_v E[v,u]\, p[u] = \sum_v E^T[u,v]\, p[u]$$

- After each iteration normalize by setting $||p|| = 1$
- p converges to the principal eigenvector of E^T

Centrality

- Importance notion based on centrality
- Used by epidimiology, social-network analysis, etc.: removing a central node disconnects the graph to a big extend

- $d(u,v)$ the shortest-path distance between u and v
- $r(u) = \max_v d(u,v)$ *radius* of node u
- $\arg\min_u r(u)$ *center* of the graph

- Various other notions of centrality in the literature

Co-citation

- Measure of similarity between nodes
- If nodes v and w are both linked by node u, then they are co-cited
- If E is the adjacency matrix of the graph, the number of nodes that co-cite both v and w is

$$p[u] = \sum_u E[u,v]\, E[u,w] = \sum_u E^T[v,u]\, E[u,w] = (E^T E)[v,w]$$

- Thus similarity is captured in the entries of matrix $E^T E$

PageRank

- [Brin and Page, 1998]
- Algorithm suggested $\alpha$for ranking results in web search
- An authority score is assigned to each Web page
- Authority scores independent of the query
- Authority scores corresponds to the stationary distribution of a random walk on the graph:
  - With probability α follow a link in the graph
  - With probability $1-\alpha$ go to a node chosen uniformly at random (teleportation)
- Random walk also known as random surfer model

PageRank

- Let *E* be the adjacency matrix of the graph, and *L* the row-stochastic version of *E*
- Each row of *E* is normalized so that it sums to *1*
- Authority score defined by

$$p_{(i+1)} = L^T p_{(i)}$$

- problematic if the graph is not strongly connected, So:

$$p_{(i+1)} = \alpha L^T p_{(i)} + (1-\alpha) 1/n\ 1$$

- where *1* is the matrix with all entries equal to *1*
- and $\alpha \in [0,1]$, common value $\alpha = 0.85$

PageRank variants and enchancements

- Personalized PageRank
  - Teleportation to a set of pages defining the preferences of a particular user
- Topic-sensitive PageRank [Haveliwala 02]
  - Teleportation to a set of pages defining a particular topic
- TrustRank [Gyöngyi 04]
  - Teleportation to “trustworthy” pages
- Many papers on analyzing PageRank and numerical methods for efficient computation

HITS

- [Kleinberg 1998]
- Exploit the intuition that there are:
  - pages that contain high-quality information (authorities)
  - pages with good navigational properties (hubs)

Good hubs point to good authorities and good authorities are pointed by good hubs

HITS algorithm

- Given a query *q*
- Use a standard wen IR system to find a set of pages *R* relevant to *q* (*root set*)
- Expand to the set of pages connected to *R* (*expanded set*) and form the graph *G=(V,E)*
- *a* authority vector: *a[u]* the authority score of node *u*
- *h* hub vector: *h[u]* the hub score of node *u*

$$a = E^T h \qquad h = E\,a$$

- *a* converges to the principal eigenvector of E^TE
- *h* converges to the principal eigenvector of EE^T

HITS

- HITS is related to SVD on the graph matrix E
- non-principal eigenvectors provide different topics
- HITS sensitive to local-topology
- PageRank is more stable – due to trandom jump step
- Researchers attempted to make HITS more stable
  - SALSA stochastic algorithm for link analysis [Lempel and Moran, 01]:
  - A random surfer model in which the surfer follows alternatively random inlinks and outlinks
  - [Ng et al. 01] introduce a random jump step in the HITS model

Discussion

- HITS introduces the notion of hub, which does not exist in PageRank
- HITS is query sensitive
- PageRank does not depend on the query; thus the authority scores can be pre-computed
- Nepotism, two-host nepotism, and clique attacks

Graph Mining

- Keep an eye on efficiency
- Web graphs are huge and any computation on them should be very efficient
- Data stream algorithms for
  - Computing the clustering coefficient
  - Counting the number of triangles
  - Estimating the diameter of a graph

Clustering coefficient

$$C_1 = \frac{3 \times \text{ number of triangles in the network}}{\text{number of connected triples of vertices}}$$

- How to compute it?
- How to compute the number of triangles in a graph?
- Assume that the graph is very large, stored on disk

Counting triangles

- Brute-force algorithm is checking every triple of vertices
- Obtain an approximation by sampling triples
- Let *T* be the set of all triples, and
- T_i the set of triples that have *i* edges, *i = 0, 1, 2, 3*
- By Chernoff bound, to get an ε-approximation, with probability 1 − δ, the number of samples should be

$$N \geq O\left(\frac{|T|}{|T_3|}\frac{1}{\epsilon^2}\log\frac{1}{\delta}\right)$$

- But |T| can be large compared to $|T_3|$

Counting triangles

- SampleTriangle Algorithm [Buriol et al., 2006]
- Incidence stream model – all edges incident on the same edge are consecutive on the disk
- Three pass algorithm:
- Pass 1: Count the number of paths of length 2
- Pass 2: Choose one path (a,u,b) uniformly at random
- Pass 3: If (a,b)∈E return 1 o/w return 0

Counting triangles

- The previous idea can be also applied to:
  - Count triangles when edges are stored in arbitrary order
  - Obtain one-pass algorithm
  - Count other minors

Diameter

- How to compute the diameter of a graph?
- Matrix multiplication in $O(n^{2.376})$ time, but $O(n^2)$ space
- BFS from a vertex takes $O(n + m)$ time,
- but need to do it from every vertex, so $O(mn)$
- Resort to approximations again

Diameter

- How to compute the diameter of a graph?
- Matrix multiplication in $O(n^{2.376})$ time, but $O(n^2)$ space
- BFS from a vertex takes $O(n + m)$ time,
- but need to do it from every vertex, so $O(mn)$
- Resort to approximations again

An introduction to Web Mining, RuSSIR 2010, Voronezh

Approximating the diameter

- [Palmer et al., 2002], see also [Cohen, 1997]
- Define:
- Individual neighborhood function

$$N(u, h) = |\{v \mid d(u, v) \le h\}|$$

- Neighborhood function

$$N(h) = |\{(u, v) \mid d(u, v) \le h\}| = \sum_u N(u, h)$$

- With $N(h)$ can obtain diameter, effective diameter, etc.

Approximating the diameter

- Define: $M(u, h) = \{v \mid d(u, v) \le h\}$, e.g., $M(u, 0) = \{u\}$
- Algorithm based on the idea that

$$x \in M(u, h) \text{ if } (u, v) \in E \text{ and } x \in M(v, h-1)$$

ANF [Palmer et al., 2002]

$M(u, 0) = \{u\}$ for all $u \in V$
for each distance h do
  $M(u, h) = M(u, h-1)$ for all $u \in V$
  for each edge (u, v) do
    $M(u, h) = M(u, h) \cup M(v, h-1)$

- Keep $M(u, h)$ in memory, make a passes over the edges
- How to maintain $M(u, h)$?

Approximating the diameter

- How to maintain *M(u, h)* that it counts distinct vertices?
- The problem of counting distinct elements in data streams
- ANF uses the sketching algorithm of
  - [Flajolet and Martin, 1985] with *O(log n)* space
  - (but other counting algorithms can be used [Bar-Yossef et al., 2002])
- What if the *M(u, h)* sketches do not fit in memory?
- Split *M(u, h)* sketches into in-memory blocks,
  - load one block at the time,
  - and process edges from that block

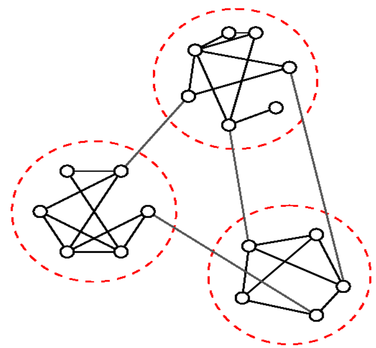
Finding communities

- A set of related Web pages
- A group of scientists collaborating with each other
- A set of blog posts discussing a specific topic
- A set of related queries
- Can be used for improving relevance of search, recommendations, propagating an idea, advertising a product, etc.
- Usually formulated as a graph clustering problem

Graph clustering

- Graph *G = (V, E)*
- Edge *(u, v)* denotes similarity between *u* and *v*
  - weighted edges can be used to denote degree of similarity
- We want to partition the vertices in clusters so that:
  - vertices within clusters are well connected, and
  - vertices across clusters are sparsely connected
- Most graph partitioning problems are NP hard

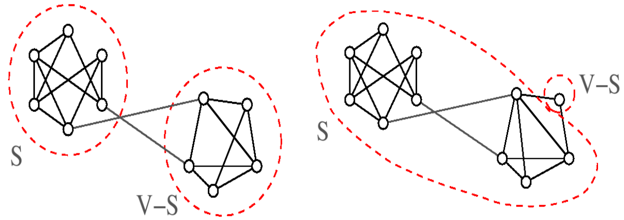
Graph clustering



Measuring connectivity

- **Minimum cut: The minimum number of edges whose removal disconnects the graph**

  $c(S) = \min_{S \subseteq V} |\{(u,v) \in E$ s.t. $u \in S$ and $v \in V\text{-}S\}|$



Graph expansion

- **Normalize the cut by the size of the smallest component**
- **Define cut ratio**

$$\alpha(G, S) = \frac{c(S)}{\min\{|S|, |V - S|\}}$$

- And graph expansion

$$\alpha(G) = \min_S \frac{c(S)}{\min\{|S|, |V - S|\}}$$

- Other similar normalized criteria have been proposed
- Related to the eigenvalues of the adjacency matrix of the graph, thus with the expansion properties of the graph

Spectral analysis

- **Let *A* be the adjacency matrix of the graph *G***
- **Define the Laplacian matrix of *A* as**

  $L = D - A,$

- $D = diag(d_1, \ldots, d_n)$, **a diagonal matrix**
- d_i **the degree of vertex *i***

$$L_{ij} = \begin{cases} d_i & \text{if } i = j \\ -1 & \text{if } (i,j) \in E, i \neq j \\ 0 & \text{if } (i,j) \notin E, i \neq j \end{cases}$$

- *L* is symmetric positive semidefinite
- The smallest eigenvalue of *L* is $\lambda_1 = 0$, with
- corresponding eigenvector $w_1 = (1, 1, \ldots, 1)^T$

Spectral analysis

- **For the second smallest eigenvector λ_2 of L**

$$\lambda_2 = \min_{\substack{\mathbf{x}^T \mathbf{w}_1 = 0 \\ ||\mathbf{x}|| = 1}} \mathbf{x}^T L \mathbf{x} = \min_{\sum x_i = 0} \frac{\sum_{(i,j) \in E} (x_i - x_j)^2}{\sum_i x_i^2}$$

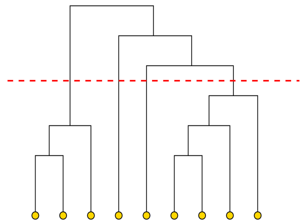
- Corresponding eigenvector w_2 is called Fielder vector
- The ordering according to the values of w_2 will group similar (connected) vertices together
- Physical interpretation: The stable state of springs placed on the edges of the graph, when graph is forced to 1 dimension

Spectral partition

- Partition the nodes according to the ordering induced by the Fielder vector
- Some partitioning rules:
- Bisection: use the median value in w_2
- Cut ratio: find the partition that minimizes
- Sign: Separate positive and negative values
- Gap: Separate according to the largest gap in the values of w_2
- Spectral partition works very well in practice
- However, not scalable

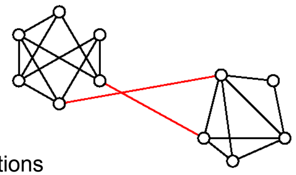
Top down algorithms

- [Newman and Girvan, 2004]
- A set of algorithms based on removing edges from the graph, one at a time
- The graph gets progressively disconnected, creating a hierarchy of communities



Top down algorithms

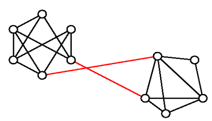
- Select edge to remove based on "betweenenss"

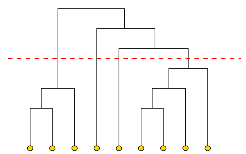


- Three definitions
- Shortest-path betweeness: Number of shortest paths that the edge belongs to
- Random-walk betweeness: Expected number of paths for a random walk from u to v
- Current-flow betweeness: Resistance derived from considering the graph as an electric circuit

Generic top-down algorithm

- Top down
- Compute betweeness value of all edges
- [Recompute betweeness vlaue of all remaining edges]
- Remove the edge with the highest betweeness
- Repeat until no edges left

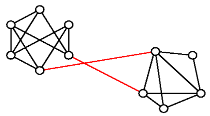


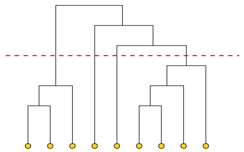


Modularity measure

- How to pick the right clustering from the whole hierarchy?
- Modularity measure [Newman and Girvan, 2004]
- Compared with a “random clustering”

- Direct optimization of modularity measure by
  - Agglomerative [Newman and Girvan, 2004] Spectral [White and Smyth, 2005]



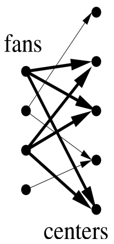


Scaling up

- How to find communities on a large graph, say, the Web?
- Web communities are characterized by dense directed bipartite graphs [Kumar et al., 1999]
- Idea similar to hubs and authorities
- Example: Pages of sport cars (Lotus, Ferrari, Lamborghini) and enthusiastic fans
- Bipartite cores: Complete bipartite cliques contained in a community
- Support from random graph theory: If *G = (U, V, E)* is a dense bipartite graph, then w.h.p. there is a $K_{i,j}$, for some *i* and *j*

An introduction to Web Mining, RuSSIR 2010, Voronezh

Detecting communities by trawling



- Many pruning phases

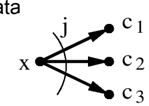
- Heuristic pruning (quality consideration)
  - Fans should point to at least 6 different hosts
  - Centers should be pointed by at most 50 fans

- Degree-based pruning
  - For a fan to participate in a $K_{i,j}$, it should have out-degree at least *j*
  - For a center to participate in a $K_{i,j}$, it should have in-degree at least *i*
  - Prune iteratively fans and centers
  - Can be done efficiently by sorting edges:
  - Sort edges by src to prune fans
  - Sort edges by dst to prune centers

An introduction to Web Mining, RuSSIR 2010, Voronezh

Detecting communities by trawling

- Inclusion-exclusion pruning
  - Either a core is output or a vertex is pruned
  - Computation is organized so that pruning is done with successive passes on the data



- A-priori pruning
  - Cores satisfy monotonicity
  - If *(X,Y)* is a $K_{i,j}$ then every *(X′,Y)* with $X' \subseteq X$ is a $K_{i',j}$
  - A-priori algorithm: start with *(1,j), (2,j), ...*
  - Most computationally demanding phase, but the graph is already heavily pruned

An introduction to Web Mining, RuSSIR 2010, Voronezh

Discussion

- Finding communities
- What is the right objective?
- Designing scalable algorithms is challenging
- How to evaluate the results?
- Studying dynamics and evolution of communities

(4) Detailed Examples

Topics

- **Statistical methods: the size of the web**
- **Content mining**
- **Social media mining**
- **Query mining**

What is the size of the web?

- **Issues**
  - The web is really infinite
    - Dynamic content, e.g., calendar
    - Soft 404: www.yahoo.com/anything is a valid page
  - Static web contains syntactic duplication, mostly due to mirroring (~20-30%)
  - Some servers are seldom connected
- **Who cares?**
  - Media, and consequently the user
  - Engine design
  - Engine crawl policy. Impact on recall

What can we attempt to measure?

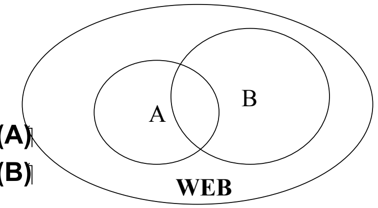
- The relative size of search engines
  - The notion of a page being indexed is *still* reasonably well defined.
  - Already there are problems
    - Document extension: e.g. Google indexes pages not yet crawled by indexing anchor-text.
    - Document restriction: Some engines restrict what is indexed (first *n* words, only relevant words, etc.)

- The coverage of a search engine relative to another particular crawling process

Relative size and overlap of search engines

- **[Bharat & Broder 98]**
- **Main idea:**
- **Pr[A&B | A] = s(A&B) / s(A)**
- **Pr[A&B | B] = s(A&B) / s(B)**
- **Thus:**

  s(A) / s(B) = Pr[A&B | B] / Pr[A&B | A]
- **Need**
  - **Sampling** a random page from the index of a SE
  - **Checking** if a page exists at the index of a SE

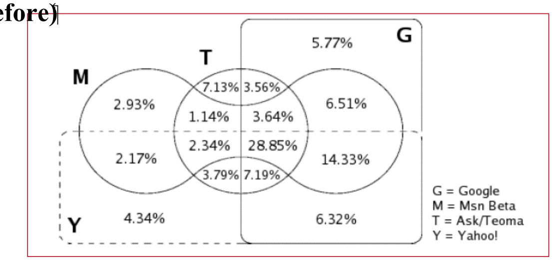


Sampling and checking pages

- **Both tasks by using the public interface SEs**
- **Sampling:**
  - Construct a large lexicon
  - Use the lexicon to fire random queries
  - Sample a page from the results
  - (introduces query and ranking biases)
- **Checking:**
  - Construct a *strong* query from the most k most distinctive terms of the page
  - (in order to deal with aliases, mirror pages, etc.)

Refinement of the B&B technique [Gulli & Signorini, 2005]

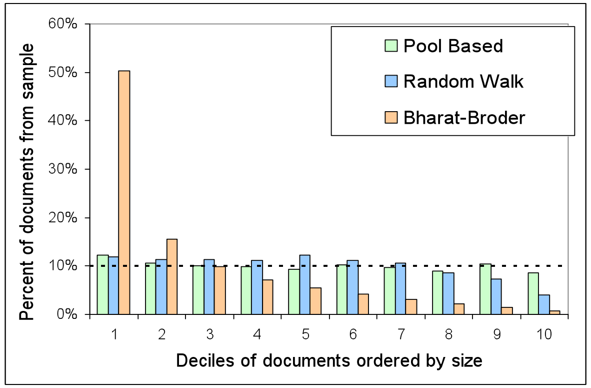
- **Total web = 11.5 B**
- **Union of major search engines = 9.5 B**
- **Common web = 2.7 B (Much higher correlation than before)**



Random-walk sampling

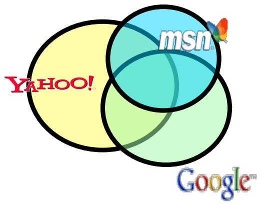
- [Bar-Yossef and Gurevich, WWW 2006]
- Define a graph on documents and queries:
  - Edge *(d,q)* indicates that document *d* is a result of a query *q*
- Random walk gives biased samples
- Bias depends on the degree of docs and queries
- Use Monte Carlo methods to unbias the samples and obtain uniform samples
- Paper shows how to obtain estimates of the degrees and weights needed for the unbiasing

Bias towards long documents



Relative size of major search engines

- [Bar-Yossef and Gurevich, 2006]



Google = 1

Yahoo! = 1.28

MSN Search = 0.73

Content mining

- **Duplicate and near-duplicate document detection**
- **Content-based Web spam detection**

Duplicate/Near-Duplicate Detection

- **Duplication: Exact match with fingerprints**
- **Near-Duplication: Approximate match**
  - Overview
    - Compute syntactic similarity with an edit-distance measure
    - Use similarity threshold to detect near-duplicates
      - E.g., Similarity > 80% => Documents are "near duplicates"
      - Not transitive though sometimes used transitively

Computing Similarity

- **Features:**
  - Segments of a document (natural or artificial breakpoints) [Brin95]
  - Shingles (Word N-Grams) [Brin95, Brod98]

    "a rose is a rose is a rose" =>

    a_rose_is_a

    rose_is_a_rose

    is_a_rose_is

  are all added in the bag of word representation

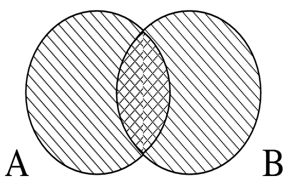
- **Similarity Measure**
  - TFIDF [Shiv95]
  - Set intersection [Brod98]

    (Specifically, Size_of_Intersection / Size_of_Union)

Jaccard coefficient

- **Consider documents *a* and *b***
- **Are represented by bag of words *A* and *B*, resp.**
- **Then:**

$$J(a,b) = |A \cap B| \,/\, |A \cup B|$$



Shingles + Jaccard coefficient

- **Computing exact Jaccard coefficient between all pairs of documents is expensive (quadratic)**
- **Approximate similarities using a cleverly chosen subset of shingles from each (a sketch)**
- **Idea based on hashing**

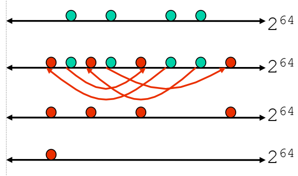
- **Also known as locality-sensitive hashing (LSH)**
  - A family of hash functions for which items that are similar have higher probability of colliding

Shingles + Jaccard coefficient

- **Estimate size_of_intersection / size_of_union based on a short sketch ([Broder 97, Broder 98])**
  - Create a “sketch vector” (e.g., of size 200) for each document
  - Documents which share more than t (say 80%) corresponding vector elements are similar
  - For doc D, sketch[i] is computed as follows:
    - Let f map all shingles in the universe to $0..2^m$ (e.g., f = fingerprinting)
    - Let π_i be a specific random permutation on $0..2^m$
    - Pick MIN π_i (f(s)) over all shingles s in D

Computing Sketch[i] for Doc1

Document 1



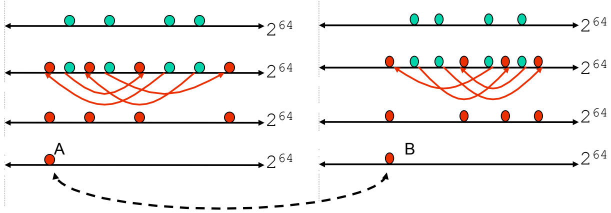
Start with 64 bit shingles

Permute on the number line with π_i

Pick the min value

Test if Doc1.Sketch[i] = Doc2.Sketch[i]

Document 1 | Document 2

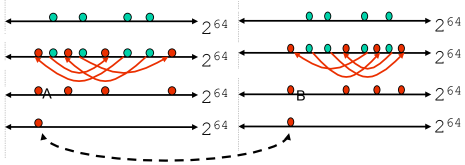


Are these equal?

Test for 200 random permutations: π_1, π_2,… π_{200}

However…

Document 1 | Document 2



A = B iff the shingle with the MIN value in the union of Doc1 and Doc2 is common to both (I.e., lies in the intersection)

This happens with probability:
`Size_of_intersection / Size_of_union`

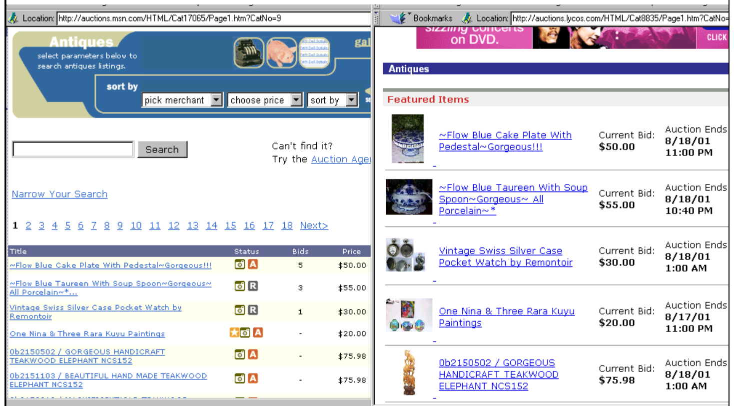
Mirror detection

- **Mirroring is systematic replication of web pages across hosts.**
  - Single largest cause of duplication on the web
- **Host1/α and Host2/β are mirrors iff**
  For all (or most) paths p such that when
  http://**Host1**/ α / p exists
  http://**Host2**/ β / p exists as well
  with identical (or near identical) content, and vice versa.
- **E.g.,**
  - http://**www.elsevier.com**/ and http://**www.elsevier.nl**/
  - Structural Classification of Proteins
    - http://**scop.mrc-lmb.cam.ac.uk**/scop
    - http://**scop.berkeley.edu**/
    - http://**scop.wehi.edu.au**/scop
    - http://**pdb.weizmann.ac.il**/scop
    - http://**scop.protres.ru**/

Repackaged Mirrors

Auctions.msn.com

Auctions.lycos.com



Aug 2001

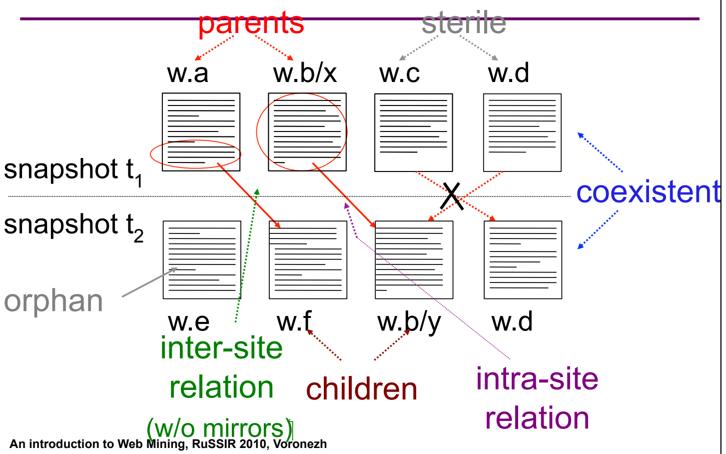
Motivation of near-duplicate detection

- **Why detect mirrors?**
  - Smart crawling
    - Fetch from the fastest or freshest server
    - Avoid duplication
  - Better connectivity analysis
    - Combine inlinks
    - Avoid double counting outlinks
  - Redundancy in result listings
    - "If that fails you can try: <mirror>/samepath"
  - Proxy caching

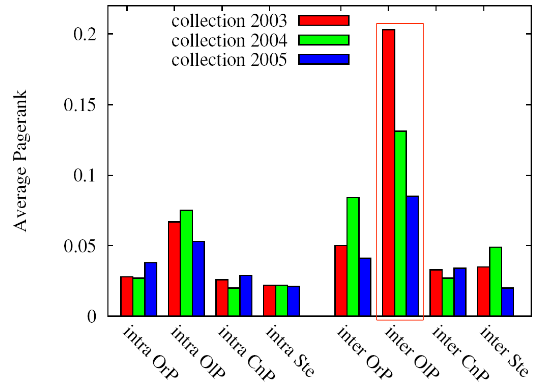
Study genealogy of the Web

- **[Baeza-Yates et al., 2008]**
- **New pages copy content from existing pages**
- **Web genealogy study:**
  - How textual content of source pages (parents) are reused to compose part of new Web pages (children)
  - Not near-duplicates, as similarities of short passages are also identified
- **How can search engines benefit?**
  - By associating more relevance to a parent page?
  - By trying to decrease the bias?

Web Genealogy



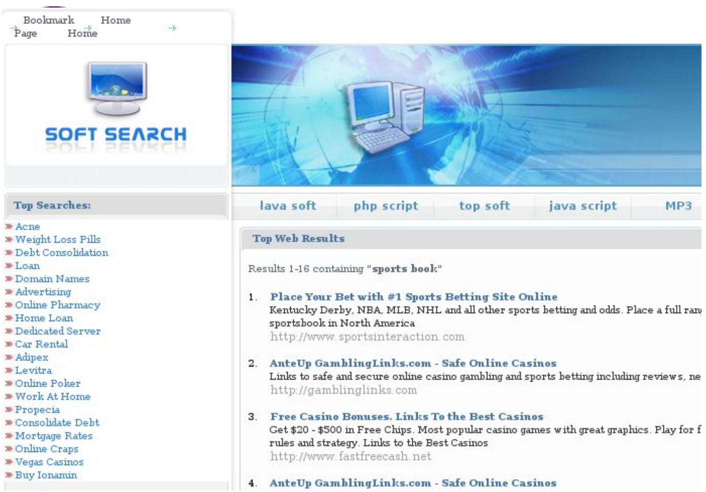
Pagerank for each component

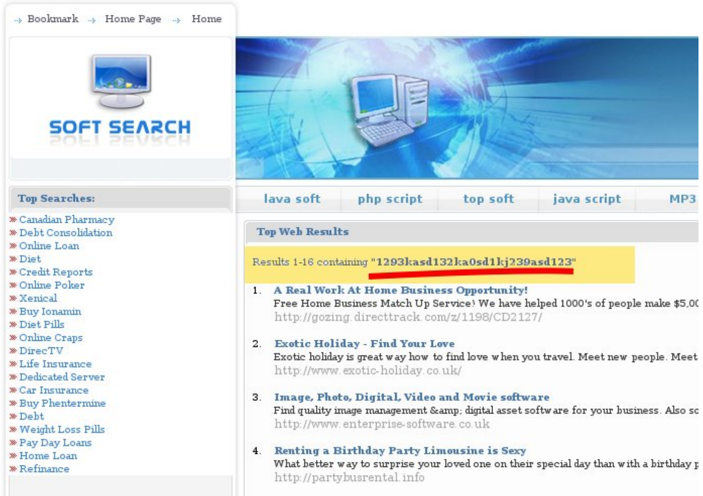


The wisdom of spammers

- Many world-class athletes, from all sports, have the ability to get in the right state of mind and when looking for **women looking for love** the state of mind is most important. [..] You should have the same attitude in looking for **women looking for love** and we make it easy for you.

- Many world-class athletes, from all sports, have the ability to get in the right state of mind and when looking for **texas boxer dog breeders** the state of mind is most important. [..] You should be thinking the same when you are looking for **texas boxer dog breeders** and we make it easy for you.





Sample query-targeted outlinks

- spam blocker
  free spam blocker
  outlook express spam blocker
  outlook spam blocker
  email spam blocker
  yahoo spam blocker
  free spam blocker outlook express
  spam blocker utility
  anti spam blocker
  microsoft spam blocker
  pop up spam blocker
  download free spam blocker
  free yahoo spam blocker
  bay area spam blocker
  blocking exchange server

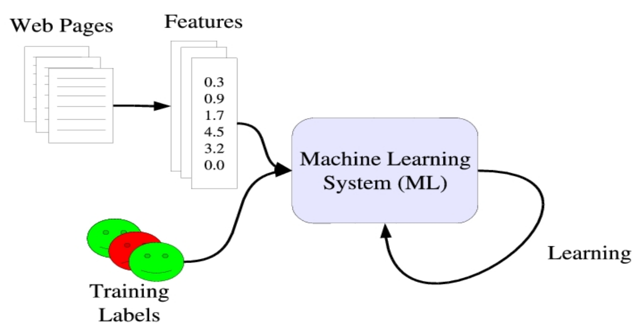
spam
spam e mail
mcafee anti spam
best anti spam
catch configuring email filter spam
blocker spam
send spam email
free junk spam filter outlook
adaptive filtering spam
anit software spam xp
blocker free spam
best spam block
free spam blocker and filter

The Power of Social Networks

- Spammers many times are (or look like) social networks
  - But the Web has larger social networks
- Examples
  - Any statistical deviation is suspicious
  - Any bounded amount of work is suspicious
    - Truncated PageRank
      - Spammers link support have shorter incoming paths

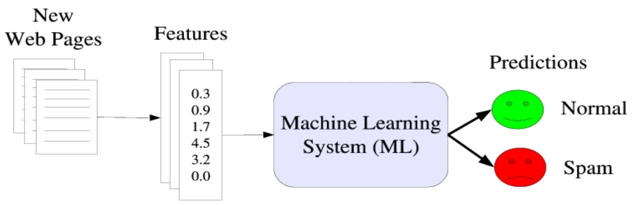
Content-based spam detection

- **Machine-learning approach --- training**



Content-based spam detection

- Machine-learning approach --- prediction



The dataset

- Label "spam" nodes on the host level
  - agrees with existing granularity of Web spam
- Based on a crawl of .uk domain from May 2006
- 77.9 million pages
- 3 billion links
- 11,400 hosts

The dataset

- 20+ volunteers tagged a subset of host
- Labels are "spam", "normal", "borderline"
- Hosts such as .gov.uk are considered "normal"
- In total 2,725 hosts were labelled by at least two judges
- hosts in which both judges agreed, and "borderline" removed
- Dataset available at

  http://www.yr-bcn.es/webspam/

Content-based features

- Number of words in the page
- Number of words in the title
- Average word length
- Fraction of anchor text
- Fraction of visible text

See also [Ntoulas et al., 06]

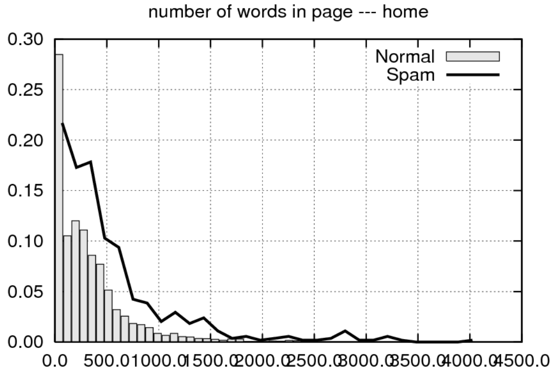
Content-based features
Entropy related

- Let $T = \{ (w_1, p_1), \ldots, (w_k, p_k) \}$ the set of trigrams in a page, where trigram w_i has frequency p_i
- Features:
  - ✓ Entropy of trigrams: $H = - \sum_i p_i \log(p_i)$
  - ✓ Independent trigram likelihood: $- (1/k) \sum_i \log(p_i)$
  - ✓ Also, compression rate, as measured by bzip

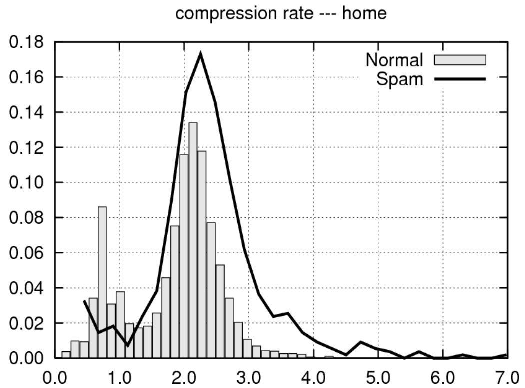
Content-based features
related to popular keywords

- F set of most frequent terms in the collection
- Q set of most frequent terms in a query log
- P set of terms in a page
- Features:
  - ✓ Corpus "precision" $| P \cap F | / | P |$
  - ✓ Corpus "recall" $| P \cap F | / | F |$
  - ✓ Query "precision" $| P \cap Q | / | P |$
  - ✓ Query "recall" $| P \cap Q | / | Q |$

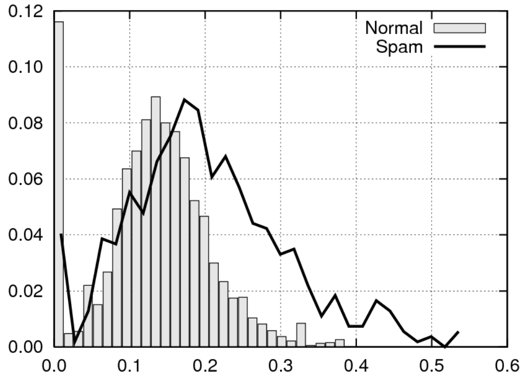
Content-based features
number of words in home page



Content-based features
compression rate



Content-based features Query precision



The classifier

- C4.5 decision tree with bagging and cost weighting for class imbalance
- With content-based features achieves:
  - True positive rate: 64.9%
  - False positive rate: 3.7%
  - F-Measure: 0.683

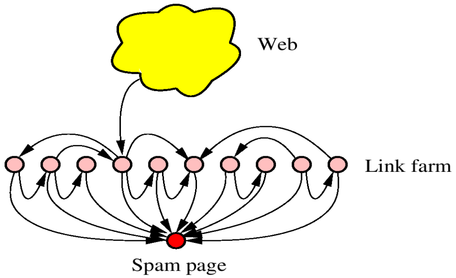
Structure and link analysis

- **Link-based spam detection**
- **Finding high-quality content in social media**

Link-based spam detection

- **Link farms used by spammers to raise popularity of spam pages**
- **Link farms and other spam strategies leave traces on the structure of the web graph**
- **Dependencies between neighbouring nodes of the web graph are created**
- **Naturally, spammers try to remove traces and dependencies**

Link farms



- **Single-level link farms can be detected by searching for nodes sharing their out-links**
- **In practice more sophisticated techniques are used**

Link-based features Degree related

- **in-degree**
- **out-degree**
- **edge reciprocity**
  - number of reciprocal links
- **assortativity**
  - degree over average degree of neighbors

Link-based features PageRank related

- **PageRank**
- **indegree/PageRank**
- **outdegree/PageRank**
- **...**
- **Truncated PageRank [Becchetti et al., 2006]**
  - A variant of PageRank that diminishes the influence of a page the PageRank score of its neighbors
- **TrustRank [Gyongyi et al., 2004]**
  - As PageRank but with teleportation at Open Directory pages

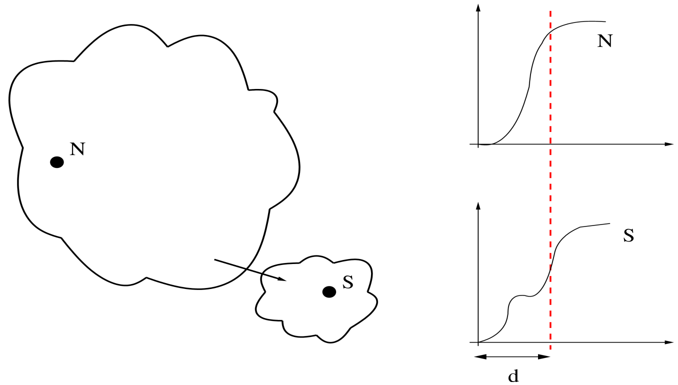
Link-based features Supporters

- **Let x and y be two nodes in the graph**
- **Say that y is a d-supporter of x, if the shortest path from y to x has length at most d**
- **Let $N_d(x)$ be the set of the d-supporters of x**
- **Define bottleneck number of x, up to distance d as**

$$b_d(x) = \min_{j <= d} N_j(x)/N_{j-1}(x)$$

- **minimum rate of growth of the neighbors of x up to a certain distance**

Link-based features
Supporters

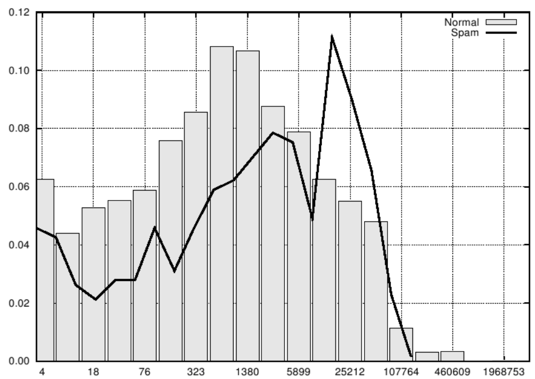


Link-based features
Supporters

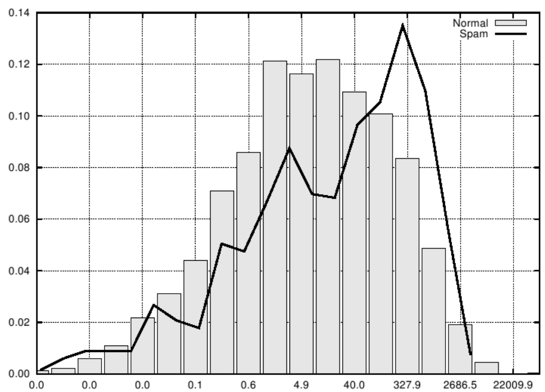
- **How to compute the supporters?**
- **Utilize *neighborhood function***

  $N(h) = |\{ (u,v) \mid d(u,v) \le h \}| = \sum_u N(u,h)$
- **and ANF algorithm [Palmer et al., 2002]**
- **Probabilistic counting using Flajolet-Martin sketches or other data-stream technology**
- **Can be done with a few passes and exchange of sketches, instead of executing BFS from each node**

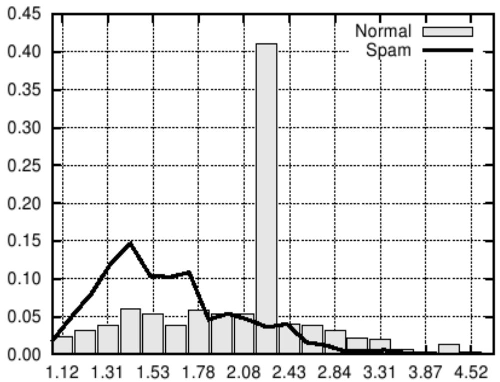
Link-based features - In-degree



Link-based features - Assortativity



Link-based features - Supporters

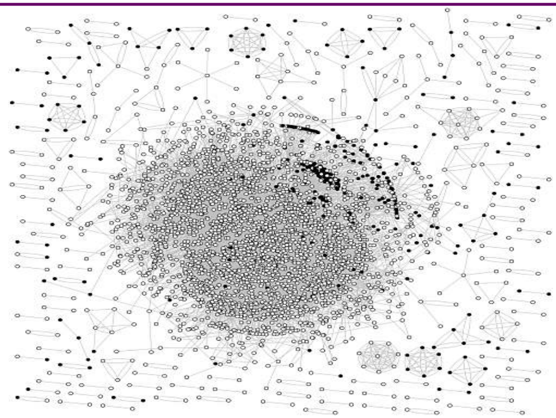


The classifier
Combining features

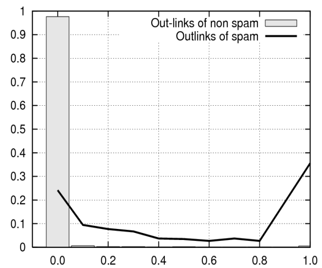
- C4.5 decision tree with bagging and cost weighting for class imbalance

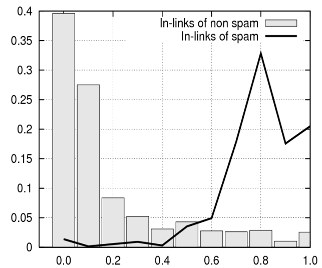
features:	Content	Link	Both
True positive rate:	64.9%	79.4%	78.7%
False positive rate:	3.7%	9.0%	5.7%
F-Measure:	0.683	0.659	**0.723**

Dependencies among spam nodes



Dependencies among spam nodes





- **Spam nodes in out-links**
- **Spam nodes from in-links**

Exploiting dependencies

- **Use a dataset with labeled nodes**
- **Extract content-based and link-based features**
- **Learn a classifier for predicting spam nodes independently**
- **Exploit the graph topology to improve classification**
  - Clustering
  - Propagation
  - Stacked learning

Exploiting dependencies Clustering

- **Let *G=(V,E,w)* be the host graph**
- **Cluster *G* into *m* disjoint clusters $C_1,\dots,C_m$**
- **Compute $p(C_i)$, the fraction of nodes classified as spam in cluster C_i**
  - if $p(C_i) > t_u$ label all as spam
  - if $p(C_i) < t_l$ label all as non-spam
- **A small improvement:**

	Baseline	**Clustering**
True positive rate:	78.7%	76.9%
False positive rate:	5.7%	5.0%
F-Measure:	0.723	**0.728**

Exploiting dependencies Propagation

- **Perform a random walk on thegraph**
- **With probability α follow a link**
- **With prob $1-\alpha$ jump to a random node labeled spam**
- **Relabel as spam every node whose stationary distribution component is higher than a threshold**

- **Improvement:**

	Baseline	**Propagation (backwds)**
True positive rate:	78.7%	75.0%
False positive rate:	5.7%	4.3%
F-Measure:	0.723	**0.733**

Exploiting dependencies Stacked learning

- **Meta-learning scheme [Cohen and Kou, 2006]**
- **Derive initial predictions**
- **Generate an additional attribute for each object by combining predictions on neighbors in the graph**
- **Append additional attribute in the data and retrain**

- **Let *p(h)* be the prediction of a classification algorithm for *h***
- **Let *N(h)* be the set of pages related to *h***
- **Compute:**

$$f(h) = \sum_{g \in N(h)} p(g) / |N(h)|$$

- **Add *f(h)* as an extra feature for instance *h* and retrain**

Exploiting dependencies Stacked learning

- **First pass:**

	Baseline	**in**	**out**	**both**
True positive rate:	78.7%	84.4%	78.3%	85.2%
False positive rate:	5.7%	6.7%	4.8%	6.1%
F-Measure:	0.723	0.733	0.742	**0.750**

- **Second pass:**

	Baseline	**1st pass**	**2nd pass**
True positive rate:	78.7%	85.2%	88.2%
False positive rate:	5.7%	6.1%	6.3%
F-Measure:	0.723	0.750	**0.763**

Current goals for Web spam effort

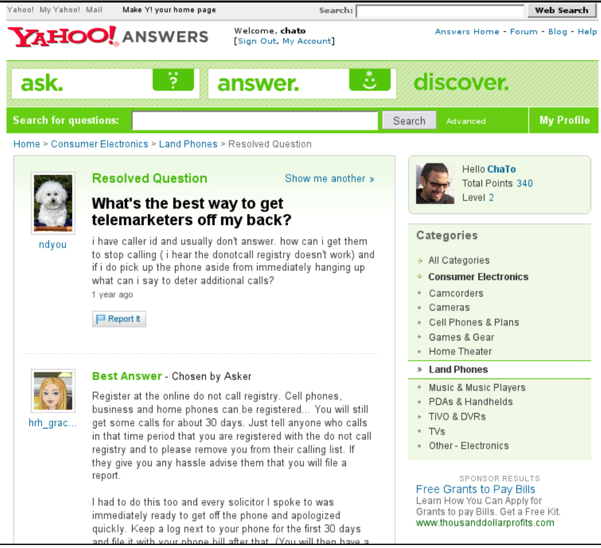
- Prevent spam from distorting ranking, but preserve:
  - Relevance
    - “Perfect spam” is a sensible category
  - Freshness
    - Can’t slow down discovery just because spammers exploit it
  - Comprehensiveness
    - Navigational queries for spam should succeed
- Focus on two kinds of spam only:
  - 1) Spam that is succeeding in ranking inappropriately highly
  - 2) Spam that consumes huge amounts of system resources

    (Everything else is “dark matter”)

The power of social media

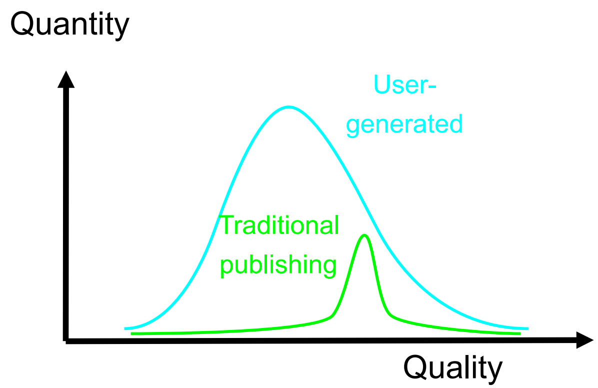
- Flickr – community phenomenon
- Millions of users share and tag each others’ photographs (why???)
- The *wisdom of the crowds* can be used to search
  - Ranking features to Yahoo! Answers
- The principle is not new – anchor text used in “standard” search
- What about generating pseudo-semantic resources?

Yahoo! Answers

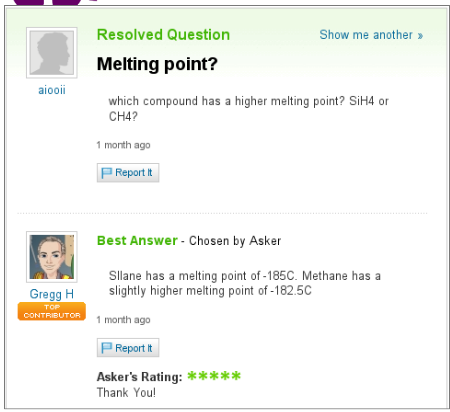


Finding high-quality content in social media

- A lot of social-media sites in which users publish their own content
- Various types of activities and information: links, social ties, comments, feedback, views, votes, stars, user status, etc.
- Quality of published items can vary greatly
- Highly relevant information might be present
- But, how do we find it?



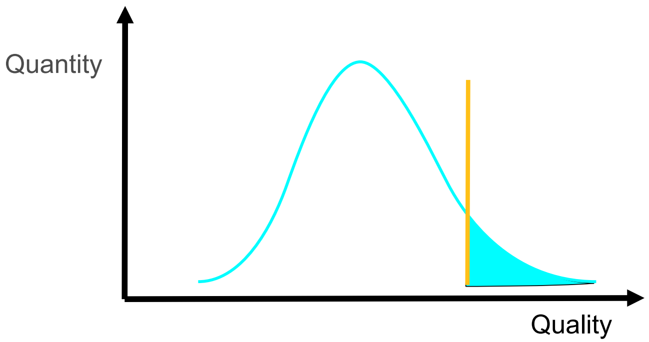
Q. Su, D. Pavlov, J.-H. Chow, W. C. Baker. "Internet-scale collection of human-reviewed data". WWW'07.



17%-45% of answers were correct

65%-90% of questions had at least one correct answer

Task: find high-quality items



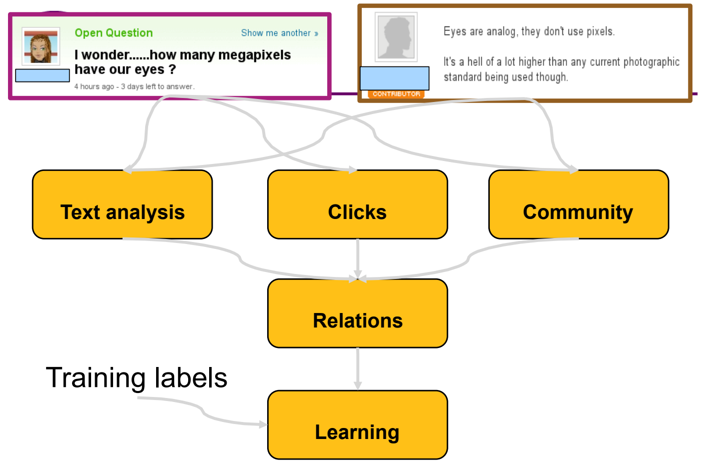
Existing techniques

- **Information retrieval methods**
- **Automatic text analysis**
- **Link-based ranking methods**
- **Propagation of trust/distrust**
- **Usage mining**

Sources of information

- **Content**
- **Usage data (clicks)**
- **Community ratings**

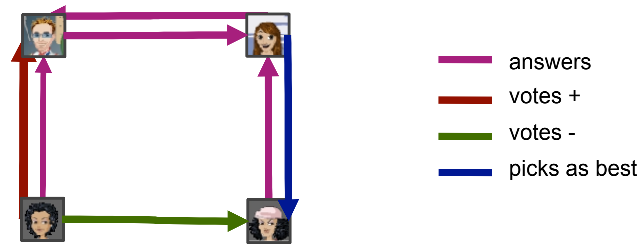
- **...but sparse, noisy, and with spam...**



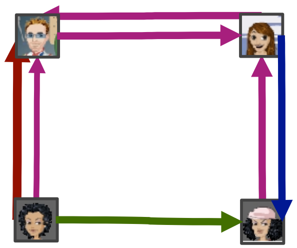
Combining the existing information

- **Text features**
  - Distribution of n-grams
- **Linguistic features**
  - Punctuation, syntactic, case, part-of-speech tags
- **Social features**
  - Consider user-interaction graphs:
    - G1: user A answers a question of user B
    - G2: user A votes for an answer of user B
  - Apply HITS and PageRank
- **Usage features**
  - Number of clicks
  - Deviation of number of clicks from mean of category

Community



Community



Propagation-based metrics

1. Pagerank score
2. HITS hub score
3. HITS authority score

Computed on each graph

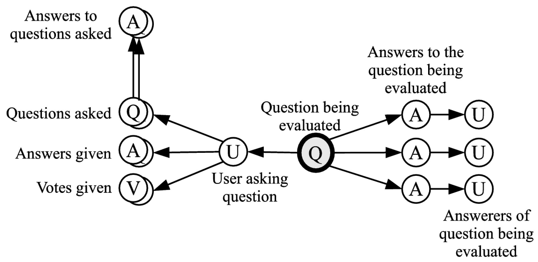
Relations

Question quality

Answer quality		High	Medium	Low
	High	41%	15%	8%
	Medium	53%	76%	74%
	Low	6%	9%	18%
		100%	100%	100%

Question quality and answer quality are not independent

Propagation of features



Task: high-quality questions

	Precision	Recall	AUC
N-grams (N)	65%	48%	0.52
N+text analysis	76%	65%	0.65
N+clicks	68%	57%	0.58
N+relations	74%	65%	0.66
All	79%	77%	0.76

Challenges in social media

- What's the ratings and reputation system?
- How do you cope with spam?
  - The wisdom of the crowd can be used against spammers
- The bigger challenge: where else can you exploit the power of the people?
- What are the incentive mechanisms?
  - Example: ESP game

Discussion

- **Relevant content is available in social media, but the variance of the quality is very high**
- **Classifying questions/answers is different than document classification**
- **Combine many orthogonal features and heterogeneous information**

Overall summary

- **Open problems and challenges:**
  - Manage and integrate highly heterogeneous information:
  - Content, links, social links, tags, feedback, usage logs, wisdom of crowd, etc.
  - Model and benefit from evolution
  - Battle adversarial attempts and collusions

Web Search Queries

- **Cultural and educational diversity**
- **Short queries & impatient interaction**
  - few queries posed & few answers seen
- **Smaller & different vocabulary**
- **Different user goals [Broder, 2000]:**
  - Information need
  - Navigational need
  - Transactional need
- **Refined by Rose & Levinson, WWW 2004**

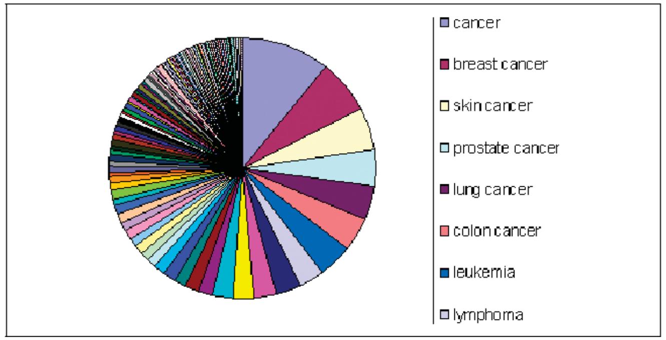
User Needs

- **Need (Broder 2002)**
  - **Informational** – want to learn about something (~40% / 65%)

    `Low hemoglobin`
  - **Navigational** – want to go to that page (~25% / 15%)

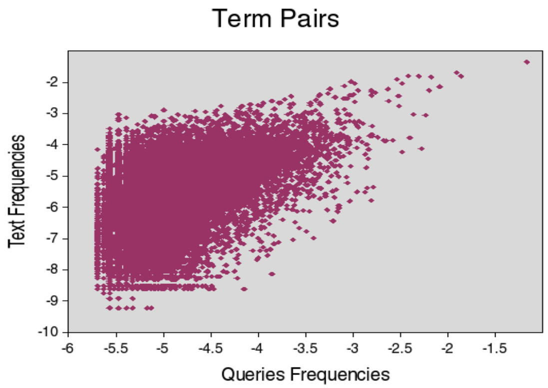
    `United Airlines`
  - **Transactional** – want to do something (web-mediated) (~35% / 20%)
    - Access a service — `Edinburgh weather`
    - Downloads — `Mars surface images`
    - Shop — `Canon S410`
  - Gray areas
    - Find a good hub — `Car rental Brasil`
    - Exploratory search "see what's there"

Query Distribution



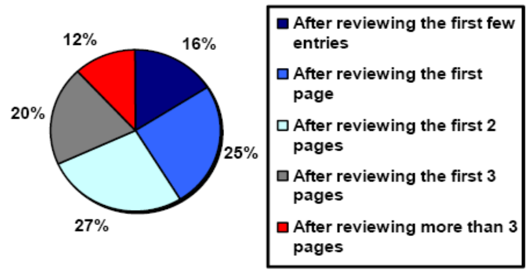
Power law: few popular broad queries, many rare specific queries

Queries and Text



How far do people look for results?

"When you perform a search on a search engine and don't find what you are looking for, at what point do you typically either revise your search, or move on to another search engine? (Select one)"



(Source: iprospect.com WhitePaper_2006_SearchEngineUserBehavior.pdf)

Typical Session

- Two queries of
- .. two words, looking at…
- .. two answer pages, doing
- .. two clicks per page

- What is the goal?

MP3
games
cars
britney spears
pictures
ski

Relevance of the Context

- **There is no information without context**
- **Context and hence, content, will be implicit**
- **Balancing act: information vs. form**
- **Brown & Diguid: *The social life of information* (2000)**
  - Current trend: less information, more context
- **News highlights are similar to Web queries**
  - E.g.: *Spell Unchecked (Indian Express, July 24, 2005)*

Context

- ***Who you are***: **age, gender, profession, etc.**
- ***Where you are and when***: **time, location, speed and direction, etc.**
- ***What you are doing***: **interaction history, task in hand, searching device, etc.**
- ***Issues***: **privacy, intrusion, will to do it, etc.**
- ***Other sources***: **Web, CV, usage logs, computing environment, ...**
- ***Goals***: **personalization, localization, better ranking in general, etc.**

Context in Web Queries

- ***Session: (q, (URL, t)*)***$^{+}$

- ***Who you are***: **age, gender, profession (IP), etc.**
- ***Where you are and when***: **time, location (IP), speed and direction, etc.**
- ***What you are doing***: **interaction history, task in hand, etc.**
- ***What you are using***: **searching device (operating system, browser, ...)**

SEARCH GOAL	DESCRIPTION	EXAMPLES
1. Navigational	My goal is to go to specific known website that I already have in mind. The only reason I'm searching is that it's more convenient than typing the URL, or perhaps I don't know the URL.	aloha airlines duke university hospital kelly blue book
2. Informational	My goal is to learn something by reading or viewing web pages	**Home page**
2.1 Directed	I want to learn something in particular about my topic	
2.1.1 Closed	I want to get an answer to a question that has a single, unambiguous answer.	what is a supercharger 2004 election dates
2.1.2 Open	I want to get an answer to an open-ended question, or one with unconstrained depth.	baseball death and injury why are metals shiny
2.2 Undirected	I want to learn anything/everything about my topic. A query for topic X might be interpreted as "tell me about X."	color blindness jfk jr
2.3 Advice	I want to get advice, ideas, suggestions, or instructions.	help quitting smoking walking with weights
2.4 Locate	My goal is to find out whether/where some real world service or product can be obtained	pella windows phone card
2.5 List	My goal is to get a list of plausible suggested web sites (I.e. the search result list itself), each of which might be candidates for helping me achieve some underlying, unspecified goal	travel amsterdam universities florida newspapers
3. Resource	My goal is to obtain a resource (not information) available on web pages	**Hub page**
3.1 Download	My goal is to download a resource that must be on my computer or other device to be useful	kazaa lite mame roms
3.2 Entertainment	My goal is to be entertained simply by viewing items available on the result page	xxx porn movie free live camera in l.a.
3.3 Interact	My goal is to interact with a resource using another program/service available on the web site I find	weather measure converter
3.4 Obtain	My goal is to obtain a resource that does not require a computer to use. I may print it out, but I can also just look at it on the screen. I'm not obtaining it to learn some information, but because I want to use the resource itself.	free jack o lantern patterns ellis island lesson plans house document no. 587

Page with resources

Rose & Levinson 2004

Kang & Kim, SIGIR 2003

- **Features:**
  - Anchor usage rate
  - Query term distribution in home pages
  - Term dependence
- **Not effective: 60%**

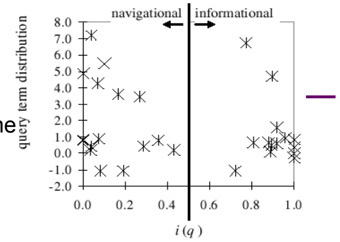


Figure 16: Query term distribution

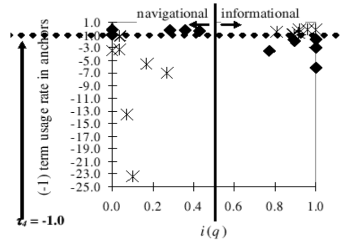


Figure 15: Anchor usage rate

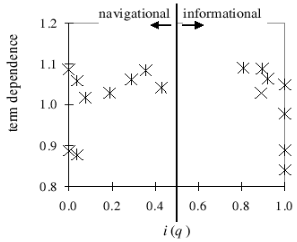


Figure 17: Term dependence

User Goals

- Liu, Lee & Cho, WWW 2005
- Top 50 CS queries
- Manual Query Classification: 28 people
- Informational goal $i(q)$
- *Remove software & person-names*
- *30 queries left*

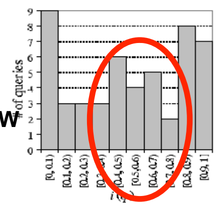


Figure 1: Query distribution along the $i(q)$ axis

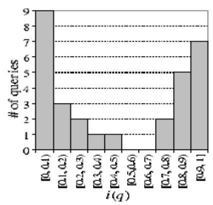


Figure 2: After removing software and person-name queries

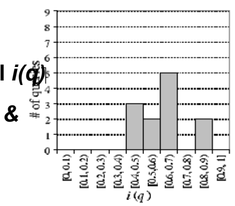


Figure 3: Distribution of the 12 software queries

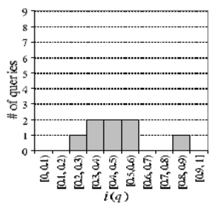
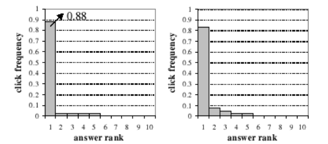


Figure 4: Distribution of the 8 person-name queries

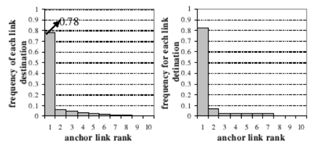
Features

- Click & anchor text distribution



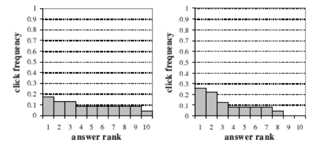
(a) pubmed ($i(q)$=0.1) (b) ucla library ($i(q)$=0)

Figure 5: Click distributions for sample navigational queries



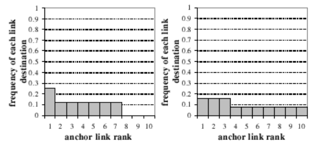
(a) pubmed ($i(q)$=0.1) (b) ucla library ($i(q)$=0)

Figure 7: Anchor-link distributions for sample navigational queries



(a) hidden markov model ($i(q)$=1) (b) simulated annealing ($i(q)$=1)

Figure 6: Click distributions for sample informational queries



(a) hidden markov model ($i(q)$=1) (b) simulated annealing ($i(q)$=1)

Figure 8: Anchor-link distributions for sample informational queries

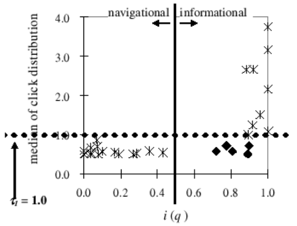


Figure 11: Median of click distribution

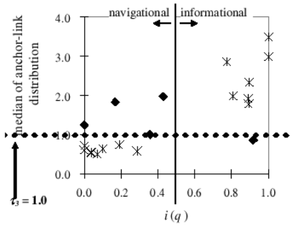


Figure 13: Median of anchor-link distribution

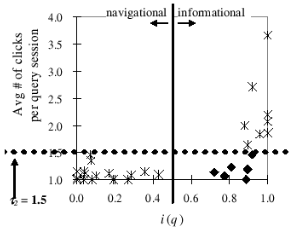


Figure 12: Avg # of clicks per query

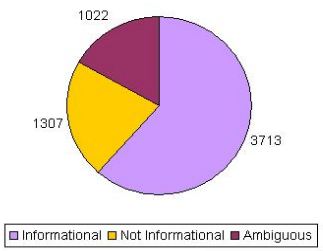
- Prediction power:
- Single features: 80%
- Mixed features: 90%

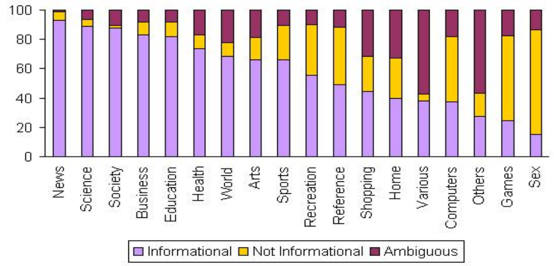
- Drawbacks:
  - Small evaluation
  - a posteriori feature

User Intention

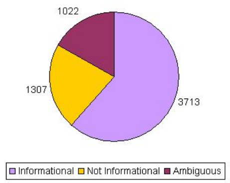
- **Manual classification of more than 6,000 popular queries**
- **Query Intention & topic**
- **Classification & Clustering**
- **Machine Learning on all the available attributes**
- **Baeza-Yates, Calderon & Gonzalez (SPIRE 2006)**

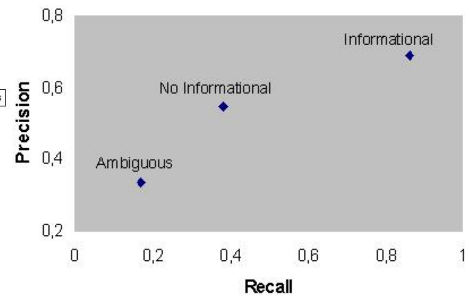
Classified Queries





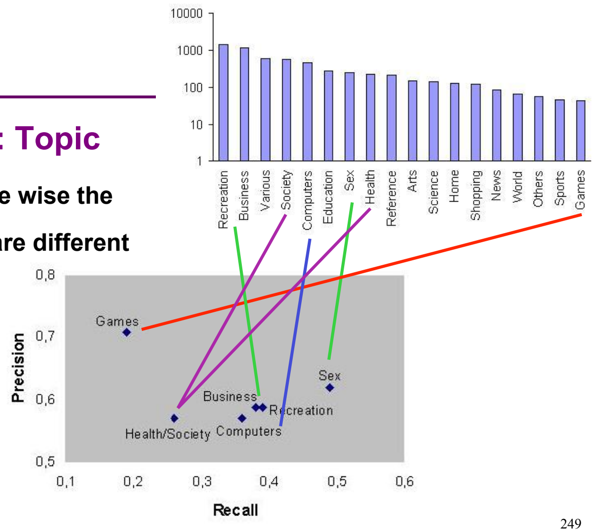
Results: User Intention

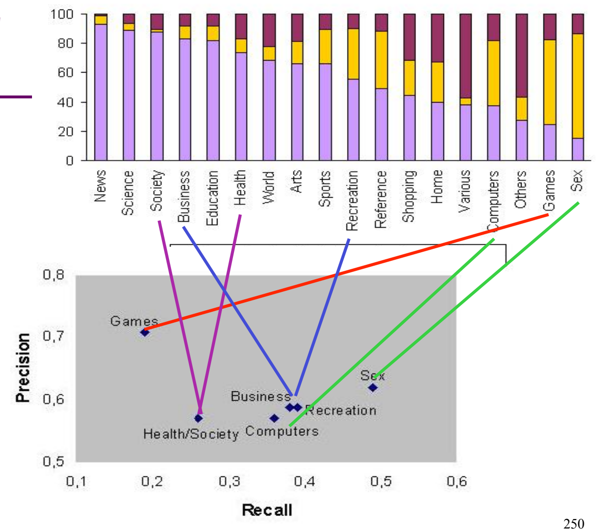




Results: Topic

- **Volume wise the results are different**



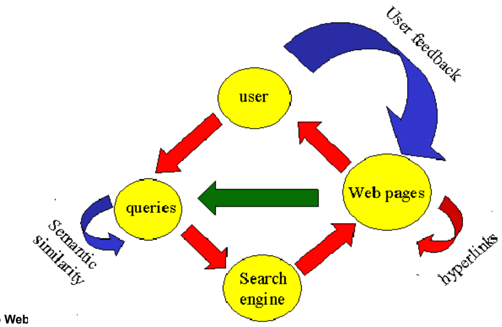


Clustering Queries

- **Define relations among queries**
  - Common words: sparse set
  - Common clicked URLs: better
  - Natural clusters
- **Define distance function among queries**
  - Content of clicked URLs [Baeza-Yates, Hurtado & Mendoza, 2004]
  - Summary of query answers [Sahami, 2006]

Goals

- **Can we cluster queries well?**
- **Can we assign user goals to clusters?**



Our Approach

- **Cluster text of clicked pages**
  - Infer query clusters using a vector model

$$q[i] = \sum_{URL u} \frac{\mathtt{Pop}(q,u) \times \mathtt{Tf}(t_i,u)}{\max_t \mathtt{Tf}(t,u)}$$

- **Pseudo-taxonomies for queries**
  - Real language (slang?) of the Web
  - Can be used for classification purposes

Clusters Examples

Q	Cluster Rank	ISim	ESim	Queries in Cluster	Descriptive keywords
q_1	252	0,447	0,007	car sales, cars Iquique, cars used, diesel, new cars,	cars (49, 4%), used (14, 2%), stock (3, 8%), pickup truck (3, 7%), jeep (1, 6%)
q_2	497	0,313	0,009	stamp, serigraph inputs, ink reload, cartridge	print (11, 4%), ink (7, 3%), stamping (3, 8%), inkjet (3, 6%)
q_3	84	0,697	0,015	office rental, rentals in Santiago, real state, apartment rental	office (11, 6%), building (7, 5%), real state (5, 9%), real state agents (4, 2%)

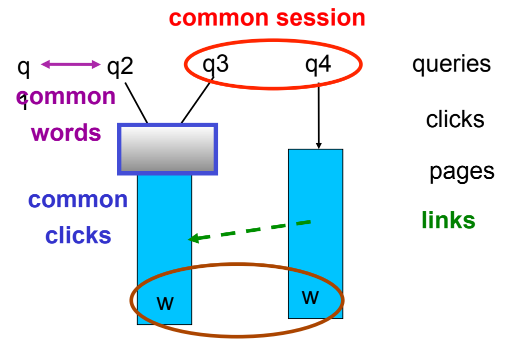
Using the Clusters

- **Improved ranking** Baeza-Yates, Hurtado & Mendoza Journal of ASIST 2007
- **Word classification**
  - Synonyms & related terms are in the same cluster
  - Homonyms (polysemy) are in different clusters
- **Query recommendation (ranking queries!)**
  - F $\mathtt{Rank}(q) = \gamma \times \mathtt{Sup}(q, q_{ini}) + (1-\gamma) \times \mathtt{Clos}(q)$

Query Recommendation

Query	Popularity	Support	Closedness	Rank
rentals apartments viña del mar owners	2	0,133	0,403	0,268
rentals apartments viña del mar	10	0,2	0,259	0,229
viel properties	4	0,1	0,315	0,207
rental house viña del mar	2	0,166	0,121	0,143
house leasing rancagua	8	0,166	0,0385	0,102
quintero	2	0,166	0,024	0,095
rentals apartments cheap vina del mar	3	0,033	0,153	0,093
subsidize renovation urban	5	0,133	0,001	0,067
houses being sold in pucon	10	0	0,114	0,057
apartments selling pucon villarrica	2	0,066	0,015	0,040
portal sell properties	3	0,033	0,023	0,028
sell house	2	0,033	0,017	0,025
sell lots pirque	2	0,033	0,0014	0,017
canete hotels	1	0	0,011	0,005

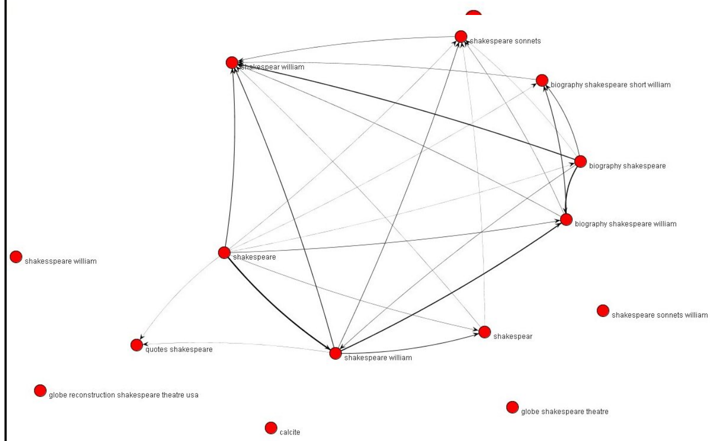
Relating Queries (Baeza-Yates, 2007)



Qualitative Analysis

Graph	Strength	Sparsity	Noise
Word	Medium	High	Polysemy
Session	Medium	High	Physical sessions
Click	High	Medium	**Multitopic pages Click spam**
Link	Weak	Medium	Link spam
Term	Medium	Low	Term spam

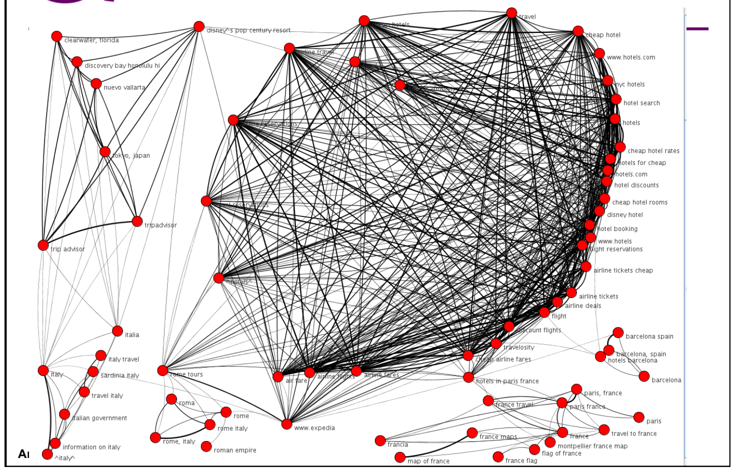
Words, Sessions and Clicks



Contributions

- **Characterization of a large click graph**
- **Proposed specific distance and relations**
- **Hint the amount of implicit knowledge**
- **Evaluate the quality of the results**

Click Graph



Formal Definition

- **There is an edge between two queries *q* and *q'* if:**
  - There is at least one URL clicked by both
- **Edges can be weighted (for filtering)**
  - We used the cosine similarity in a vector space defined by URL clicks

$$W(e) = \frac{\bar{q} \cdot \bar{q}'}{|\bar{q}|\ |\bar{q}'|} = \frac{\sum_{i \le D} q(i) \cdot q'(i)}{\sqrt{\sum_{i \le D} q(i)^2} \cdot \sqrt{\sum_{i \le D} q'(i)^2}}$$

URL based Vector Space

- **Consider the query *"complex networks"***
- **Suppose for that query the clicks are:**
  - *www.ams.org/featurecolumn/archive/networks1.html* (3 clicks)
  - en.wikipedia.org/wiki/Complex_network (1 click)

0	0	0	0	...	1/4	...	3/4	...	0	0	0	0

"Complex networks"

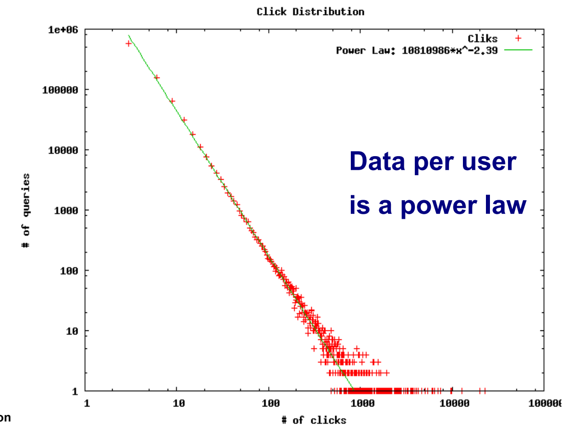
Building the Graph

- **The graph can be built efficiently:**
  - Consider the tuples (query, clicked url)
  - Sort by the second component
  - Each block with the same URL *u* gives the edges induced by *u*
  - Complexity: *O(max {M*|E|, n log n})* where *M* is the maximum number of URLs between two queries, and *n* is the number of nodes

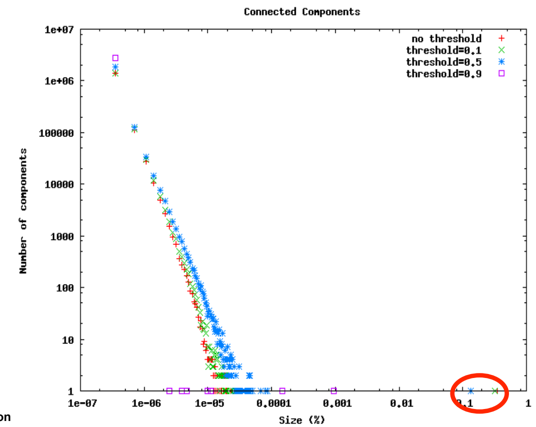
Anatomy of a Click Graph

- **We built graphs using logs with up to 50 millions queries**
  - For all the graphs we studied our findings are qualitatively the same (*scale-free network?*)
- **Here we present the results for the following graph**
  - 20M query occurrences
  - 2.8M distinct queries (nodes)
  - 5M distinct URLs
  - 361M edges

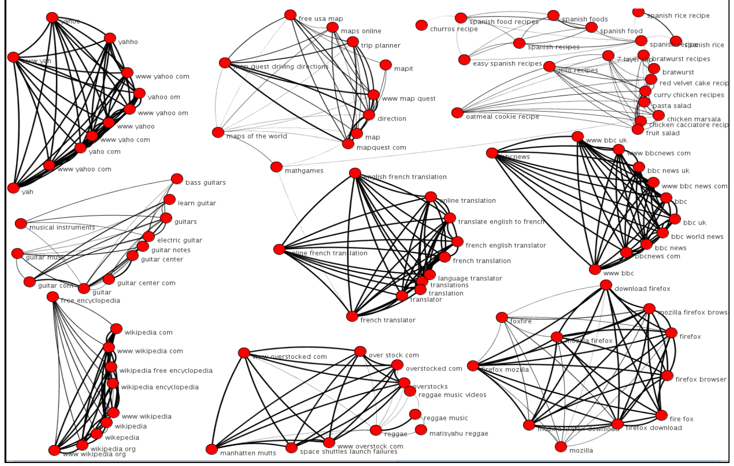
Click Distribution



Connected Components



Implicit Folksonomy?

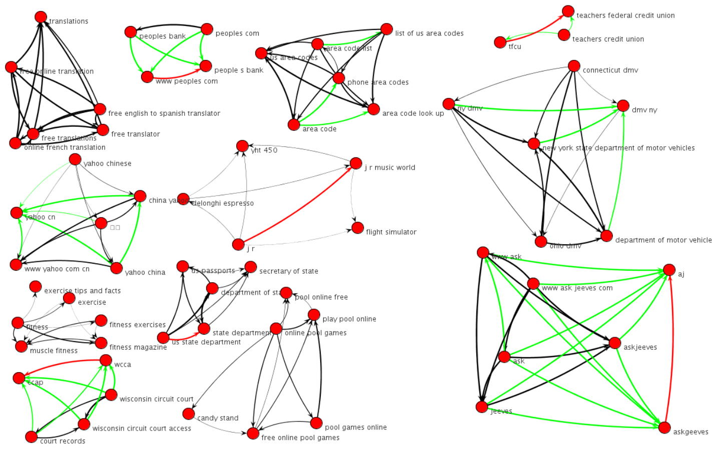


Set Relations and Graph Mining

- **Identical sets: equivalence**
- **Subsets: specificity**
  - directed edges
- **Non empty intersections (with threshold)**
  - degree of relation
- **Dual graph: URLs related by queries**
  - High degree: multi-topical URLs

Baeza-Yates & Tiberi ACM KDD 2007

Implicit Knowledge? Webslang!



Evaluation: ODP Similarity

- **A simple measure of similarity among queries using ODP categories**
  - Define the similarity between two categories as the length of the longest shared path over the length of the longest path
  - Let *c_1,.., c_k* and *c'_1,.., c'_k* be the top *k* categories for two queries. Define the similarity (@*k*) between the two queries as *max{sim(c_i,c'_j) | i,j=1,..,K }*

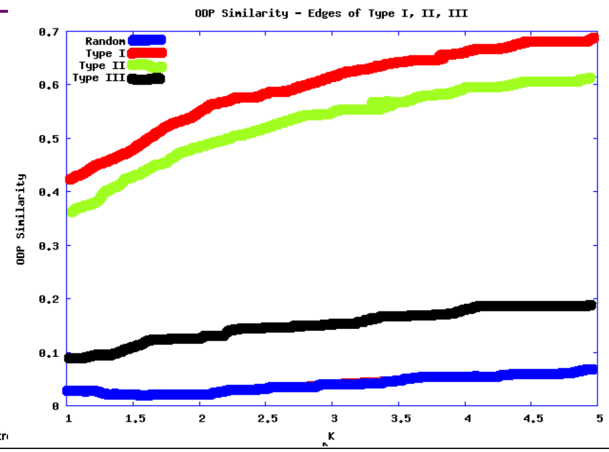
ODP Similarity

- **Suppose you submit the queries "*Spain*" and "*Barcelona*" to ODP.**
- **The first category matches you get are:**
  - Regional/ Europe/ Spain
  - Regional/ Europe/ Spain/ Autonomous Communities/ Catalonia/ Barcelona
- **Similarity @1 is 1/2 because the longest shared path is "Regional/ Europe/ Spain" and the length of the longest is 6**

Experimental Evaluation

- We evaluated a 1000 thousand edges sample for each kind of relation
- We also evaluated a sample of random pairs of not adjacent queries (baseline)
- We studied the similarity as a function of *k* (the number of categories used)

Experimental Evaluation



Open Issues

- **Implicit social network**
  - Any fundamental similarities?
- **How to evaluate with partial knowledge?**
  - Data volume amplifies the problem
- **User aggregation vs. personalization**
  - Optimize common tasks
  - Move away from privacy issues

(5) Final Remarks

Epilogue

- **The Web is scientifically young**
- **The Web is intellectually diverse**
- **The technology mirrors the economic, legal and sociological reality**
- **Web Mining: large potential for many applications**
  - A fast prototyping platform is needed
- **Plenty of open problems**

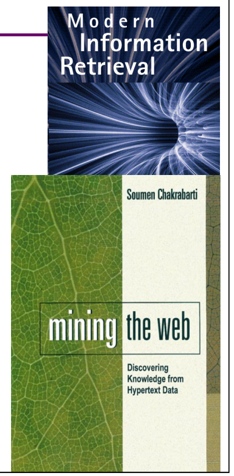
Overall summary

- **Many open problems and challenges:**
  - Manage and integrate highly heterogeneous information:
  - Content, links, social links, tags, feedback, usage logs, wisdom of crowds, etc.
  - Model and benefit from evolution
  - Battle adversarial attempts and collusions

Special thanks

- **Andrei Broder**
- **Carlos Castillo**
- **Barbara Poblete**
- **Alvaro Pereira**
- **Prabhakar Raghavan**
- **Alessandro Tiberi**

Bibliography – General



- **Modern Information Retrieval** by R. Baeza-Yates & B. Ribeiro-Neto, Addison-Wesley, 1999. Second edition in preparation.
- **Managing Gigabytes: Compressing and Indexing Documents and Images** by I.H. Witten, A. Moffat, and T.C. Bell. Morgan Kaufmann, San Francisco, second edition, 1999.
- **Mining the Web: Analysis of Hypertext and Semi Structured Data** by Soumen Chakrabarti. Morgan Kaufmann; August 15, 2002.
- **The Anatomy of a Large-scale Hypertextual Web Search Engine** by S. Brin and L. Page. 7th International WWW Conference, Brisbane, Australia; April 1998.
- **Websites:**
  - http://www.searchenginewatch.com/
  - http://www.searchengineshowdown.com/