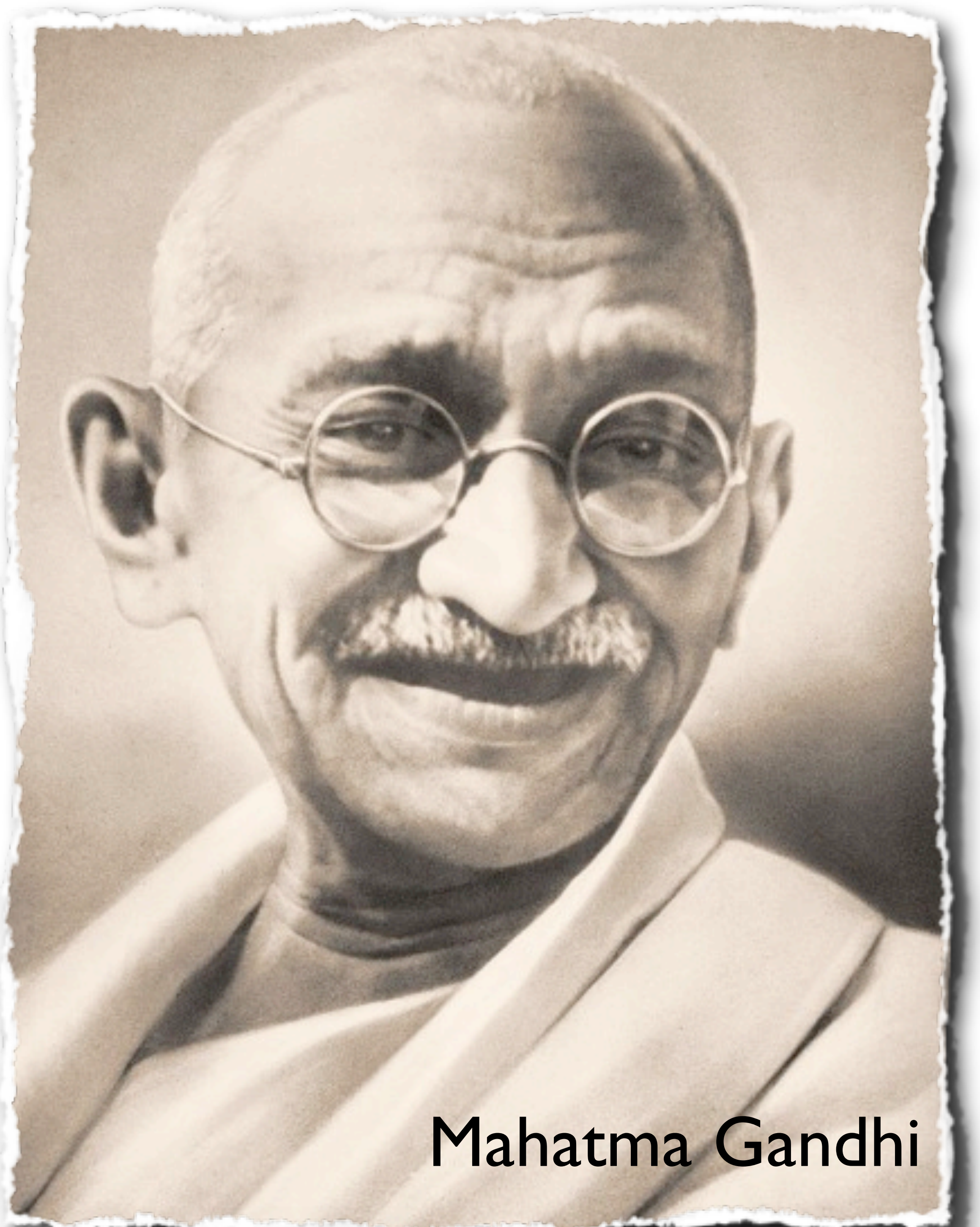
Computational Approaches in Social Computing

Irwin King

Department of Computer Science and Engineering
The Chinese University of Hong Kong

king@cse.cuhk.edu.hk
http://www.cse.cuhk.edu.hk/~king



Interdependence is and ought to be as much the ideal of man as self-sufficiency.

Man is a social being.


intel.
revolution in evolution
Highlights from the Journey to 1 Billion PCs
For more information, please visit http://www.intel.com

Social Networking

Billionaires’ Shuffle

2007

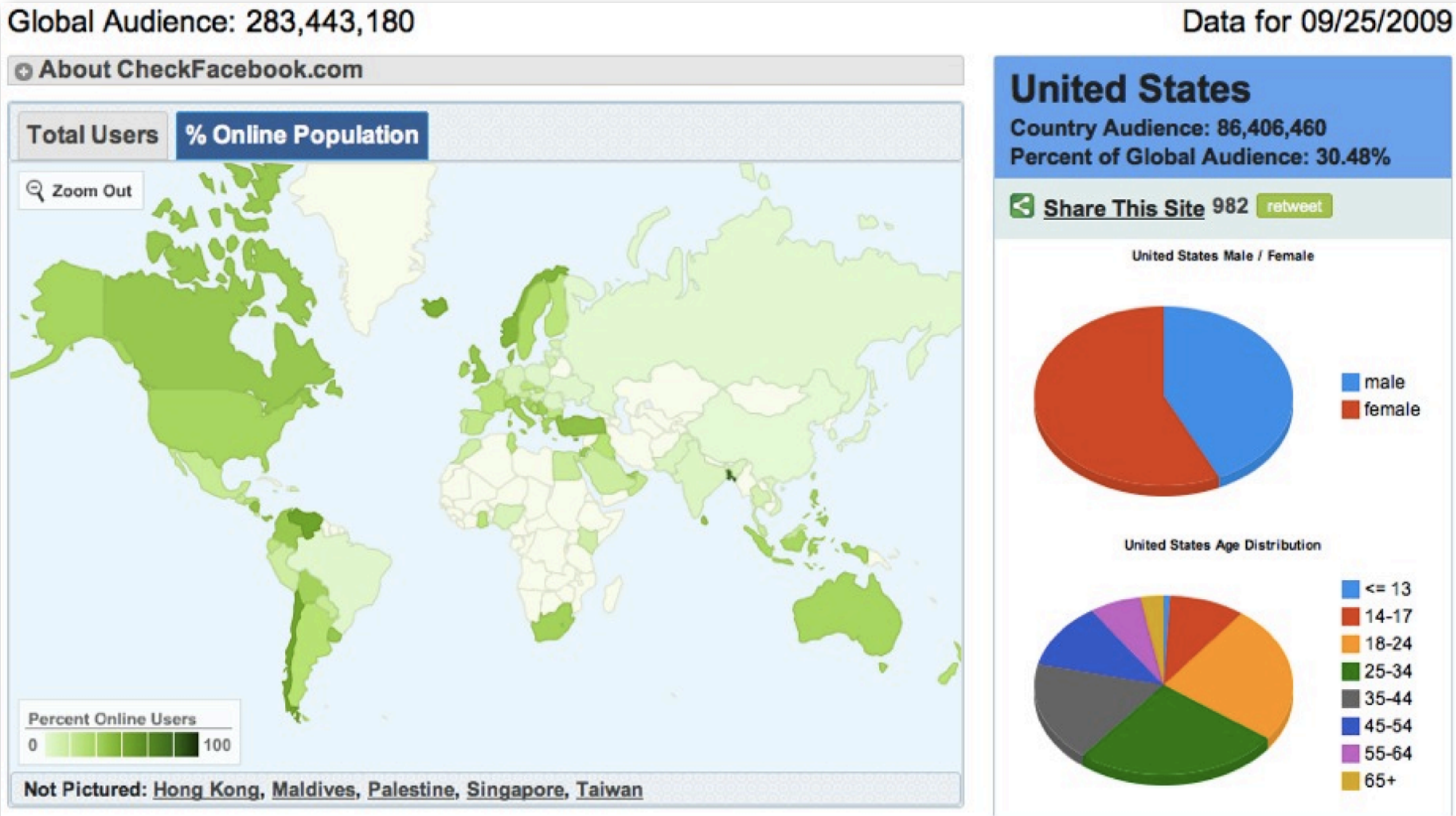
Facebook in 2004.02

2008

at **23** and $**1.5** billion later...

2008

Facebook's Global Audience



Facebook's Growth Table

General Growth More than 300 million active users
50% of our active users log on to Facebook in any given day
The fastest growing demographic is those 35 years old and older

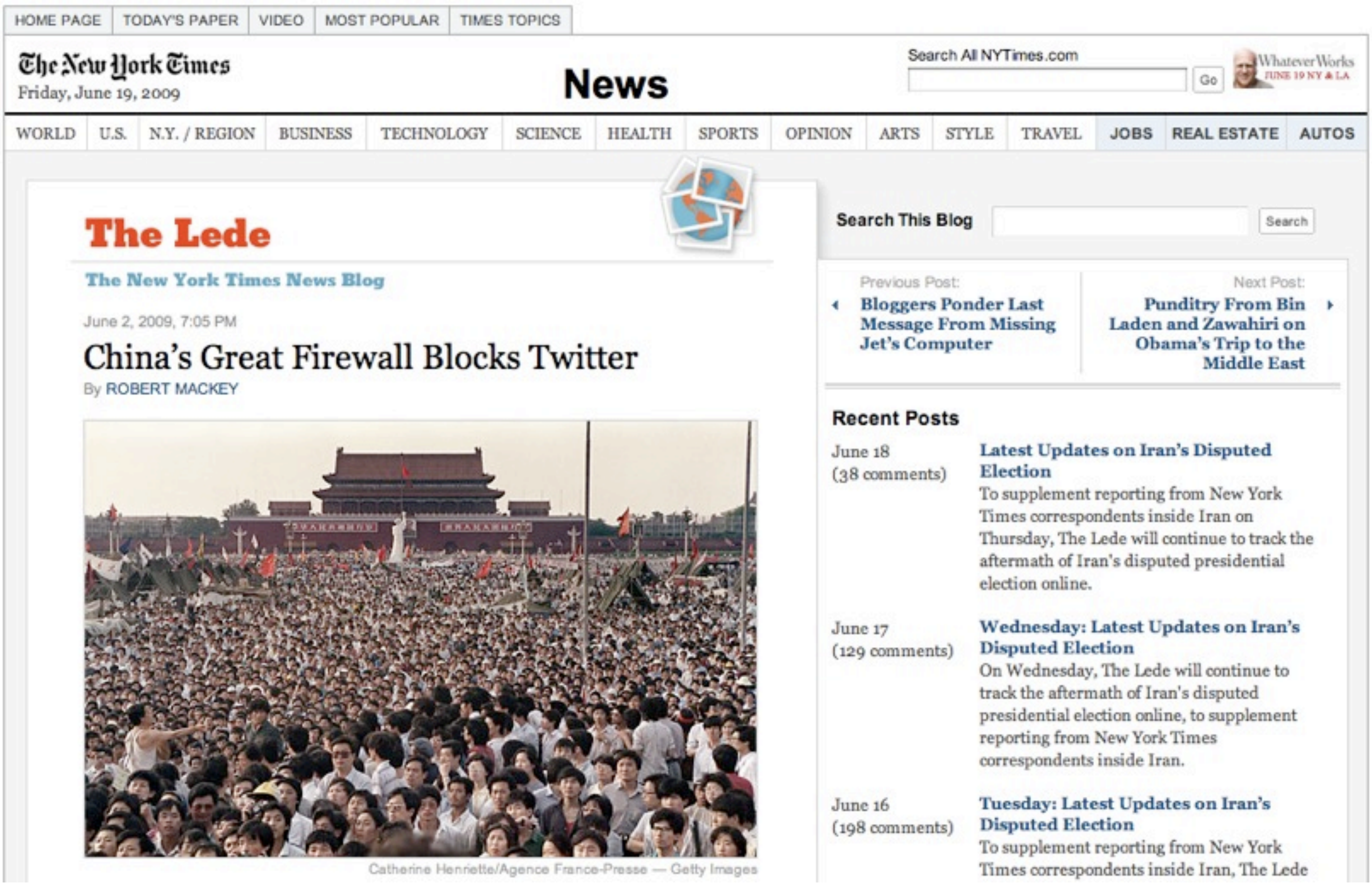
10 Largest Countries		
1.	United States	86,406,460
2.	United Kingdom	20,214,180
3.	Turkey	13,104,960
4.	Canada	12,862,140
5.	France	12,245,140
6.	Italy	11,573,640
7.	Indonesia	9,642,620
8.	Australia	6,572,900
9.	Spain	6,554,500
10.	Argentina	6,380,080

10 Fastest Growing Over Past Week			
1.	China	100.58 %	6,920
2.	Taiwan	11.14 %	322,900
3.	Vietnam	8.91 %	74,460
4.	Philippines	6.77 %	360,360
5.	Iraq	6.05 %	4,800
6.	Romania	5.17 %	15,300
7.	Sweden	5.11 %	127,760
8.	Ireland	5.1 %	47,220
9.	Ukraine	4.81 %	7,780
10.	Qatar	4.49 %	8,500

Global Internet Traffic

Alexa as of May 2009	China	USA	Japan	India	Brazil	Global
1	Baidu	Google	Yahoo.jp	Google.in	Google	Google
2	**QQ**	Yahoo	**FC2**	Google	**Orkut.br**	Yahoo
3	Sina	**Facebook**	Google.jp	Yahoo	Windows Live	**YouTube**
4	Google.cn	**YouTube**	**YouTube**	**Orkut.in**	Universo Online	**Facebook**
5	Taobao	**Myspace**	Rakuten	**YouTube**	**YouTube**	Windows Live
6	163	MSN	Livedoor	**Blogger**	Globo	MSN
7	Google	Windows Live	**Ameblo.jp**	Rediff	MSN	**Wikipedia**
8	Sohu	**Wikipedia**	**mixi**	**Facebook**	Google	**Blogger**
9	Youku	Craigslist	**Wikipedia**	**Wikipedia**	Yahoo	Baidu
10	Yahoo	EBay	Google	Windows Live	Terra	**Myspace**

Twitter in Spotlight



Topics in Social Computing

- Social Behavior Analysis and Modeling
- Social Media
- Social Network Theory and Models
- Link Analysis/Graph Mining/ Large Graph Algorithms
- Recommender Systems/ Collaborative Filtering
- QA/Sentiment Analysis/ Opinion Mining
- Human Computation/ Crowdsourcing
- Risk, Trust, Security, and Privacy
- Monetization of Social Computing
- Software Tools and Applications
- and many, many more...

Outline

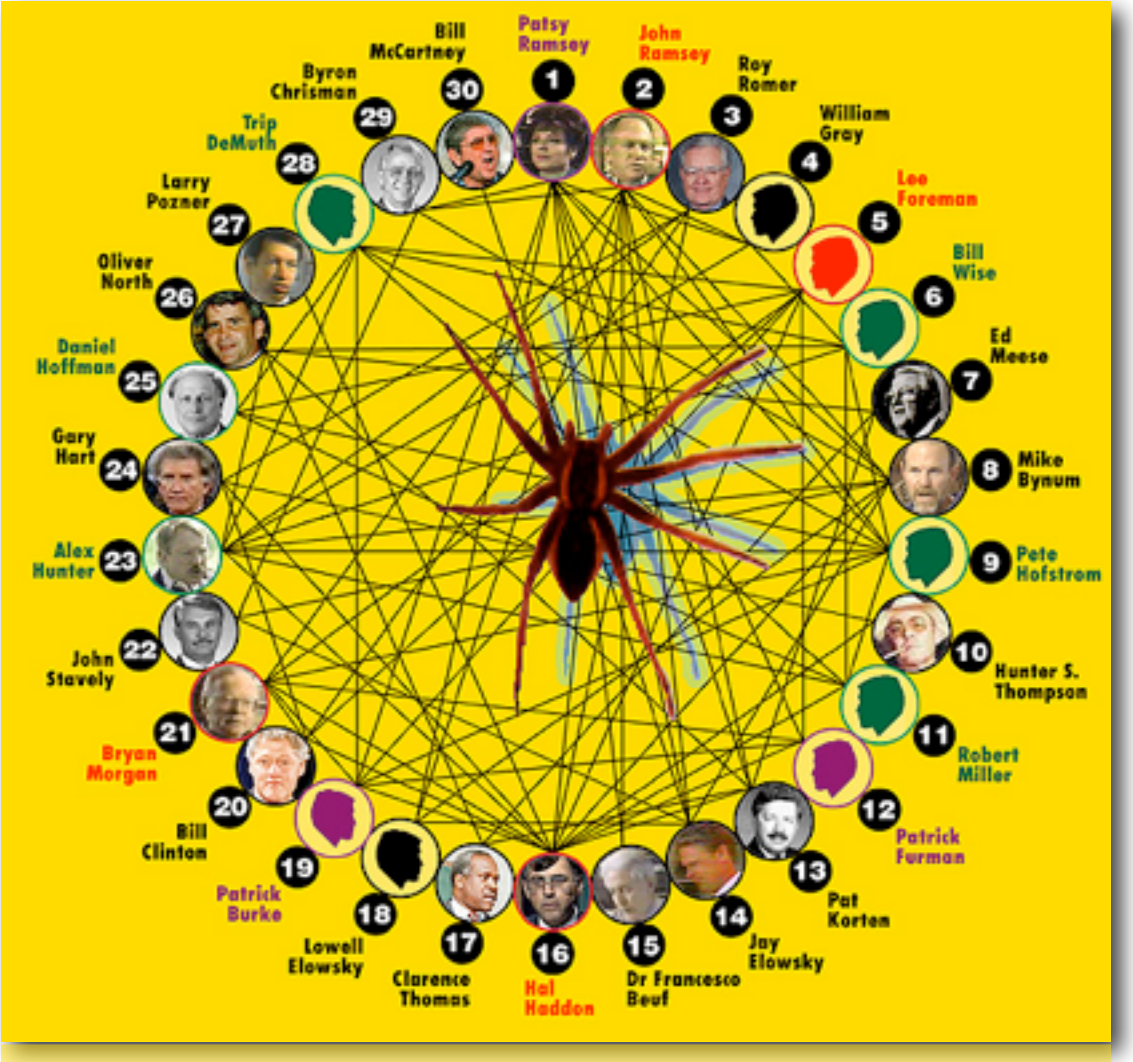
- Introduction to Social Computing
- ~~Graph Mining~~
- Link Analysis
- Learning to Rank
- Query Suggestion
- ~~Collaborative Filtering~~
- ~~Human Computation~~
- Privacy and Trust in Social Network

Web 2.0

- Web as a medium vs. **Web as a platform**
- Read-Only Web vs. **Read-and-Write Web**
- Static vs. **Dynamic**
- Restrictive vs. **Freedom & Empowerment**
- Technology-centric vs. **User-centric**
- Limited vs. **Rich User Experience**
- Individualistic vs. **Group/Collective Behavior**
- Consumer vs. **Producer**
- Transactional vs. **Relational**
- Top-down vs. **Bottom-up**
- People-to-Machine vs. **People-to-People**
- Search & browse vs. **Publish & Subscribe**
- Closed application vs. **Service-oriented Services**
- Functionality vs. **Utility**
- Data vs. **Value**

Social Networks

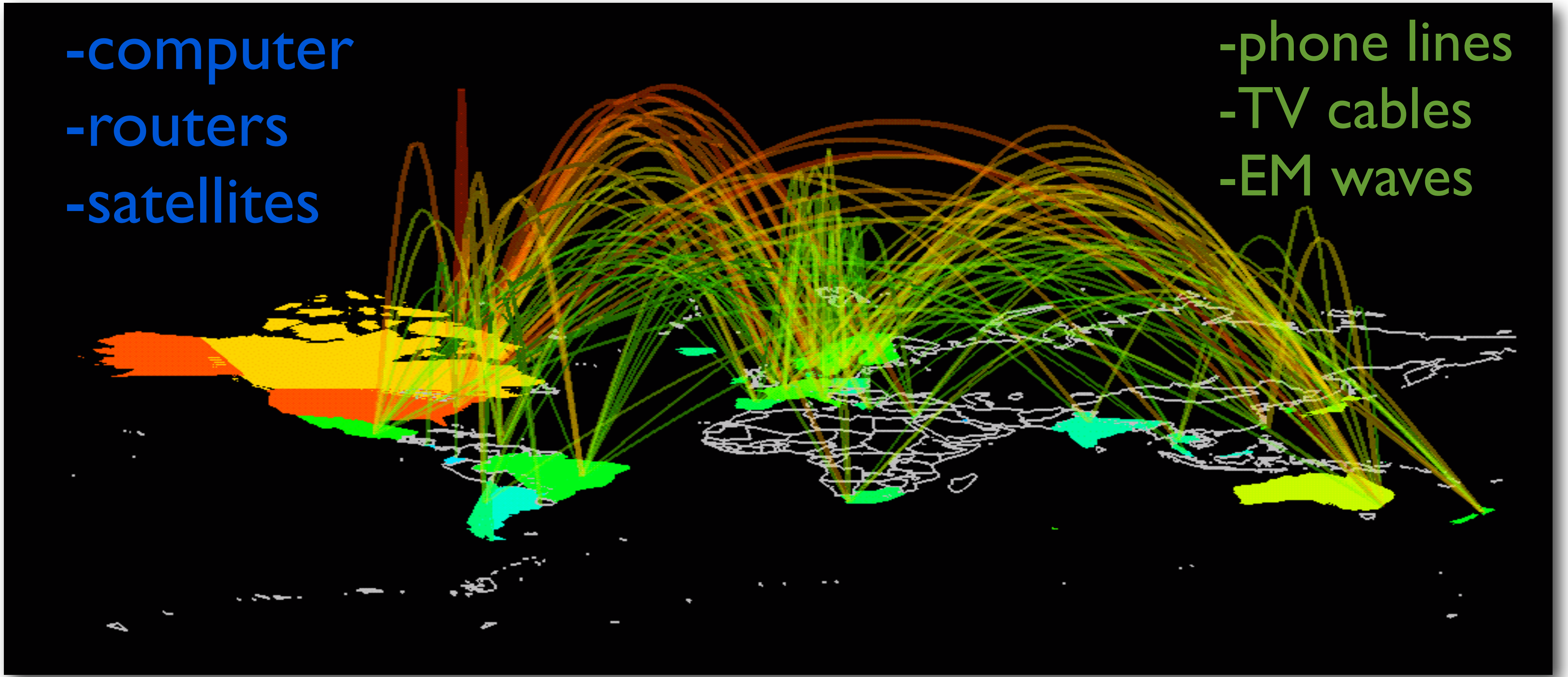
Society:
Nodes: individuals
Links: social relationship (family/work/friendship/etc.)



S. Milgram and John Guare: Six Degree of Separation.
Social networks: Many individuals with diverse social interactions between them.

Social Networks

- The Earth is developing an electronic nervous system, a network with diverse nodes and links.



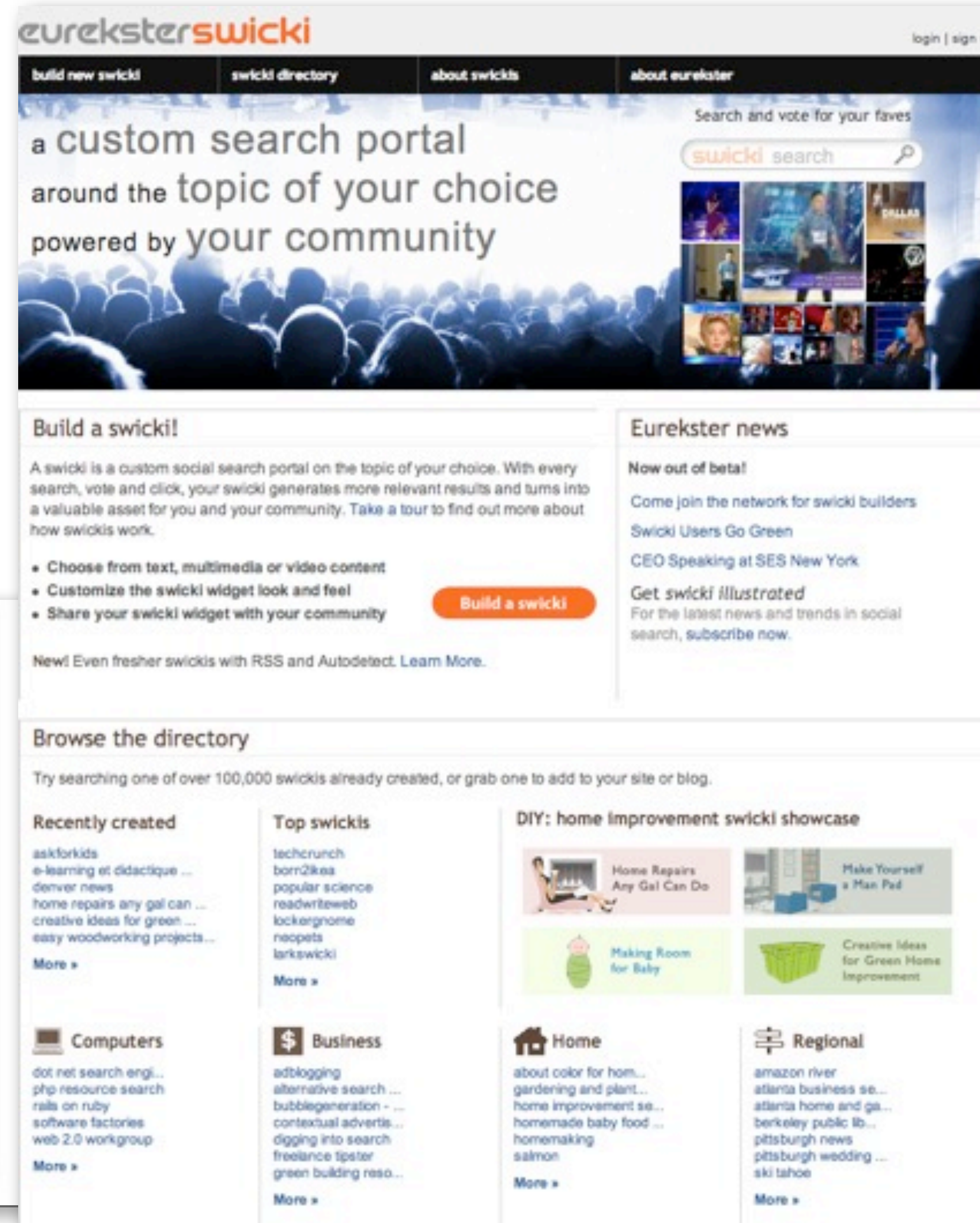
Communication networks: many non-identical components with diverse connections between them.

Social Networking Sites

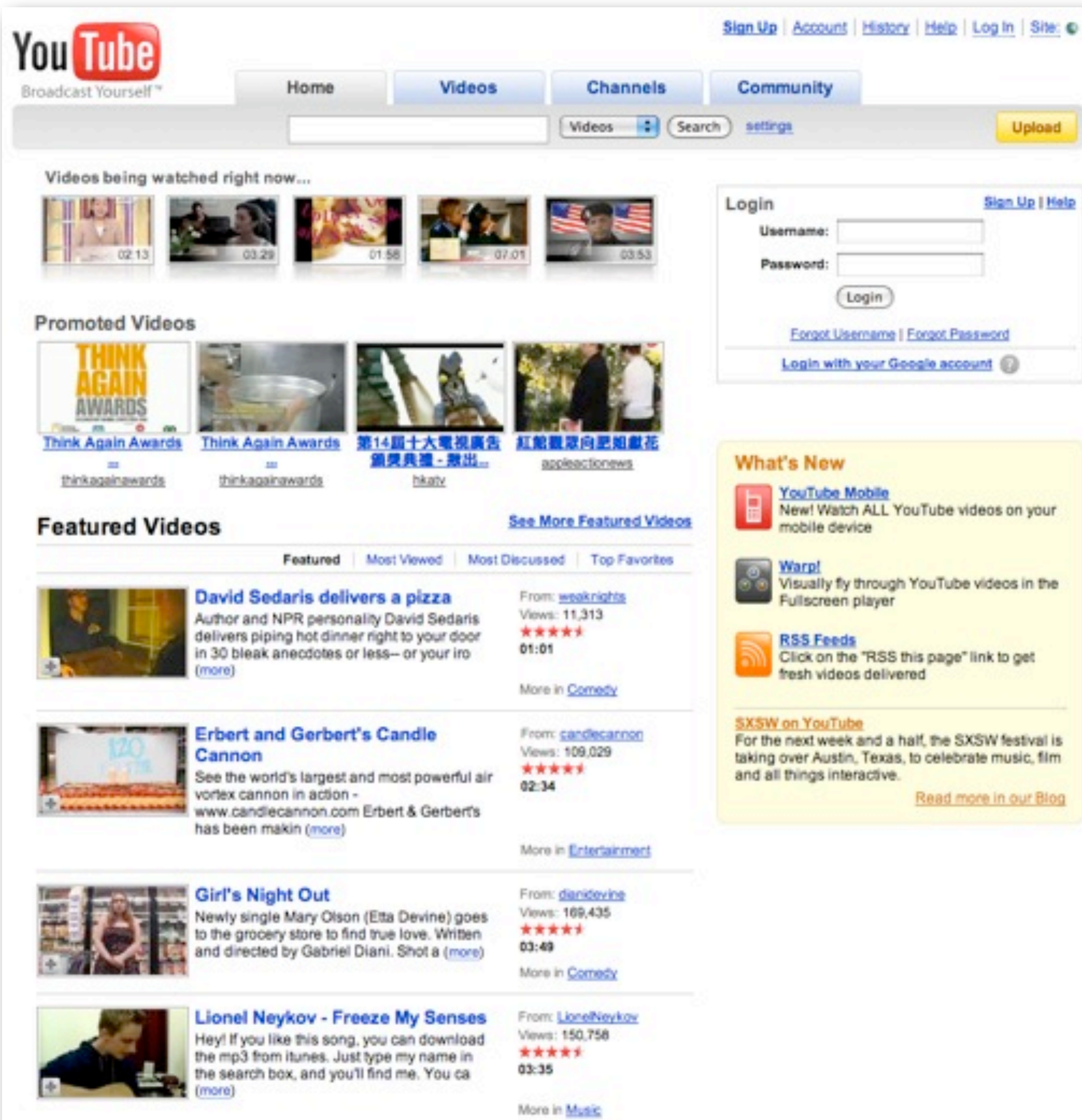
- Example of Social Networking Sites: FaceBook, MySpace, Blogger, QQ, etc.

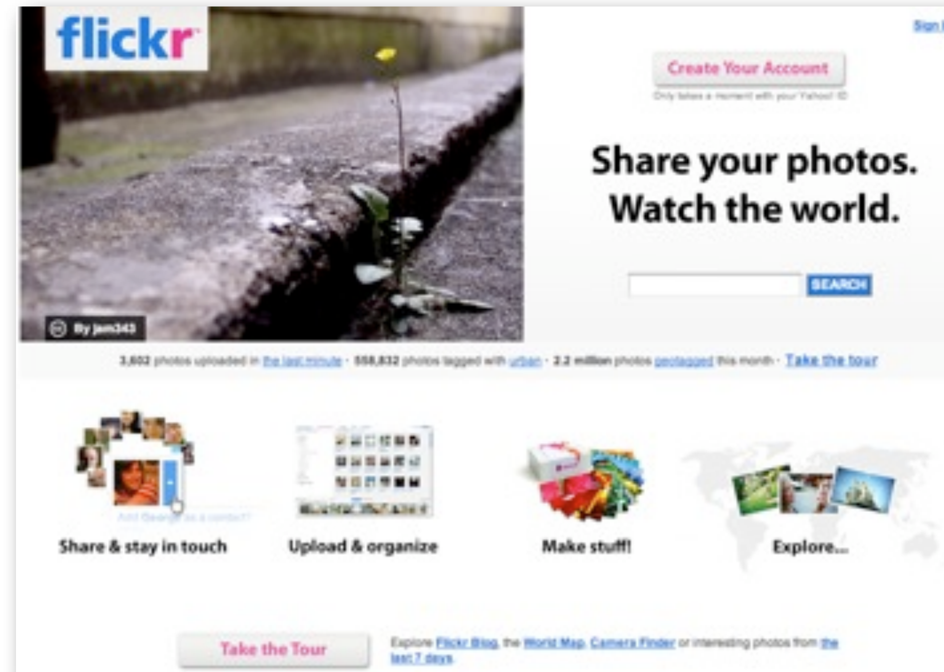
Social Search

- Social Search Engine
- Leveraging your social networks for searching

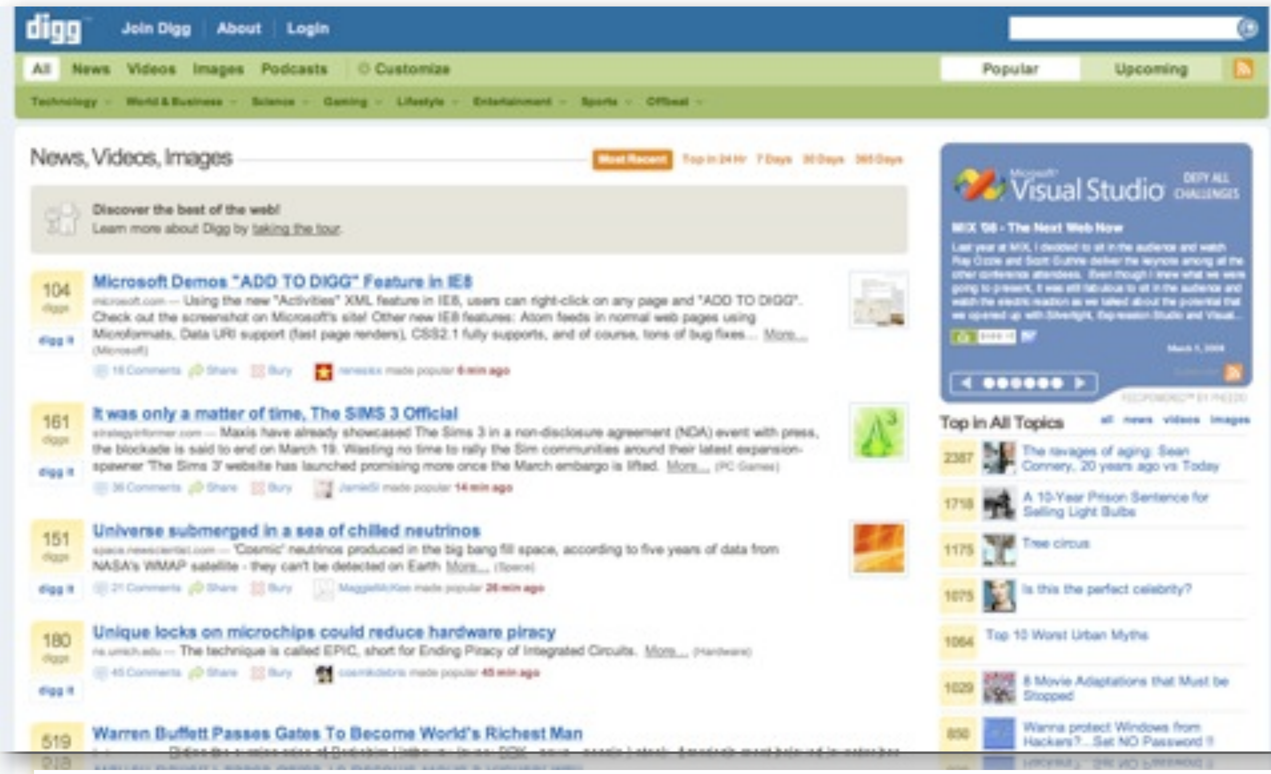


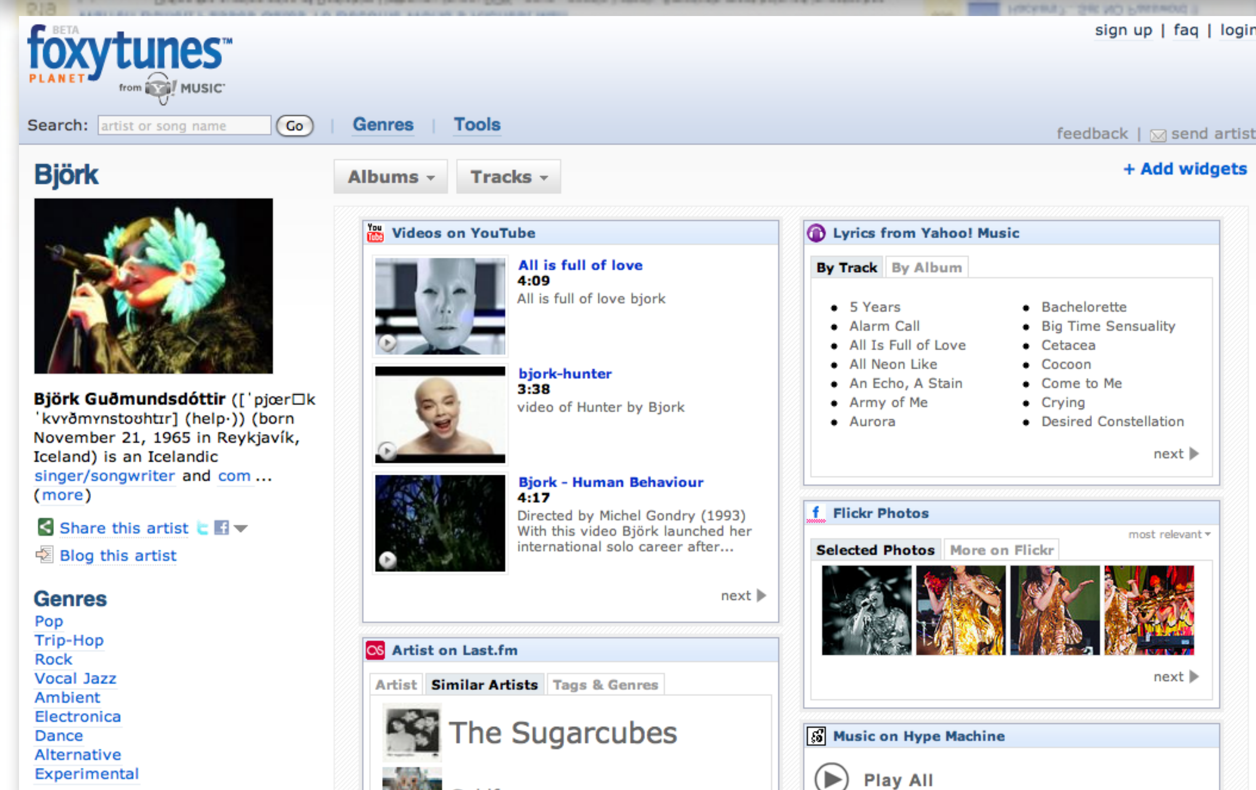
Social Media

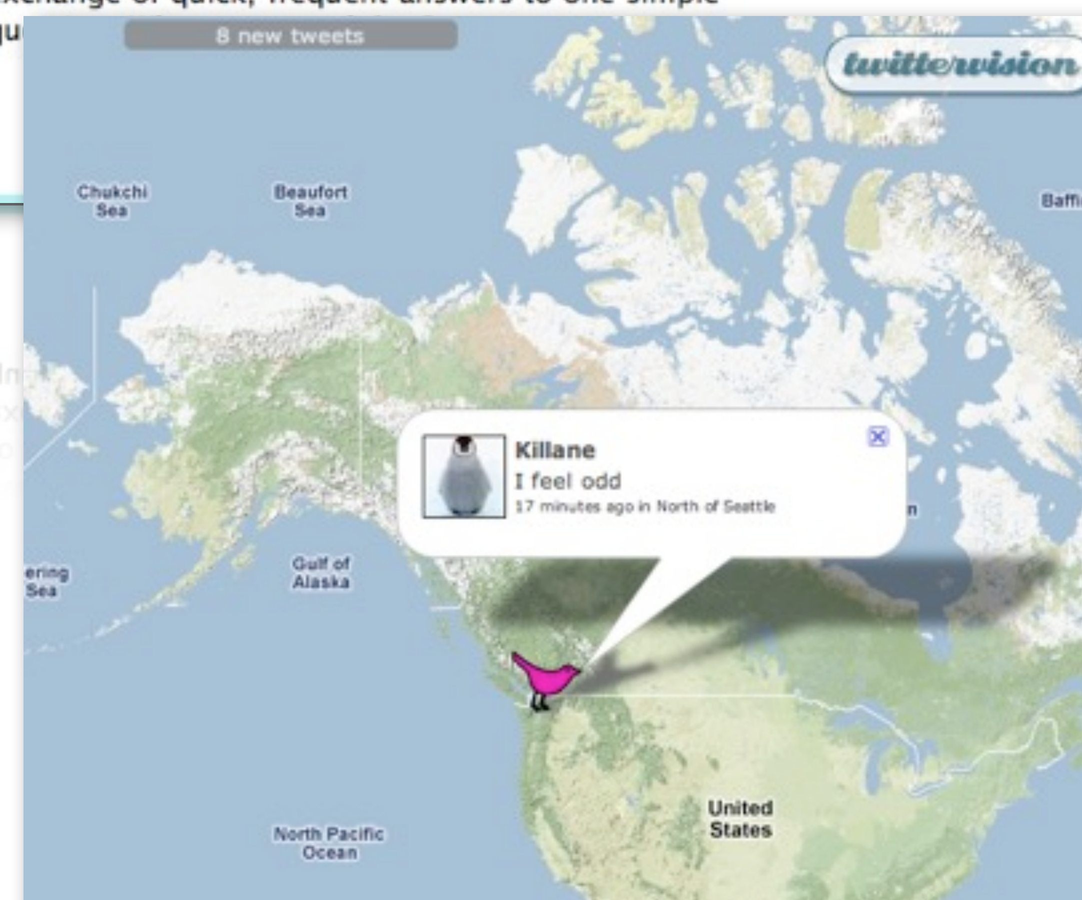




Social News/Mash Up

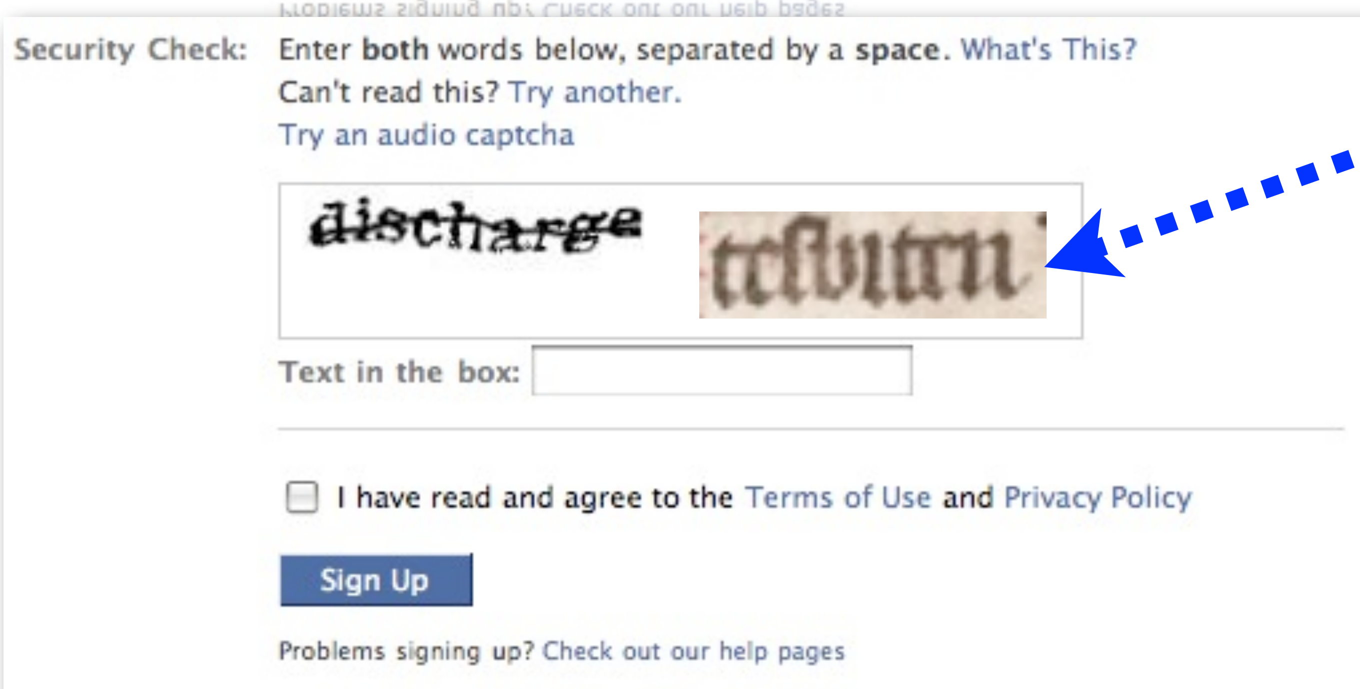


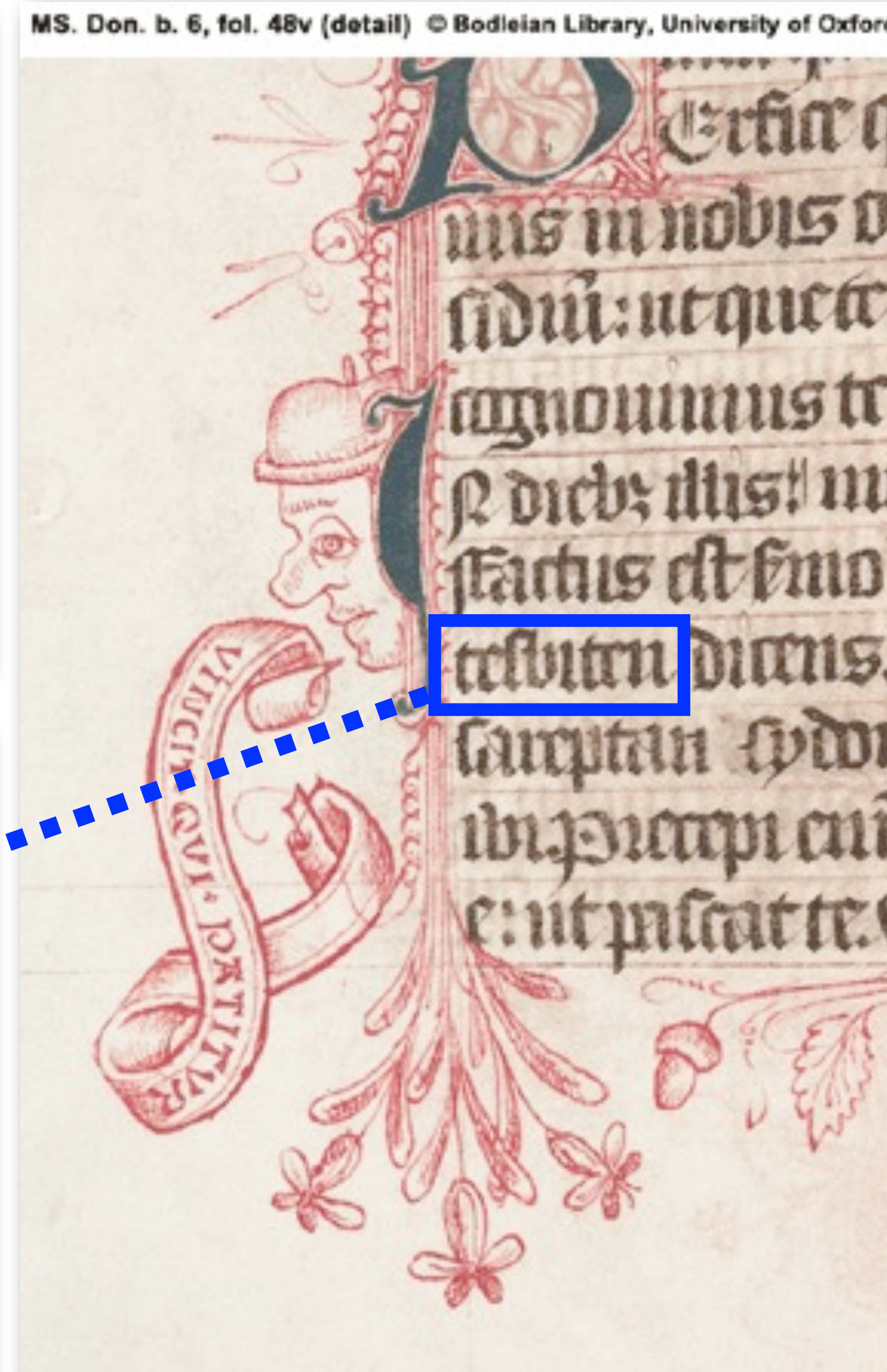




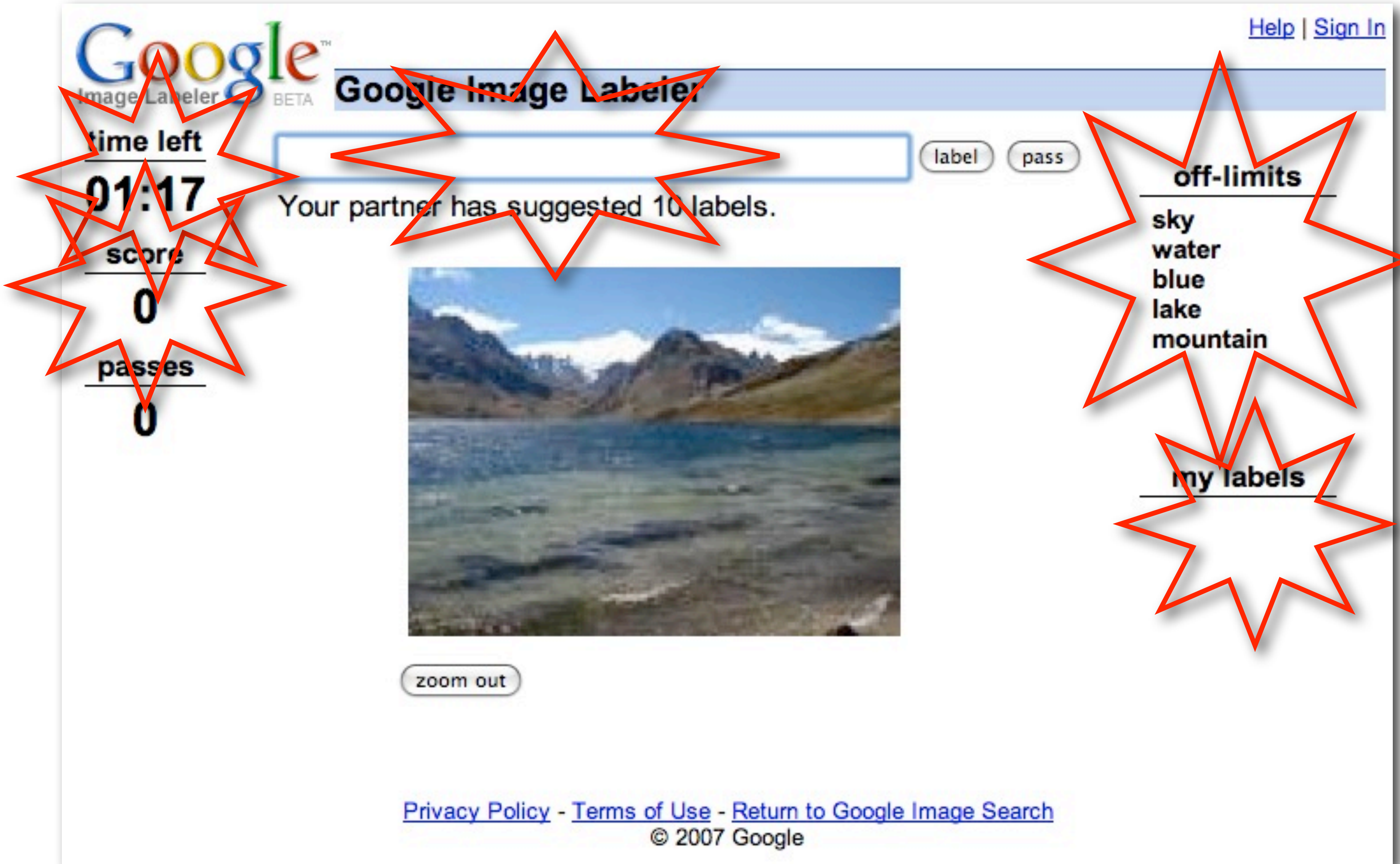
Social Knowledge Sharing

Social/Human Computation





Human Computation



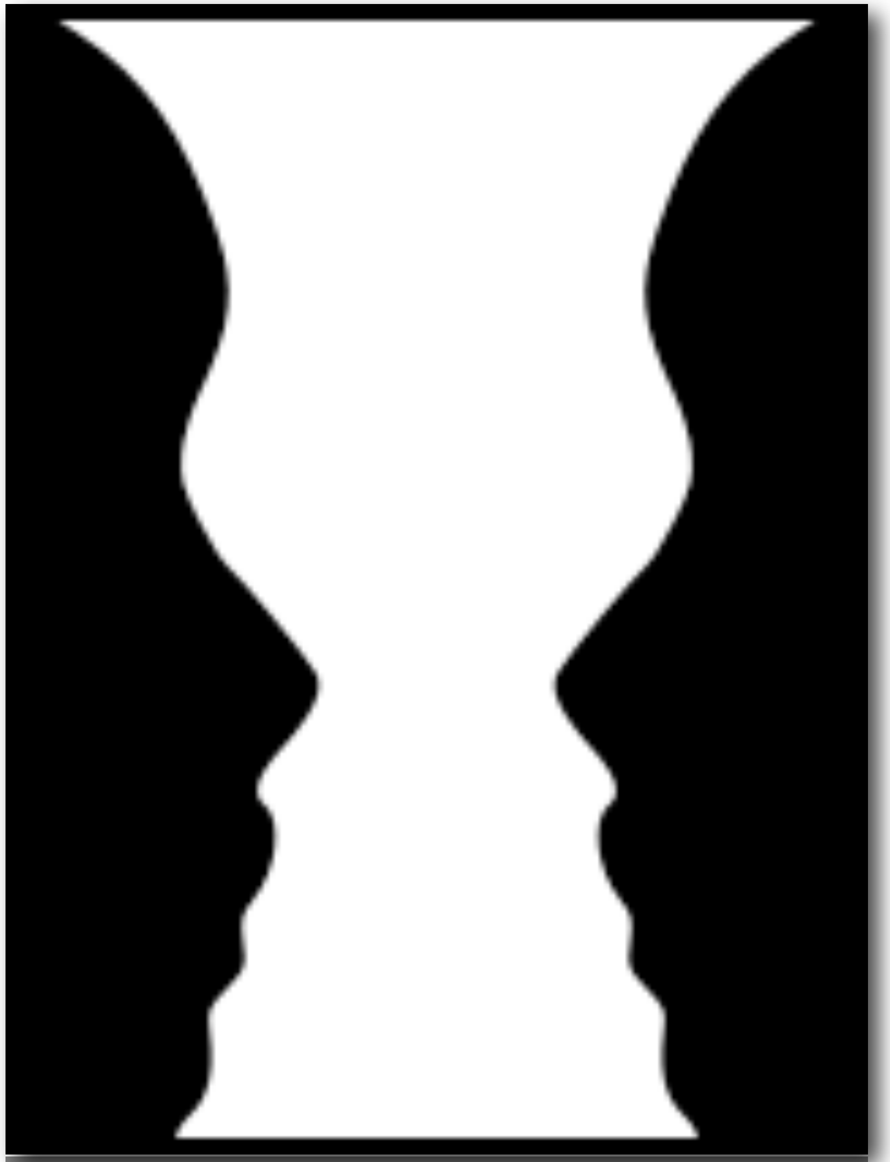
Web 2.0 Revolution

- Glocalization-think globally and act locally!
- Weblication-Web is the application!
- Three C's

  Connectivity

  Collaboration

  Communities



Social Relations

crew
teams
populations
squad
organizations
cohorts
markets
communities
partners
groups

binary
cardinal
integer
real

presence
identity
social role
reputation
expertise
trust
ownership
accountability
knowledge

Social Computing

social network services

wikis

blogs

emails

instant messaging

mobile devices

social bookmarking

ranking

tagging

collaborative filtering

social marketing

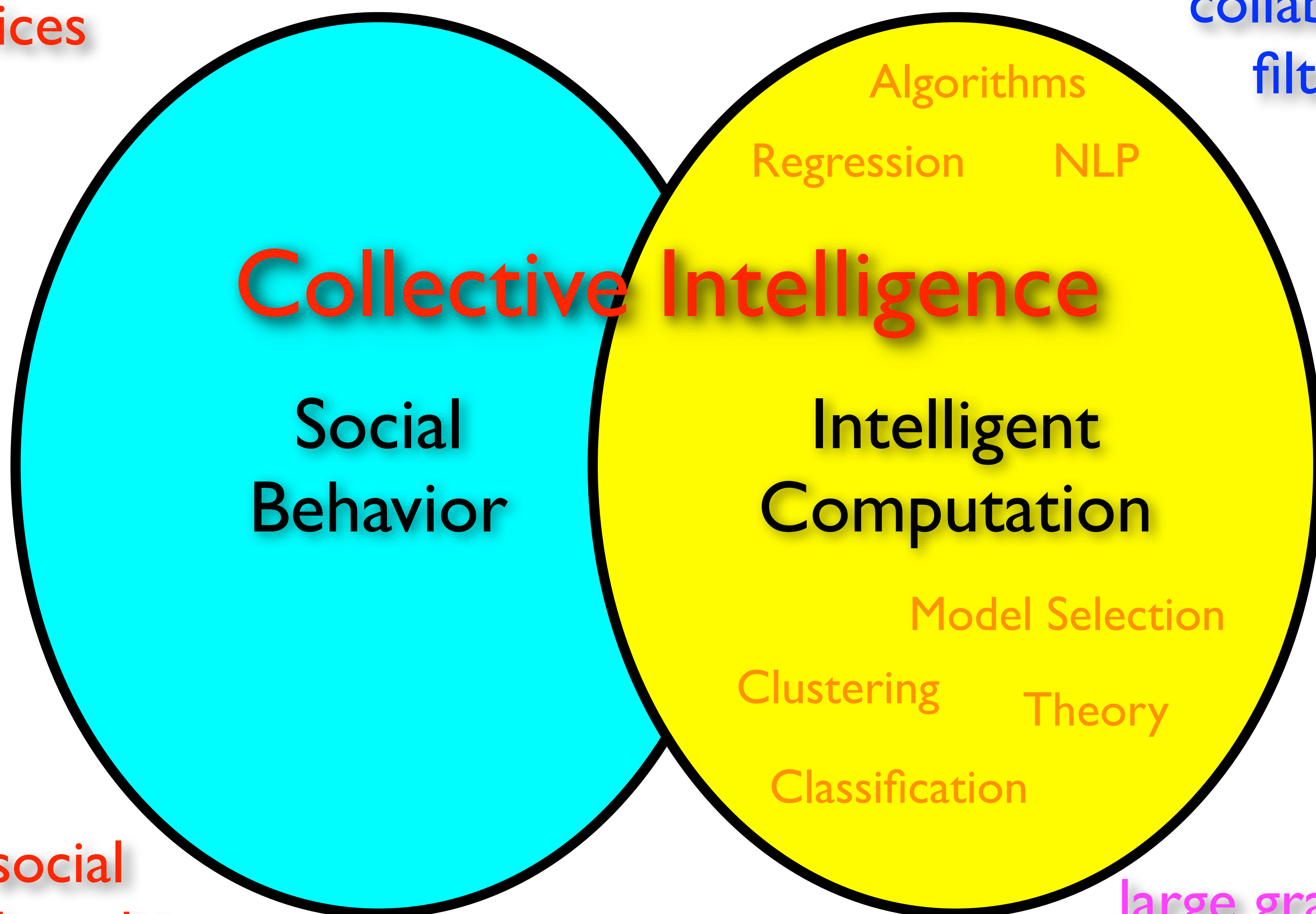
human computation

opinion mining/ sentiment analysis

query logs analysis

large graph algorithms

security & privacy



Definition of Social Computing

[wiki]

- Any Computer-mediated communication and interaction
- In the weaker sense: supporting any sort of social behavior
  - blogs, email, instant messaging, wiki, social network services, social bookmarking
- In the stronger sense: supporting "computations" that are carried out by a group of people
  - collaborative filtering, online auctions, prediction markets, reputation systems, tagging, verification games

Emerging Issues

- Theory and models
- Seach, mining, and ranking of existing information, e.g., spatial (relations) and temporal (time) domains
    - Dealing with partial and incomplete information, e.g., collaborative filtering, ranking, tagging, etc.
- Scalability and algorithmic issues
- Security and privacy issues
- Monetization of social interactions

Computational Perspective

- Classification, clustering, regression, etc.
- New insights on the data
  - Social relations are often hidden (latent)
  - Change data from (x, y) to $(x, c_1(x), c_2(x), \cdots, y)$
- $c(x)$ = context in *tags*, *relations*, *ratings*, etc.
- data type = *binary*, *integer*, *real*, *cardinal*, etc.

Social Network Theory

- Consider many kinds of networks:
  - social, technological, business, economic, content, ...
- These networks tend to share certain informal properties:
  - large scale; continual growth
  - distributed, organic growth: vertices "decide" who to link to
  - interaction restricted to links
  - mixture of local and long-distance connections
  - abstract notions of distance: geographical, content, social,…

Social Network Theory

- Do these networks share more quantitative universals?
- What would these "universals" be?
- How can we make them precise and measure them?
- How can we explain their universality?
- This is the domain of social network theory

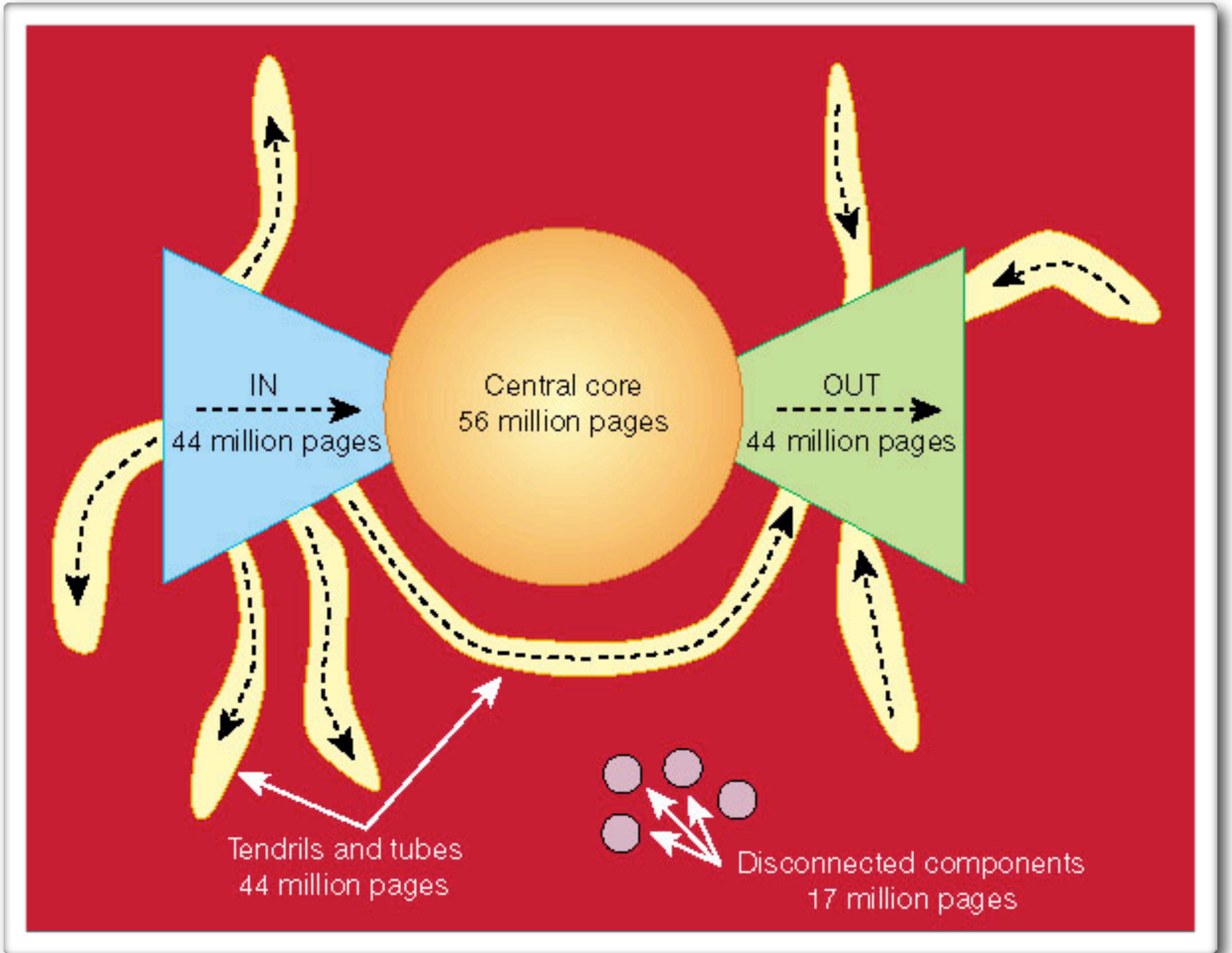
Some Interesting Quantities

- Connected components
  - how many, and how large?
- Network diameter
  - maximum (worst-case) or average?
  - exclude infinite distances? (disconnected components)
  - the small-world phenomenon
- Clustering
  - to what extent that links tend to cluster "locally"?
  - what is the balance between local and long-distance connections?
  - what roles do the two types of links play?
- Degree distribution
  - what is the typical degree in the network?
  - what is the overall distribution?

Link Analysis

Irwin King
Department of Computer Science and Engineering
The Chinese University of Hong Kong
http://wiki.cse.cuhk.edu.hk/irwin.king/home

What Does the Web Look Like?



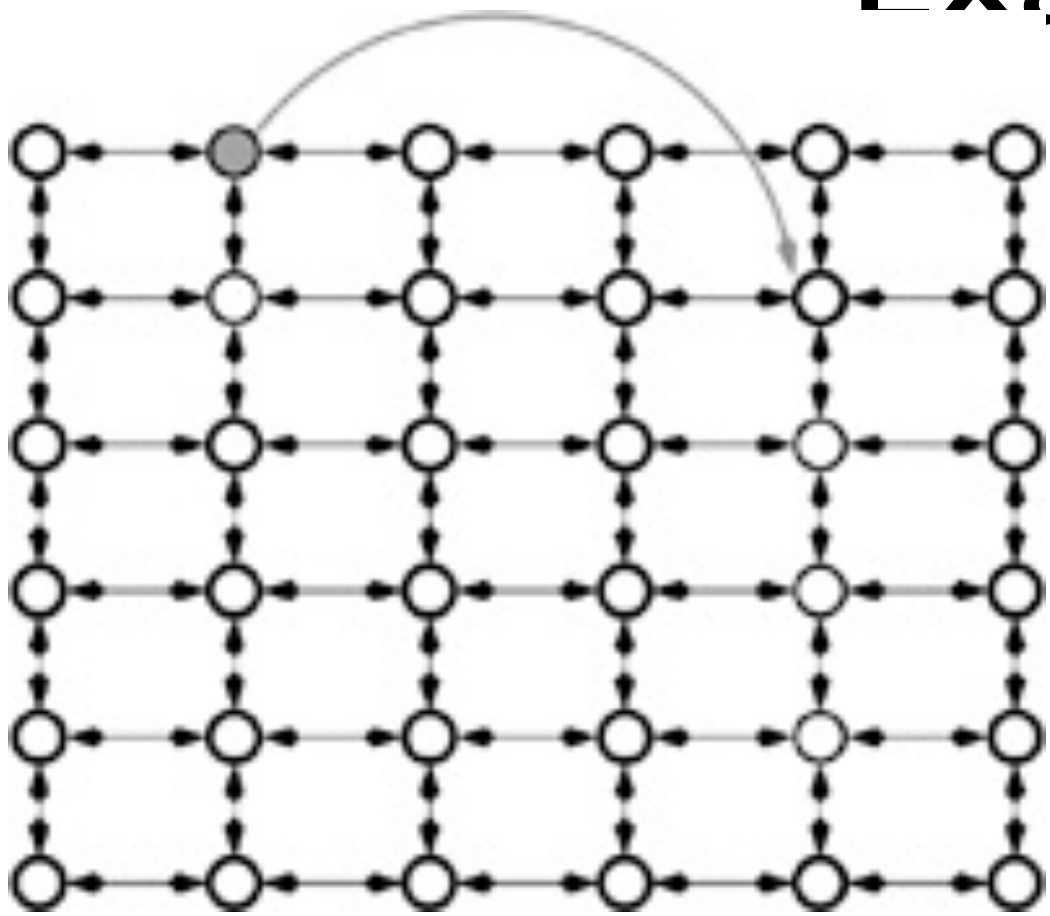
Small-World Phenomenon

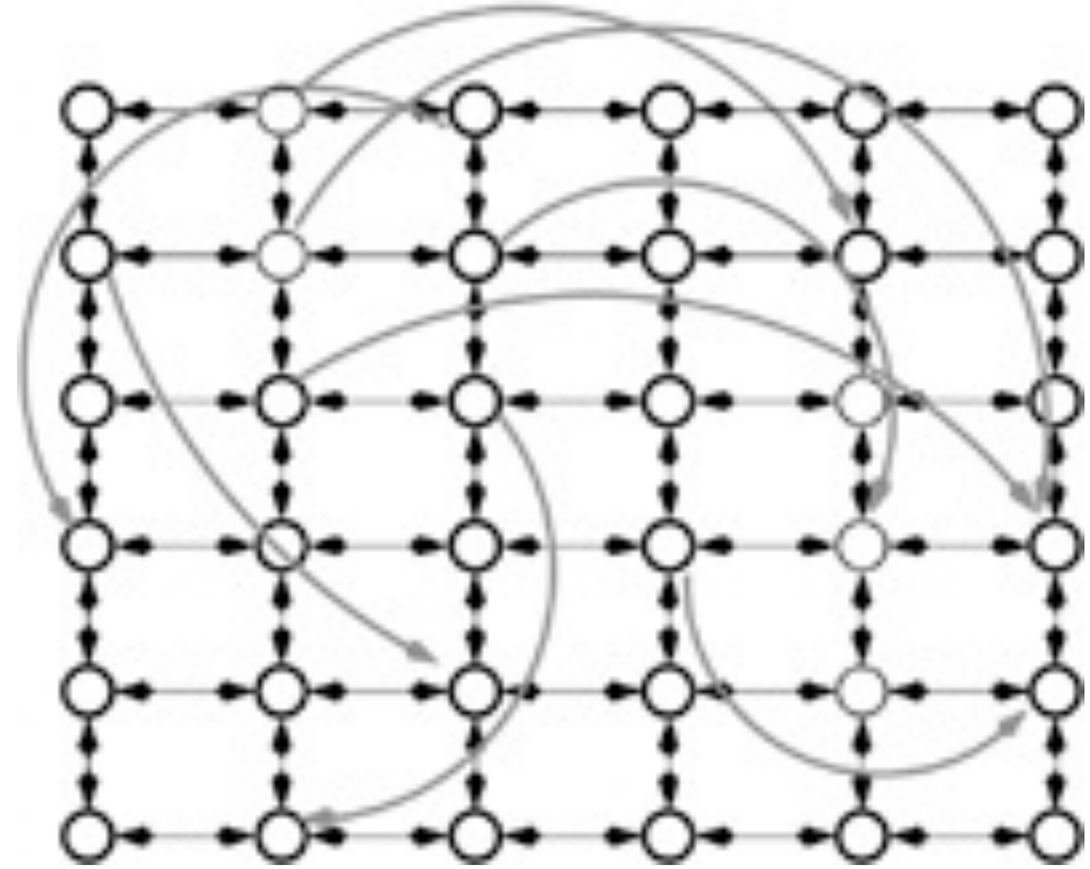
- We are all linked by short chains of acquaintances, or "six degrees of separation"
- An abundance of short paths in a social network graph
- Started by a Social Psychologist Stanley Milgram in the 1960s with two important discoveries
  - The existence of short paths among people
  - People in society, with knowledge of only their own personal acquaintances, were collectively able to forward the letter to a distant target so quickly
- The power of an effective routing algorithm--equipped with purely local information, to find efficient paths to a destination; that such a decentralized routing scheme is effective

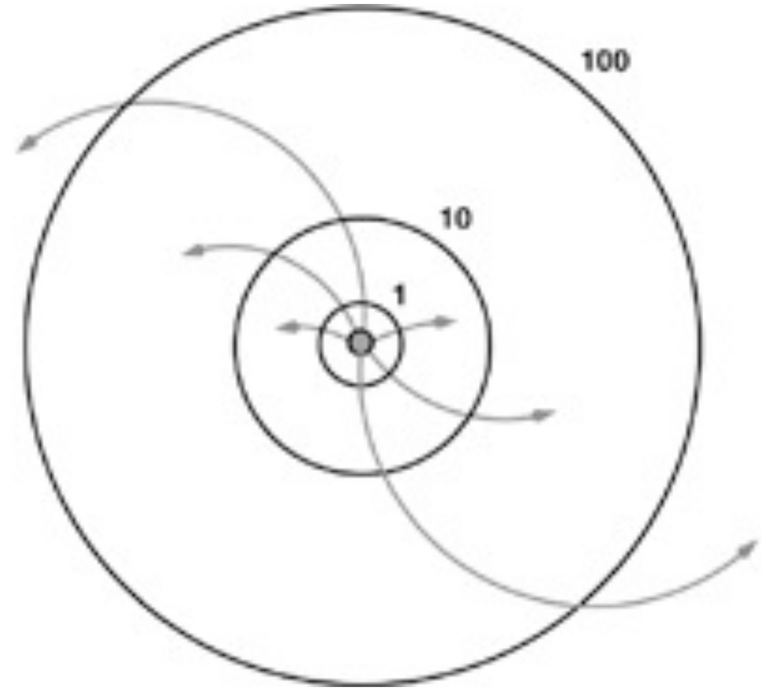
Watts and Strogatz

- Highly clustered sub-network consisting of the "local acquaintances" of nodes
- A collection of random long-range shortcuts
- Start with a *d*-dimensional lattice network, and add a small number of long-range links out of each node, to destinations chosen uniformly at random
- In the model of a *d*-dimensional lattice with uniformly random shortcuts, no decentralized algorithm can find short paths (so short paths exist, but local knowledge does not suffice to construct them!)
- However, add links between nodes of this network with a probability that decays like the *d*-th power of their distance (in *d* dimensions). It is quite useful in P2P networks in sharing local information for decentralized searching.

Examples







Traditional Information Retrieval

- Content matching against the query
  - Occurrence of query words
  - Location of query words
  - Document weighting
- Not much of ranking
- Science Citation Index and Impact Factor

Challenges of Web Search

- Voluminous
- Dynamic (generated deep web)
- Self-organized
- Hyperlinked
- Quality of Information
- Accessibility

The PageRank Algorithm

- Hyperlinked documents are different!
  - Similar to academic papers
  - In-links = authorities
  - Out-links = citations
  - Citations give better approximation of the quality of pages

Define PageRank

The PageRank calculation is defined as follows. We assume page A has pages $T_1, \cdots, T_n$ which point to it (i.e., are citations). The parameter d is a damping factor which can be set between 0 and 1. $C(A)$ is defined as the number of links going out of page A. The PageRank of a page A is given as follows:

$$PR(A) = (1-d) + d(PR(T_1)/C(T_1) + \cdots + PR(T_n)/C(T_n)). \qquad (1)$$

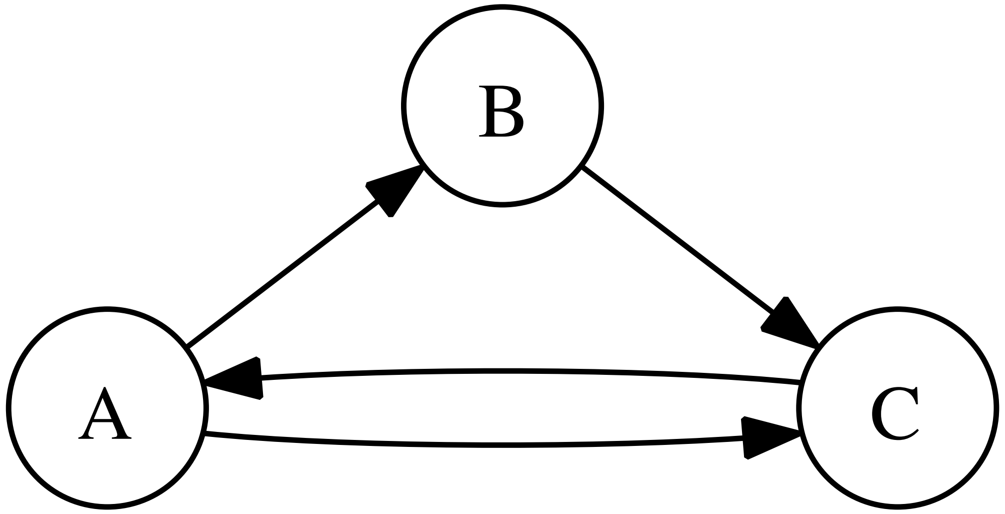
$$PR(A) = (1-d) + d\sum_{i}^{n} \frac{PR(T_i)}{C(T_i)}.$$

- PageRanks form a probability distribution over web pages, so the sum of all web pages' PageRanks will be one
- It can be calculated using a simple iterative algorithm, and corresponds to the principal eigenvector of the normalized link matrix of the web

Assumptions

- A "random surfer" who is given a web page at random
- The surfer keeps clicking on links, never hitting "back"
- The surfer gets bored and starts on another random page
- The probability that the random surfer visits a page is its PageRank
- The *d* damping factor is the probability at each page the Surfer will get bored and request another random page.
- Instead of a global *d*, one may consider a page damping factor d_i for each individual page or a group of pages

Examples



$$d = 0.5 \quad (1)$$

$$PR(A) = 0.5 + 0.5(PR(A)/2) \quad (2)$$

$$PR(C) = 0.5 + 0.5(PR(A)/2 + PR(B)) \quad (3)$$

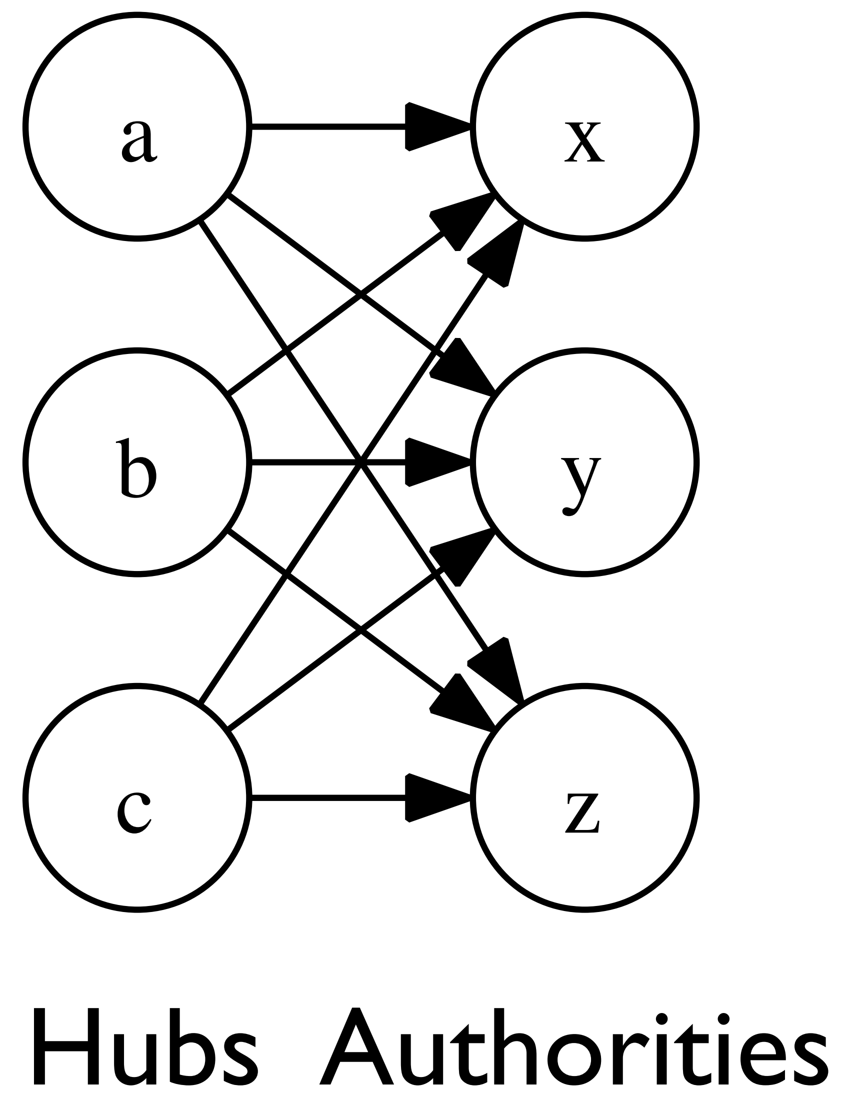
$$PR(A) = 14/13 = 1.07692308 \quad (4)$$

$$PR(B) = 10/13 = 0.76923077 \quad (5)$$

$$PR(C) = 15/13 = 1.15384615 \quad (6)$$

Kleinberg's Algorithm

- Web page importance should depend on the search query being performed
- Each page should have a separate "authority" rating (based on the links going to the page) that captures the quality of the page as a resource itself
- Each page should also have a "hub" rating (based on the links going from the page) that captures the quality of the pages as a pointer to useful resources



Define HITS Algorithm

- The HITS (Hyperlink Induced Topic Distillation) algorithm computes lists of hubs and authorities for WWW search topics
- Start with a search topic, specified by one or more query terms
    - Sampling Stage--constructs a focused collection of several thousand Web pages likely to be rich in relevant authorities
    - Weight-propagation Stage-- determines numerical estimates of hub and authority weights by an iterative procedure
- The pages with the highest weights are returned as hubs and authorities for the search topic

The HITS Algorithm

Let the Web be a digraph $G = (V, E)$. Given a subgraph $S \subseteq V$ with $u, v \in S$ and $(u, v) \in E$. The authority and hub weights are updated as follows.

1. If a page is pointed to by many good hubs, we would like to increase its authority weight.

$$x_p = \sum_{q \text{ such that } q \to p} y_q, \tag{1}$$

   where the notation $q \to p$ indicates taht q links to p.

2. If a page points to many good authorities, we increase its hub weight

$$y_p = \sum_{q \text{ such that } p \to q} x_q. \tag{2}$$

The above can be rewritten in a matrix notation as

$$x \leftarrow A^T y \leftarrow A^T A x = (A^T A)x \tag{3}$$

and

$$y \leftarrow Ax \leftarrow AA^T y = (AA^T)y \tag{4}$$

The HITS Pseudocode

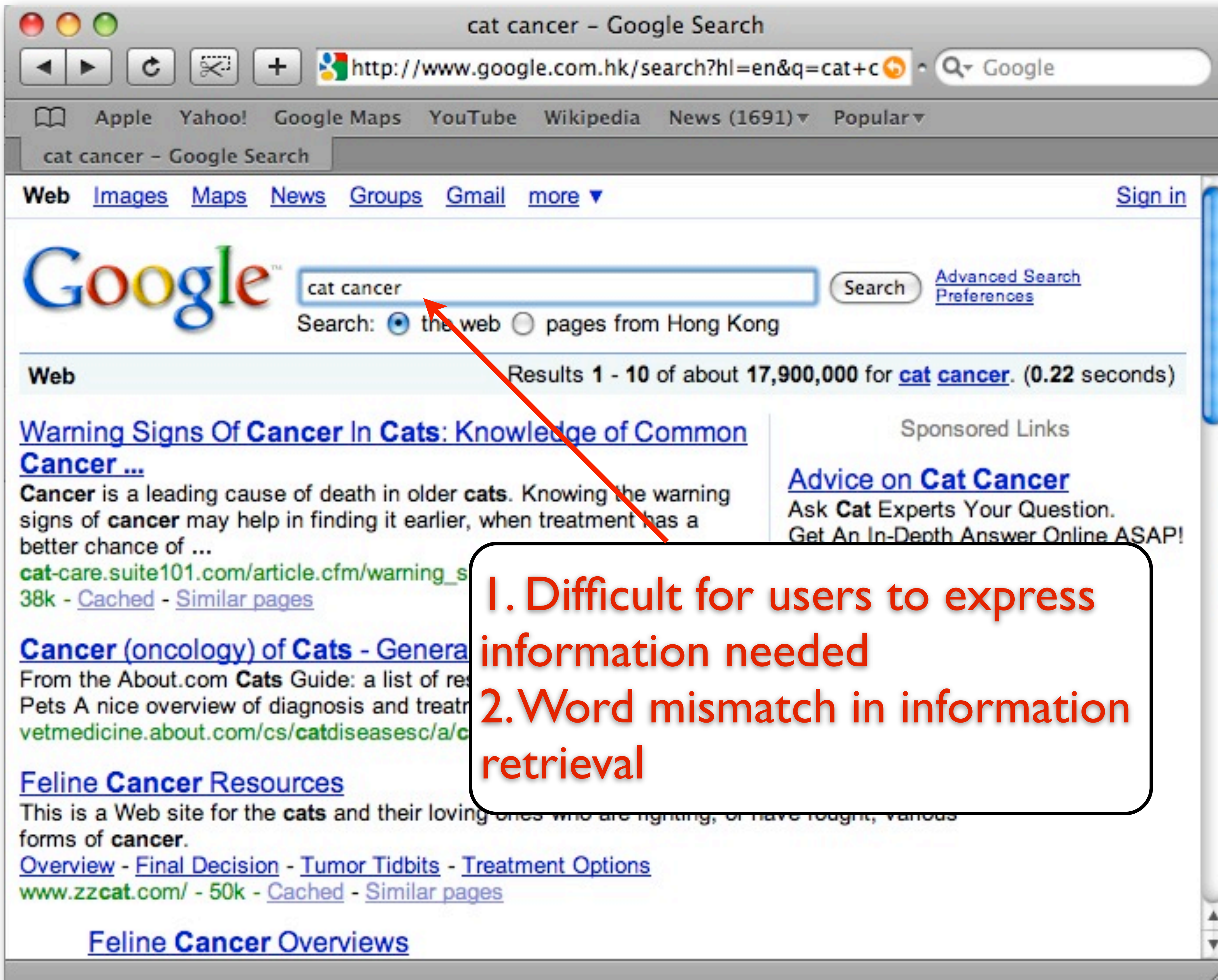
- It is executed at query time, not at indexing time
- The hub and authority scores assigned to a page are query-specific.
- It computes two scores per document, hub and authority, as opposed to a single score.
- It is processed on a small subset of 'relevant' documents, not all documents as was the case with PageRank.

```
G := set of pages
for each page p in G do
  p.auth = 1 // p.auth is the authority score of the page p
  p.hub = 1 // p.hub is the hub score of the page p
function HubsAndAuthorities(G)
  for step from 1 to k do // run the algorithm for k steps
    for each page p in G do  // update all authority values first
      for each page q in p.incomingNeighbors do // p.incomingNeighbors is the set of pages that link to p
        p.auth += q.hub
    for each page p in G do  // then update all hub values
      for each page r in p.outgoingNeighbors do // p.outgoingNeighbors is the set of pages that p links to
        p.hub += r.auth
```

Query Suggestion

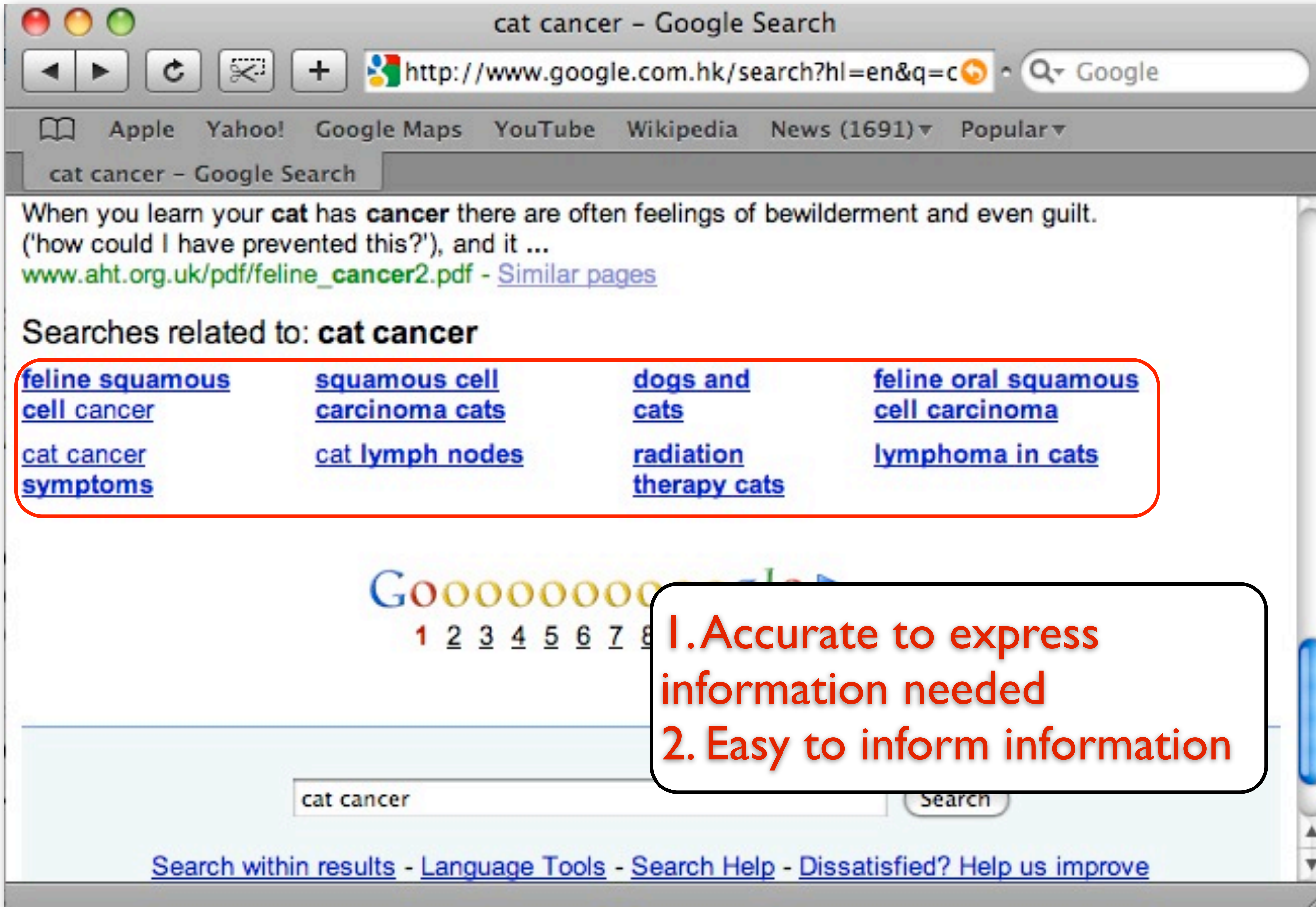
Irwin King
Department of Computer Science and Engineering
The Chinese University of Hong Kong
http://wiki.cse.cuhk.edu.hk/irwin.king/home

Motivation



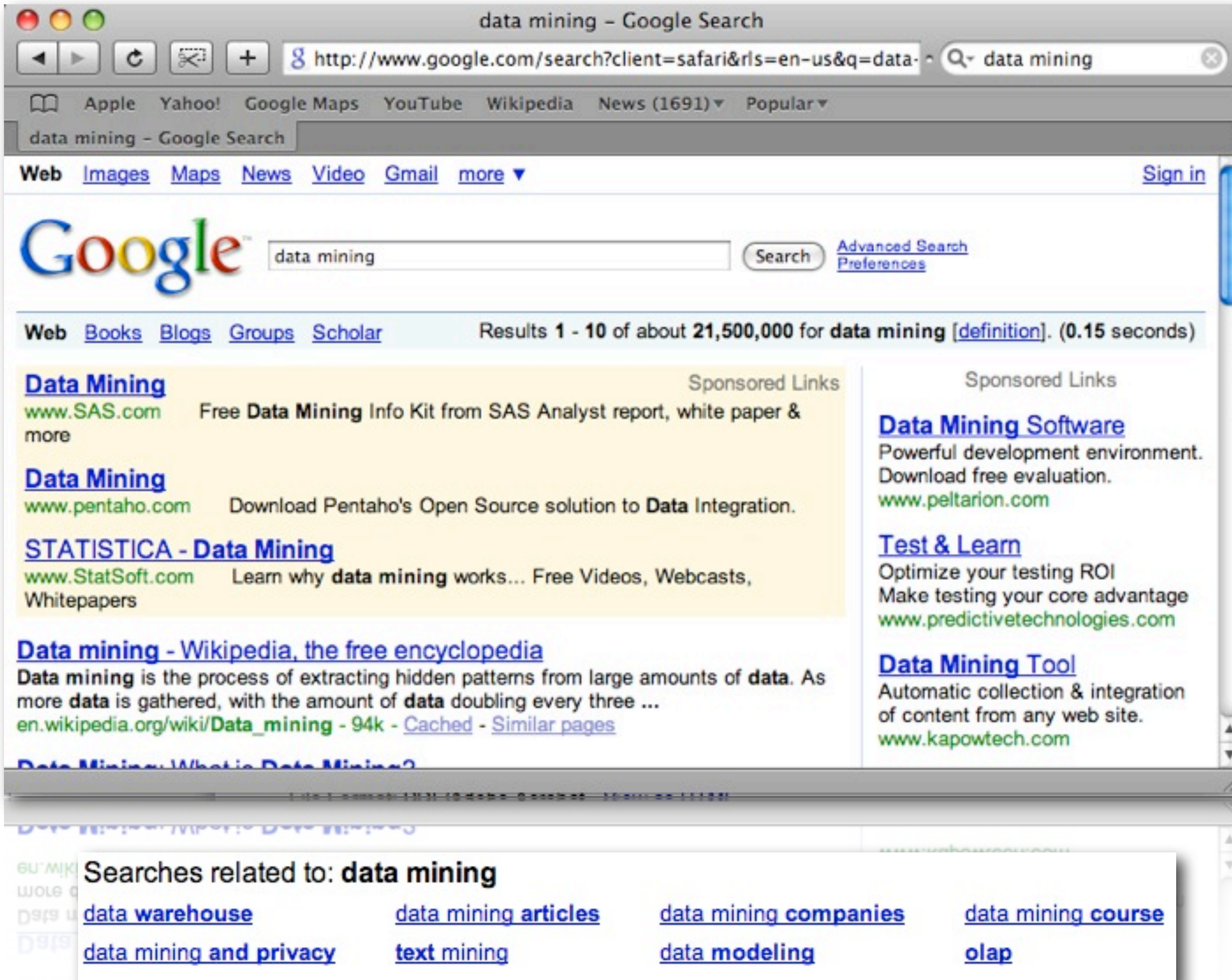
1. Difficult for users to express information needed
2. Word mismatch in information retrieval

Motivation



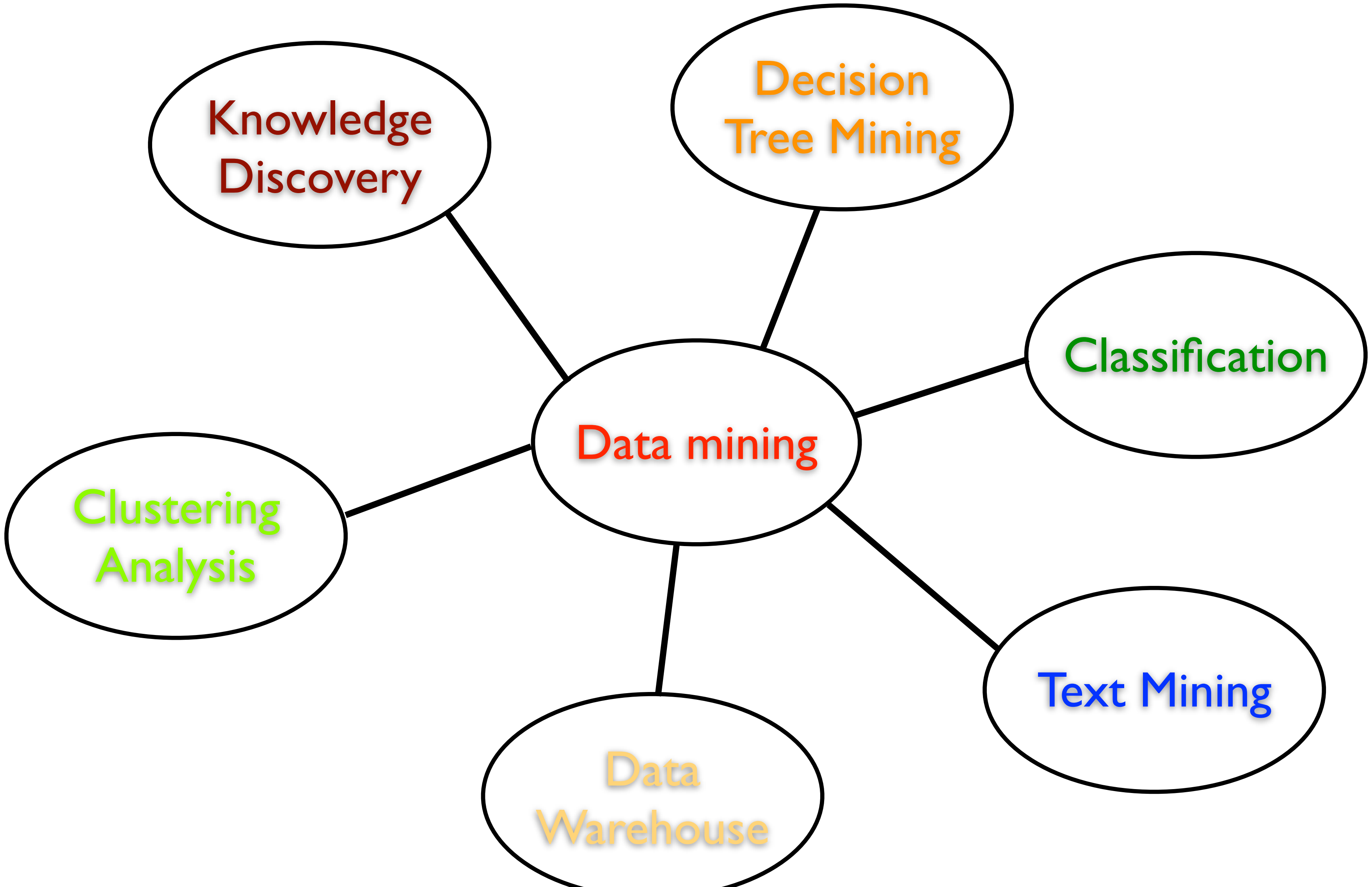
1. Accurate to express information needed
2. Easy to inform information

Motivation


data mining – Google Search
http://www.google.com/search?client=safari&rls=en-us&q=data-
data mining
Apple Yahoo! Google Maps YouTube Wikipedia News (1691) Popular
Web Images Maps News Video Gmail more ▼
Sign in
Google
Search
Advanced Search
Preferences
Web Books Blogs Groups Scholar
Results 1 - 10 of about 21,500,000 for data mining [definition]. (0.15 seconds)
Sponsored Links
Data Mining
www.SAS.com Free Data Mining Info Kit from SAS Analyst report, white paper & more
Data Mining
www.pentaho.com Download Pentaho's Open Source solution to Data Integration.
STATISTICA - Data Mining
www.StatSoft.com Learn why data mining works... Free Videos, Webcasts, Whitepapers
Data mining - Wikipedia, the free encyclopedia
Data mining is the process of extracting hidden patterns from large amounts of data. As more data is gathered, with the amount of data doubling every three ...
en.wikipedia.org/wiki/Data_mining - 94k - Cached - Similar pages
Data Mining Software
Powerful development environment. Download free evaluation.
www.peltarion.com
Test & Learn
Optimize your testing ROI
Make testing your core advantage
www.predictivetechnologies.com
Data Mining Tool
Automatic collection & integration of content from any web site.
www.kapowtech.com
Searches related to: data mining
data warehouse
data mining articles
data mining companies
data mining course
data mining and privacy
text mining
data modeling
olap

Challenges

- Word mismatch: people often use different words to describe concepts in their queries than authors use to describe the same concepts in their documents.



Challenges

- Queries contain ambiguous and new terms
    - apple: "apple computer" or "apple pie"?
    - NDCG:?
- Users tend to submit short queries consisting of only one or two words
    - almost 20% one-word queries
    - almost 30% two-word queries

- Users may have little or even no knowledge about the topic they are searching for!

Classes of Suggestion Relevance

[Jones, 2006]

- Precise rewriting
  - The rewritten form of query matches user's intent
- Approximate rewriting
  - The rewritten form has a direct close relationship to the topic described by the initial query
- Possible rewriting
  - The rewritten form either has some categorical relationship to the initial query or describes a complementary product
- Clear mismatch
  - The rewritten form has no clear relationship to user's intent

Example Queries and Query-suggestion

Class	Score	Examples		
Precise rewriting	1	automotive insurance	↦	automobile insurance
		corvette car	↦	chevrolet corvette
		apple music player	↦	apple ipod
		apple music player	↦	ipod
		cat cancer	↦	feline cancer
		help with math homework	↦	math homework help
Approximate rewriting	2	apple music player	↦	ipod shuffle
		personal computer	↦	compaq computer
		hybrid car	↦	toyota prius
		aeron chair	↦	office furniture
Possible rewriting	3	onkyo speaker system	↦	yamaha speaker system
		eye-glasses	↦	contact lenses
		orlando bloom	↦	johnny depp
		cow	↦	pig
		ibm thinkpad	↦	laptop bag
Clear mismatch	4	jaguar xj6	↦	os x jaguar
		time magazine	↦	time and date magazine

Typical Query Suggestion

[Jinxi Xu, 1996]

- Global analysis
  - Selects expansion terms on the basis of the information on the whole document set
  - Relatively robust
  - Expensive in terms of disk space and computer time
- Local analysis
  - Formulate expansion terms based on top-ranked results
  - Relatively efficient
  - Perform badly for queries with few relevant documents

Query Expansion by Mining Query Log

[Hang Cui, 2003]

- TF-iDF
  - Each document is represented as a document vector $\{W_1^{(d)}, W_2^{(d)}, \ldots W_N^{(d)}\}$, where $W_i^{(d)}$ is the weight of the *i*th item in a document, defined as

$$W_i^{(d)} = \frac{\ln(1 + tf_i^{(d)}) \times idf_i^{(d)}}{\sqrt{\sum \ln^2(1 + tf_i^{(d)}) \times \sum (idf_i^{(d)})^2}},$$

$$idf_i^{(d)} = \ln \frac{N}{n_i},$$

  - Similarity between query terms and document terms

$$Similarity = \frac{\sum_{i=1}^{N} W_i^{(q)} W_i^{(d)}}{\sqrt{\sum_{i=1}^{N} (W_i^{(q)})^2} \sqrt{\sum_{i=1}^{N} (W_i^{(d)})^2}}.$$

Query Suggestion Using Clickthrough Data

- Query logs recorded by search engines

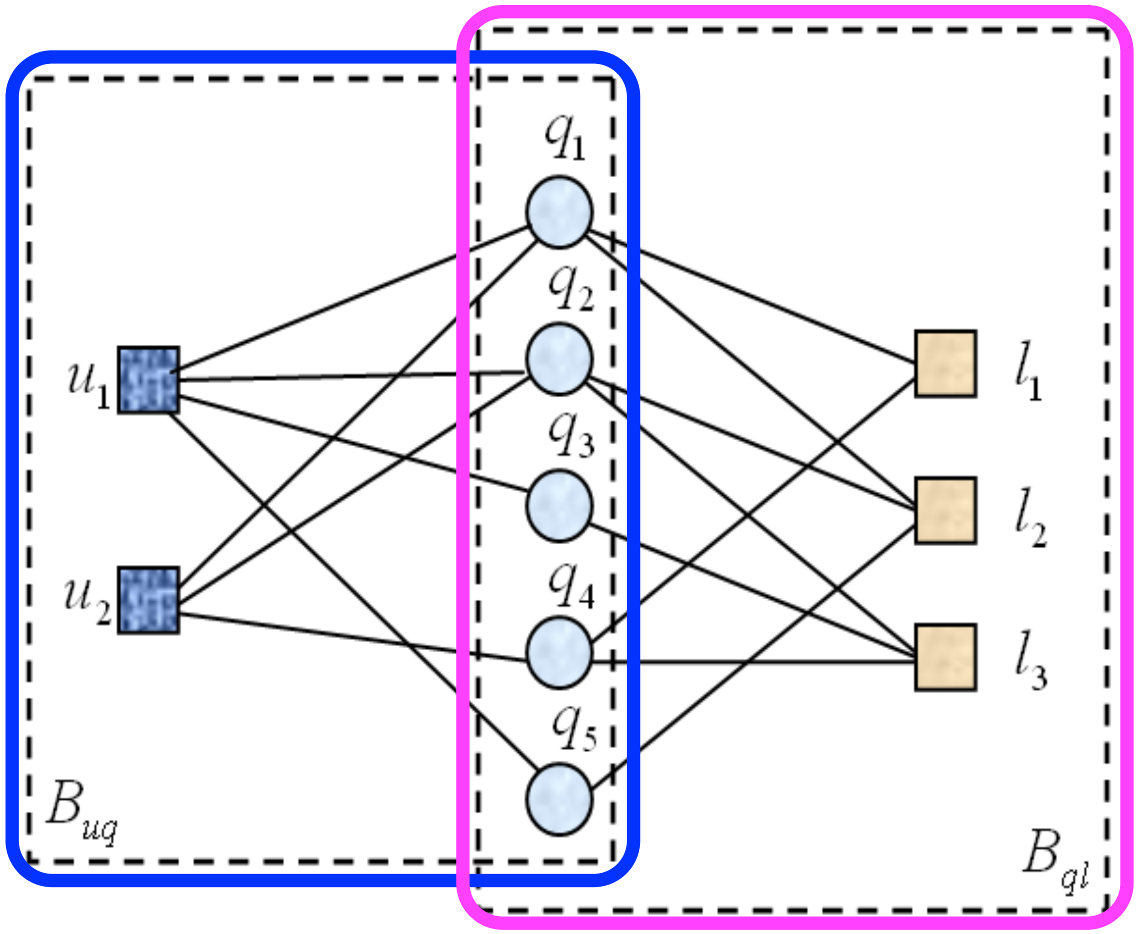
$$\langle u, q, l, r, t \rangle$$

Table 1: Samples of search engine clickthrough data

ID	Query	URL	Rank	Time
358	facebook	http://www.facebook.com	1	2008-01-01 07:17:12
358	facebook	http://en.wikipedia.org/wiki/Facebook	3	2008-01-01 07:19:18
3968	apple iphone	http://www.apple.com/iphone/	1	2008-01-01 07:20:36
...	...	...	...	...

- Users' relevance feedback to indicate desired/preferred/ target results

Joint Bipartite Graph



$B_{uq} = (V_{uq}, E_{uq})$
$V_{uq} = U \cup Q$
$U = \{u_1, u_2, ..., u_m\}$
$Q = \{q_1, q_2, ..., q_n\}$
$E_{uq} = \{(u_i, q_j)|$ there is an edge from u_i to $q_j\}$
is the set of all edges.
The edge (u_i, q_j) exists in this bipartite graph if and only if a user u_i issued a query q_j.

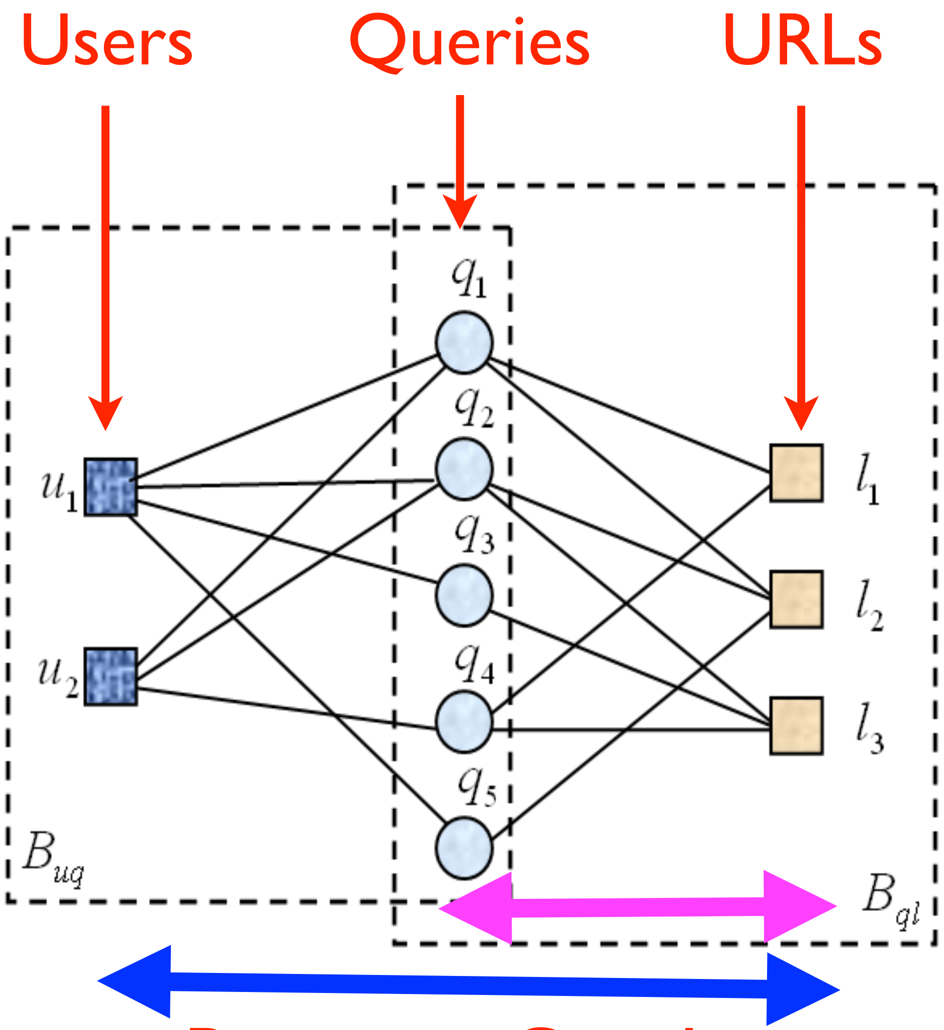
$B_{ql} = (V_{ql}, E_{ql})$
$V_{ql} = Q \cup L$
$Q = \{q_1, q_2, ..., q_n\}$
$L = \{l_1, l_2, ..., l_p\}$
$E_{ql} = \{(q_i, l_j)|$ there is an edge from q_i to $l_j\}$
is the set of all edges.
The edge (q_j, l_k) exists if and only if a user u_i clicked a URL l_k after issuing an query q_j.

Key Points

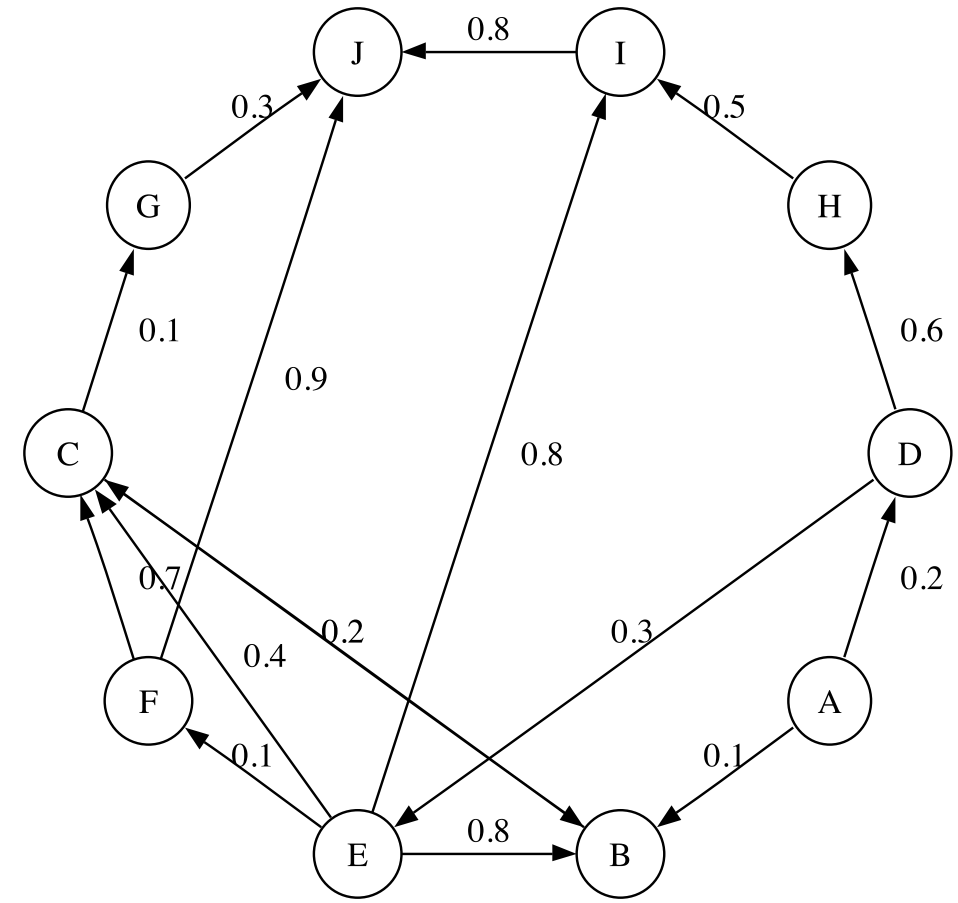
- Two-level latent semantic analysis
  - Consider the use of a joint user-query and query-URL bipartite graphs for query suggestion
  - Use matrix factorization for learning query features in constructing the Query Similarity Graph
  - Use heat diffusion for similarity propagation for query suggestions

Level 1

Level 2

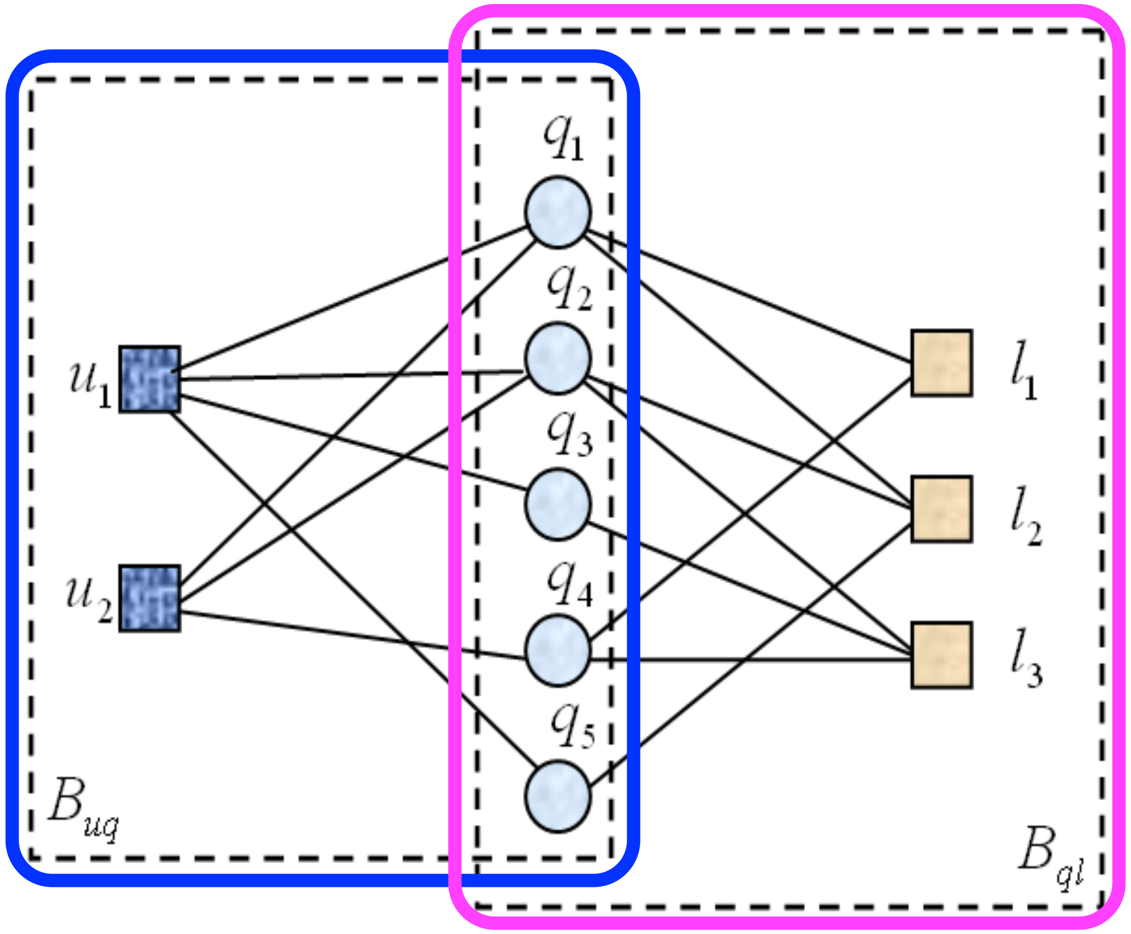


Bipartite Graphs



Query Similarity Graph

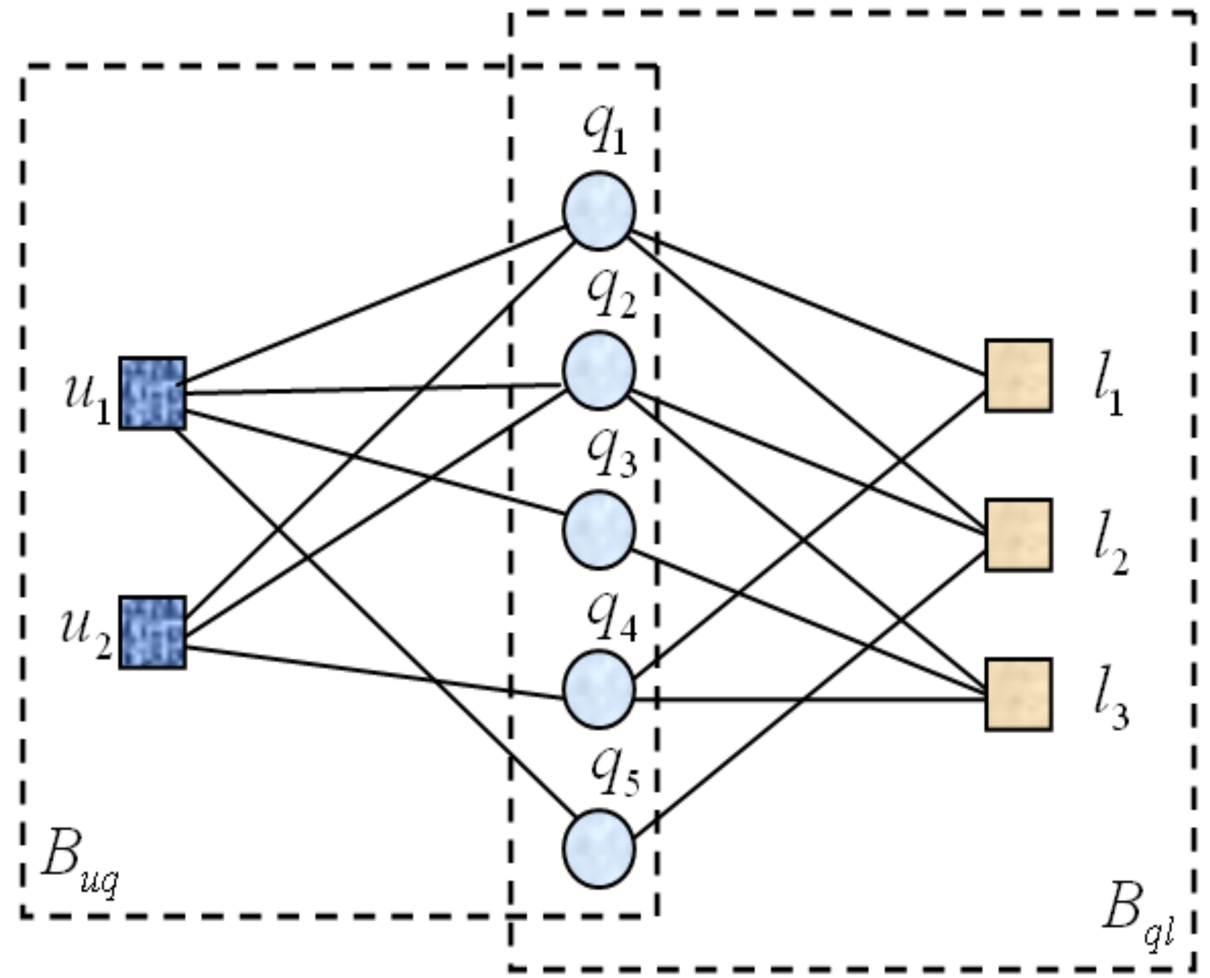
- Queries are issued by the users, and which URLs to click are also decided by the users
- Two distinct users are similar if they issued similar queries
- Two queries are similar if they are issued by similar users



r^*_{ij}	Normalized weight, how many times u_i issued q_j
s^*_{jk}	Normalized weight, how many times q_j is linked to l_k
U_i	L-dimensional vector of user u_i
Q_j	L-dimensional vector of query q_j
L_k	L-dimensional vector of URL l_k

$$\begin{aligned}\mathcal{H}(R,U,Q) &= \min_{U,Q} \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I^R_{ij}(r^*_{ij} - g(U_i^T Q_j))^2 \\ &+ \frac{\alpha_u}{2}\|U\|_F^2 + \frac{\alpha_q}{2}\|Q\|_F^2\end{aligned}$$

$$\begin{aligned}\mathcal{H}(S,Q,L) &= \min_{Q,L} \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{p} I^S_{jk}(s^*_{jk} - g(Q_j^T L_k))^2 \\ &+ \frac{\alpha_q}{2}\|Q\|_F^2 + \frac{\alpha_l}{2}\|L\|_F^2\end{aligned}$$



$$\mathcal{H}(S,R,U,Q,L) = \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{p} I_{jk}^{S}(s_{jk}^{*} - g(Q_j^T L_k))^2 + \frac{\alpha_r}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}^{R}(r_{ij}^{*} - g(U_i^T Q_j))^2 + \frac{\alpha_u}{2}\|U\|_F^2 + \frac{\alpha_q}{2}\|Q\|_F^2 + \frac{\alpha_l}{2}\|L\|_F^2,$$

- A local minimum can be found by performing gradient descent in U_i, Q_j and L_k

Gradient Descent Equations

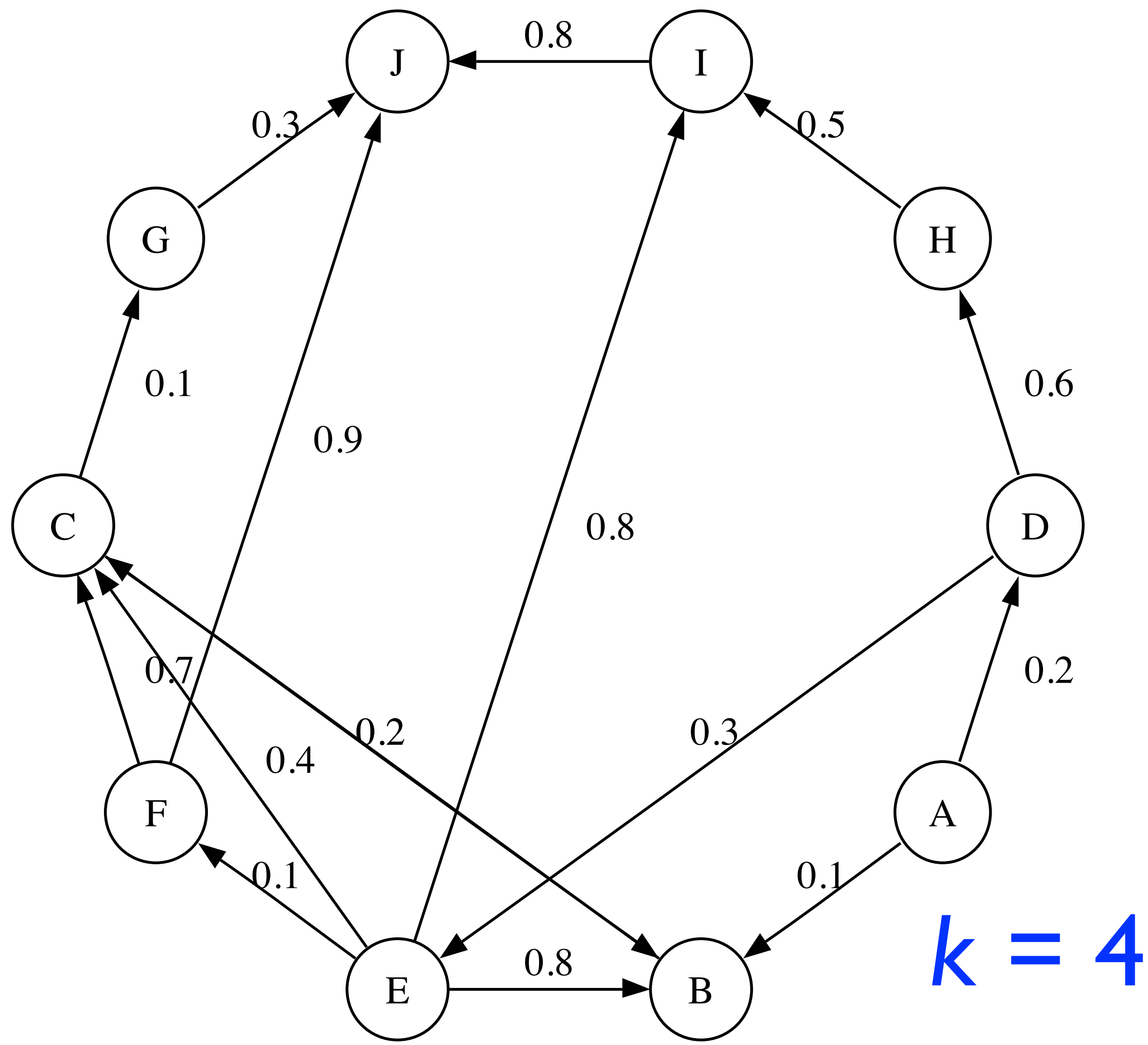
$$\frac{\partial \mathcal{H}}{\partial U_i} = \alpha_r \sum_{j=1}^{n} I_{ij}^R g'(U_i^T Q_j)(g(U_i^T Q_j) - r_{ij}^*) Q_j + \alpha_u U_i,$$

$$\boxed{\begin{aligned}\frac{\partial \mathcal{H}}{\partial Q_j} &= \sum_{k=1}^{p} I_{jk}^S g'(Q_j^T L_k)(g(Q_j^T L_k) - s_{jk}^*) L_k \\ &+ \alpha_r \sum_{i=1}^{m} I_{ij}^R g'(U_i^T Q_j)(g(U_i^T Q_j) - r_{ij}^*) U_i + \alpha_q Q_j,\end{aligned}}$$

$$\frac{\partial \mathcal{H}}{\partial L_k} = \sum_{j=1}^{n} I_{jk}^S g'(Q_j^T L_k)(g(Q_j^T L_k) - s_{jk}^*) Q_j + \alpha_l L_k,$$

Only the Q matrix, the queries' latent features,
is being used to generate the query similarity graph!

Query Similarity Graph



- Similarities are calculated using queries' latent features
- Only the top-k similar neighbors (terms) are kept

Similarity Propagation

- Based on the Heat Diffusion Model
- In the query graph, given the heat sources and the initial heat values, start the heat diffusion process and perform *P* steps
- Return the Top-*N* queries in terms of highest heat values for query suggestions

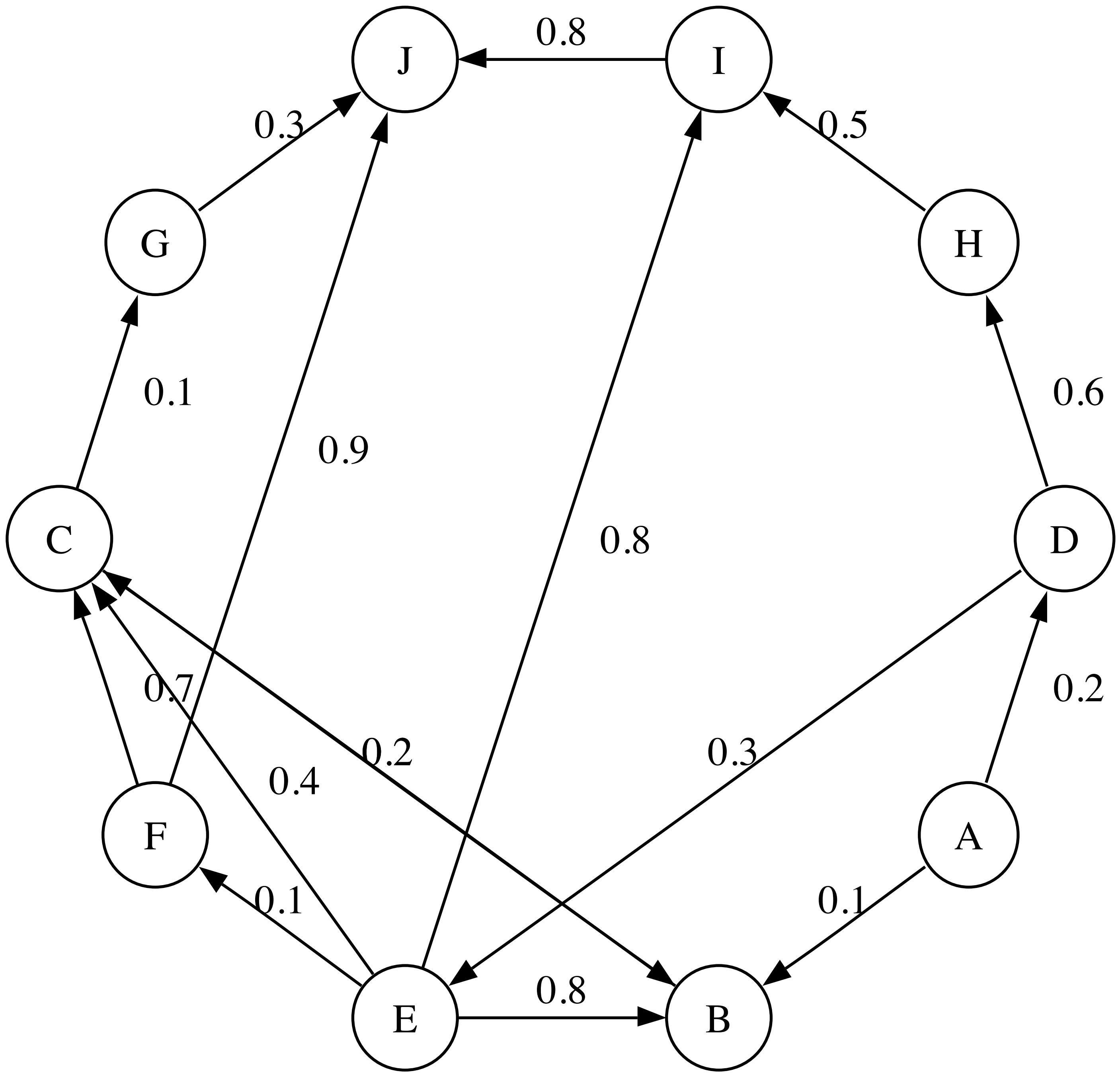
Heat Diffusion Model

- Heat diffusion is a physical phenomena
- Heat flows from high temperature to low temperature in a medium
- Heat kernel is used to describe the amount of heat that one point receives from another point
- The way that heat diffuse varies when the underlying geometry varies

$$\boxed{\rho C_P \frac{\partial T}{\partial t} = Q + \nabla \cdot (k \nabla T)}$$

ρ	Density
C_P	Heat capacity and constant pressure
$\frac{\partial T}{\partial t}$	Change in temperature over time
Q	Heat added
k	Thermal conductivity
∇T	Temperature gradient
$\nabla \cdot \mathbf{v}$	Divergence

Heat Diffusion Process



Similarity Propagation Model

$$\frac{f_i(t+\Delta t)-f_i(t)}{\Delta t} = \alpha\left(-\frac{\tau_i}{d_i} f_i(t) \sum_{k:(q_i,q_k)\in E} w_{ik} + \sum_{j:(q_j,q_i)\in E} \frac{w_{ji}}{d_j} f_j(t)\right) \quad (1)$$

$$\mathbf{f}(1) = e^{\alpha \mathbf{H}} \mathbf{f}(0) \quad (2)$$

$$H_{ij} = \begin{cases} w_{ji}/d_j, & (q_j, q_i) \in E, \\ -(\tau_i/d_i) \sum_{k:(i,k)\in E} w_{ik}, & i = j, \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

$$\mathbf{f}(1) = e^{\alpha \mathbf{R}} \mathbf{f}(0), \quad \boxed{\mathbf{R} = \gamma \mathbf{H} + (1-\gamma)\mathbf{g}\mathbf{1}^T} \quad (4)$$

α	Thermal conductivity
d_i	Heat value of node i at time t
$\mathrm{f}_i(t)$	Heat value of node i at time t
w_{ik}	Weight between node i and node k
$\mathbf{f}(0)$	Vector of the initial heat distribution
$\mathbf{f}(1)$	Vector of the heat distribution distribution at time 1
τ_i	Equal to 1 if node i has outlinks, else equal to 0
γ	Random jump parameter, and set to 0.85
$\mathbf{g}$	Uniform stochastic distribution vector

Discrete Approximation

- Compute $e^{\alpha \mathbf{R}}$ is time consuming
- We use the discrete approximation to substitute

$$\mathbf{f}(1) = \left(\mathbf{I} + \frac{\alpha}{P}\mathbf{R}\right)^{P} \mathbf{f}(0)$$

- For every heat source, only diffuse heat to its neighbors within *P* steps
- In our experiments, *P* = 3 already generates fairly good results

Query Suggestion Procedure

- For a given query *q*

1. Select a set of *n* queries, each of which contains at least one word in common with *q*, as heat sources
2. Calculate the initial heat values by

$$f_{\hat{q}_i}(0) = \frac{|\mathcal{W}(q) \cap \mathcal{W}(\hat{q}_i)|}{|\mathcal{W}(q) \cup \mathcal{W}(\hat{q}_i)|}$$

q = "Sony"
"Sony" = 1
"Sony Electronics" = 1/2
"Sony Vaio Laptop" = 1/3

3. Use $\mathbf{f}(1) = e^{\alpha \mathbf{R}} \mathbf{f}(0)$ to diffuse the heat in graph
4. Obtain the Top-*N* queries from $\mathbf{f}(1)$

Physical Meaning of α

- If set α to a large value
  - The results depend more on the query graph, and more semantically related to original queries, e.g., travel => lowest air fare
- If set α to a small value
  - The results depend more on the initial heat distributions, and more literally similar to original queries, e.g., travel => travel insurance

Experimental Dataset

Data Source	Clickthrough data from AOL search	After Pre-Processing
Collection Period	March 2006 to May 2006 (3 months)	
Lines of Logs	19,442,629	
Unique user IDS	657,426	192,371
Unique queries	4,802,520	224,165
Unique URLs	1,606,326	343,302
Unique words		69,937

Pre-processing

- Computer set-up
  Intel Pentium D CPU, 3.0 Gz, Dual Core with 1G memory
- Keep valid words which contains only ‘a’, ‘b’,…, ‘z’ and spaces
- Remove those queries which appear less than three times

Query Suggestions

Table 2: Examples of LSQS Query Suggestion Results ($k = 50$)

Testing Queries	Suggestions				
	$\alpha = 10$			$\alpha = 1000$	
	Top 1	Top 2	Top 3	Top 4	Top 5
michael jordan	michael jordan shoes	michael jordan bio	pictures of michael jordan	nba playoff	nba standings
travel	travel insurance	abc travel	travel companions	hotel tickets	lowest air fare
java	sun java	java script	java search	sun microsystems inc	virtual machine
global services	ibm global services	global technical services	staffing services	temporary agency	manpower professional
walt disney land	world of disney	disney world orlando	disney world theme park	disneyland grand hotel	disneyland in california
intel	intel vs amd	amd vs intel	pentium d	pentium	centrino
job hunt	jobs in maryland	monster job	jobs in mississippi	work from home online	monster board
photography	photography classes	portrait photography	wedding photography	adobe elements	canon lens
internet explorer	ms internet explorer	internet explorer repair	internet explorer upgrade	microsoft com	security update
fitness	fitness magazine	lifestyles family fitness	fitness connection	womens health magazine	family fitness
m schumacher	schumacher	red bull racing	formula one racing	ferrari cars	formula one
solar system	solar system project	solar system facts	solar system planets	planet jupiter	mars facts
sunglasses	replica sunglasses	cheap sunglasses	discount sunglasses	safilo	marhon
search engine	audio search engine	best search engine	search engine optimization	song lyrics search	search by google
disease	grovers disease	liver disease	morgellons disease	colic in babies	oklahoma vital records
pizzahut	pizza hut menu	pizza coupons	pizza hut coupons	papa johns pizza coupon	papa johns
health care	health care proxy	universal health care	free health care	great west healthcare	uhc
flower delivery	global flower delivery	online florist	flowers online	send flowers	virtual flower
wedding	wedding guide	wedding reception ideas	wedding decoration	unity candle	centerpiece ideas
astronomy	astronomy magazine	astronomy pic of the day	star charts	space pictures	comet

References

- S. Cucerzan and R. W. White. Query suggestion based on user landing pages. In SIGIR, pages 875–876, 2007.
- H. Cui, J.-R. Wen, J.-Y. Nie, and W.-Y. Ma. Query expansion by mining user logs. IEEE Trans. Knowl. Data Eng., 15(4):829–839, 2003.
- W. Gao, C. Niu, J.-Y. Nie, M. Zhou, J. Hu, K.-F. Wong, and H.-W. Hon. Cross-lingual query suggestion using query logs of different languages. In SIGIR, pages 463–470, 2007.
- R. Jones, B. Rey, O. Madani, and W. Greiner. Generating query substitutions. In L. Carr, D. D. Roure, A. Iyengar, C. A. Goble, and M. Dahlin, editors, WWW, pages 387–396. ACM, 2006.
- H. Ma, H. Yang, I. King, and M. R. Lyu. Learning latent semantic relations from clickthrough data for query suggestion. In CIKM, pages 709–718, 2008.
- Q. Mei, D. Zhou, and K. W. Church. Query suggestion using hitting time. In CIKM, pages 469–478, 2008.
- J. Xu and W. B. Croft. Query expansion using local and global document analysis. In SIGIR, pages 4–11, 1996.

Learning To Rank

Irwin King
Department of Computer Science and Engineering
The Chinese University of Hong Kong
http://wiki.cse.cuhk.edu.hk/irwin.king/home

Learning to Rank

- Booming Search Industry

Learning to Rank

- Given query q and set of docs $d_1, ...d_n$
    - Find documents relevant to q
    - Typically expressed as a ranking on $d_1, ...d_n$
    - Are social signals important?

Widely-used Judgement

- Pointwise
  - Binary judgment (Relevant vs. Irrelevant)
  - Multi-valued discrete (Perfect > Excellent > Good > Fair > Bad)
- Pairwise
  - Pairwise preference
    - Document A is more relevant than document B w.r.t. query q
- Listwise
  - Partial or total orders
  - Could be mined from click-through logs

Conventional Ranking Models

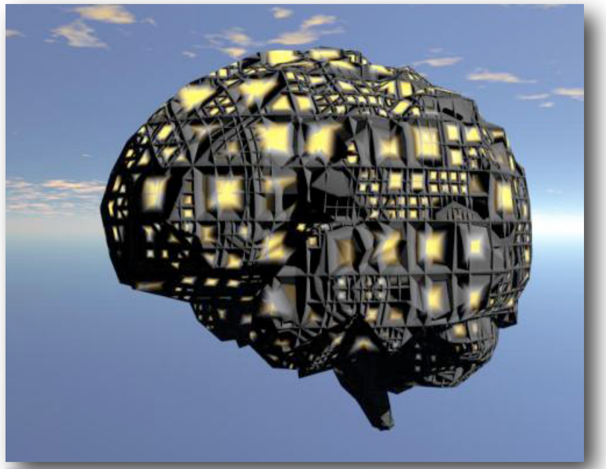
- Content relevance
  - Boolean model, extended Boolean model, etc.
  - Vector space model, latent semantic indexing (LSI), etc.
  - BM25 model, statistical language model, etc.
  - Span based model, distance aggregation model, etc.
- Page Quality
  - Link analysis: HITS, PageRank, TrustRank, etc.
  - Log mining: DirectHITS, BrowseRank, etc

Discussion on Conventional Models

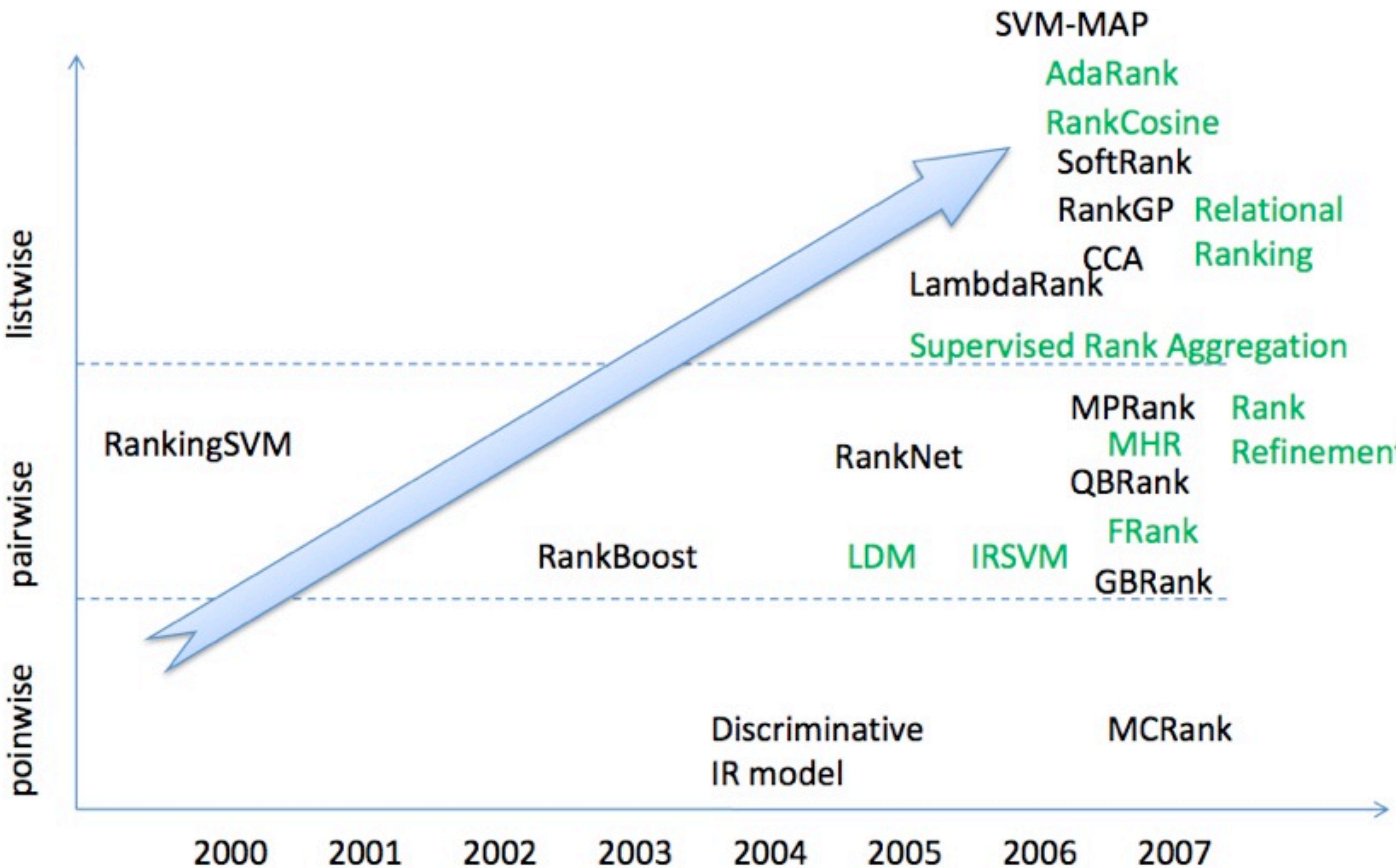
- For a particular model
  - Manual parameter tuning is usually difficult, especially when there are many parameters.
- For comparison between two models
  - Given a test set, it is difficult / unfair to compare two models if one is over-tuned while the other is not.
- For a collection of models
  - There are hundreds of models proposed in the literature.
  - It is non-trivial to combine them to produce a even more effective model

Machine Learning Can Help

- Machine learning is an effective tool
  - To automatically tune parameters
  - To combine multiple evidences
  - To avoid over-fitting (by means of regularization, etc.)
- Learning to Rank
  - Use machine learning technologies to train the ranking model
  - A hot research topic these years



Learning To Rank Techniques



http://research.microsoft.com/en-us/people/tyliu/default.aspx

Resources

- LETOR benchmark: a package of benchmark data sets for learning to rank, released by Microsoft Research Asia.
- Current LETOR baselines
  - Ranking SVM
  - RankBoost
  - AdaRank
  - Multiple hyperline ranker
  - FRank
  - ListNet

Define Metric

A metric on a set X is a function (called the distance function or simply distance)

$$d : X \times X \rightarrow \mathcal{R} \tag{1}$$

where $\mathcal{R}$ is the set of real numbers. For all $x, y, z \in X$, this function is required to satisfy the following conditions:

1. $d(x, y) \geq 0$ (non-negativity)
2. $d(x, y) = 0$ if and only if $x = y$ (identity of indiscernible)
3. $d(x, y) = d(y, x)$ (symmetry)
4. $d(x, z) \leq d(x, y) + d(y, z)$ (subadditivity or triangle inequality)

Define Ranking

A ranking is a relationship between a set of items. Weak order or total preorder.

A total order is a binary relation on some set X. The relation is transitive, antisymmetric, and total. If X is totally order under $\leq$, then the following statemetns hold for all a, b, and c in X:

- If $a \leq b$ and $b \leq a$ then $a = b$ (antisymmetry);
- If $a \leq b$ and $b \leq c$ then $a \leq c$ (transitivity);
- $a \leq b$ or $b \leq a$ (totality).

IR Evaluation

- Objective
  - Evaluate the effectiveness of a ranking model
- A standard test set
  - Contain a large number of (randomly sampled) queries, their associated documents, and the labels (relevance judgments) of these documents.
- A measure
  - Evaluate the effectiveness of a ranking model for a particular query.
  - Average the measure over the entire test set to represent the expected effectiveness of the model.

Ranking Evaluation

- Binary judgment
  - Relevant vs. Irrelevant
- Multi-level ratings
  - Excellent > Good > Fair > Poor
- Pairwise preferences
  - Document *A* is more relevant than document *B* with respect to query *q*

Measures

- Precision--measure of exactness
- Recall--measure of completeness
- They are usually linked closely together
- Often, there is an inverse relationship between Precision and Recall
- Increasing one at the cost of reducing the other, e.g., increase its Recall by retrieving more documents, at the cost of increasing number of irrelevant documents retrieved (decreasing Precision)

Confusion Matrix

- True positives
- True negatives
- False positives
- False negatives

		actual value		
		p	***n***	**total**
prediction outcome	***p'***	True Positive	False Positive	P'
	n'	False Negative	True Negative	N'
	total	P	N	

In Classification

- Precision–the number of true positives divided by the total number of elements labeled as belonging to the positive class

$$\text{Precision} = \frac{tp}{tp + fp} \tag{1}$$

It can also be interpreted as the probability that a (randomly selected) retrieved document is relevant.

- Recall–the number of true positives divided by the total number of elements that actually belong to the positive class.

$$\text{Recall} = \frac{tp}{tp + fn} \tag{2}$$

Recall in this context is also referred to as the True Positive Rate. It can also be interpreted as the probability that a (randomly selected) relevant document is retrieved in a search.

In Classification

- True Negative Rate

$$\text{True Negative Rate} = \frac{tn}{tn + fp} \tag{1}$$

- Accuracy

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn} \tag{2}$$

In Information Retrieval

- Precision
  - In classification, precision for a class is the number of true positives divided by the total number of elements labeled as belonging to the positive class
  -

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{ retrieved documents}\}|}{|\{\text{retrieved documents}\}|} \quad (1)$$

  - Precision takes all retrieved documents into account
  - Precision can also be evaluated at a given cut-off-rank. This is called precision at n or P@n.
- Recall
  - Recall is the fraction of the documents that are relevant to the query that are successfully retrieved.

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{ retrieved documents}\}|}{|\{\text{relevant documents}\}|} \quad (2)$$

Fall-Out

- Fall-Out–the proportion of non-relevant documents that are retrieved, out of all non-relevant documents available:

$$\text{Fall-Out} = \frac{|\{\text{non-relevant documents}\} \cap \{\text{ retrieved documents}\}|}{|\{\text{non-relevant documents}\}|} \tag{1}$$

F-Measure

- F-Measure–Weighted harmonic mean of precision and recall.

$$F = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1}$$

This is also known as the F_1 measure since recall and precision are evenly weighted.

For the general F_β measure (for non-negative real values of β):

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}} \tag{2}$$

The F_2 measure weights recall twice as much as precision, and the $F_{0.5}$ measure weights precision twice as much as recall.

Average Precision and Recall

- Average Precision of Precision and Recall–it emphasizes returning more relevant earlier. It is average of precisions computed after truncating the list after each of the relevant documents in turn:

$$\text{AP} = \frac{\sum_{r=1}^{N} (\text{P@}r \times \text{rel}(r))}{\text{number of relevant documents}} \tag{1}$$

where r is the rank, N the number retrieved, rel() a binary function on the relevance of a given rank, and P@r precision at a given cut-off rank, r.

Example

Given the list of seven retrieved documents as, $\{r_1, nr_2, nr_3, r_4, r_5, nr_6, r_7\}$ where r_i are relevant documents and nr_j are non-relevant documents. The Average Precision is then

$$\mathrm{AP} = \frac{1}{4} \cdot \left(\frac{1}{1} + \frac{2}{4} + \frac{3}{5} + \frac{4}{7} \right) \approx 0.67 \tag{1}$$

Evaluation Measures

- MAP (Mean Average Precision)–averaged AP over all queries in the test set
- NDCG (Normalized Discounted Cumulative Gain)
- MRR (Mean Reciprocal Rank)
  - For query q_i, rank position of the first relevant document: r_i
  - MRR: average of $1/R_i$ over all queries
- WTA (Winner-Take-All)
  - If top ranked document is relevant: 1; otherwise 0
  - Average over all queries

Discounted Cumulative Gain

DCG is a measure of effectiveness of a Web search engine algorithm or related applications, often used in information retrieval. DCG measures the usefulness, or gain, of a document based on its position in the result list. The gain is accumulated cumulatively from the top of the result list to the bottom with the gain of each result discounted as lower ranks.

- Assumptions
  - Highly relevant documents are more useful when appearing earlier in a search engine result list (have higher ranks)
    - Highly relevant documents are more useful than marginally relevant documents, which are in turn more useful than irrelevant documents.

Cumulative Gain

Cumulative Gain (CG) is the predecessor of DCG and does not include the position of a result in the consideration of the usefulness of a result set. It is the sum of the graded relevance values of all results in a search result list. The CG at a particular rank position p is

$$\mathrm{CG}_p = \sum_{i=1}^{p} rel_i \tag{1}$$

where rel_i is the graded relevance of the result at position i.

The value computed with the CG function is unaffected by changes in the ordering of search results, i.e., moving a highly relevant document d_i above a higher ranked, less relevant, document d_j does not change the computed value for CG.

Discounted Cumulative Gain

Discounted Cumulative Gain (DCG) The premise of DCG is that highly relevant documents appearing lower in a search result list should be penalized as the graded relevance value is reduced logarithmically proportional to the position of the result. The discounted CG accumulated at a particular rank position p is defined as

$$\mathrm{DCG}_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i} \tag{1}$$

The logarithmic reduction factor has not shown any theoretical justification. An alternative formulation of DCG places much stronger emphasis on retrieving relevant documents sooner using a power distribution and is formulated as

$$\mathrm{DCG}_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2(1+i)} \tag{2}$$

The function is equivalent to the previous DCG function when the relevance values of documents are binary, i.e., $rel_i \in \{0, 1\}$.

The summation $\sum_{i=1}^{p}$ is cumulating, the term $2^{rel_i} - 1$ is the gain, and the term $\log_2(1+i)$ is the position discount.

Normalizing DCG

Search result lists vary in length depending on the query. Comparing a search engine's performance from one query to the next cannot be consistently achieved using DCG alone, so the cumulative gain at each position for a chosen value of p should be normalized across queries. This is done by sorting documents of a result list by relevance, producing an ideal DCG (IDCG) at position p. For a query, the normalized discounted cumulative gain, or nDCG, is computed as:

$$\mathrm{nDCG}_p = \frac{\mathrm{DCG}_p}{\mathrm{IDCG}_p} \tag{1}$$

Note that in a perfect ranking algorithm, the DCG_p will be the same as the IDCG_p producing an nDCG of 1.0.

Example

Presented with a list of documents in response to a search query, an experiment participant is asked to judge the relevance of each document to the query. Each document is to be judged on a scale of 0-3 with 0 meaning irrelevant, 3 meaning completely relevant, and 1 and 2 meaning "somewhere in between". For the documents ordered by the ranking algorithm as

$$D_1, D_2, D_3, D_4, D_5, D_6$$

the user provides the following relevance scores:

$$\mathrm{CG}_p = \sum_{i=1}^{p} rel_i = 3 + 2 + 3 + 0 + 1 + 2 = 11$$

Changing the order of any two documents does not affect the CG measure.

Example

DCG is calculated as follows:

i	rel_i	$\log_i$	$\frac{rel_i}{\log_2 i}$
1	3	N/A	N/A
2	2	1	2
3	3	1.59	1.887
4	0	2.0	0
5	1	2.32	0.431
6	2	2.59	0.772

Now a switch of D_3 and D_4 results in a reduced DCG so a more relevant document is discounted more by being placed in a lower rank.

Example

To normalize DCG values, an ideal ordering for the given query is needed. For this example, that ordering would be the monotonically decreasing sort of the relevance judgments provided by the experiment participant, which is:

$$3, 3, 2, 2, 1, 0$$

The DCG of this ideal ordering, or IDCG, is then:

$$\mathrm{IDCG}_6 = \frac{\mathrm{DCG}_6}{\mathrm{IDCG}_6} = \frac{8.09}{8.693} = 0.9306$$

so the DCG_6 of this ranking is

$$\mathrm{DCG}_6 = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i} = 3 + (2 + 1.887 + 0 + 0.431 + 0.772) = 8.09$$

Properties of Ranking in IR

- Loss function should be defined on ranked list w.r.t. a query
- Relative order is important
- Position sensitive
- Rank based evaluation

Categorization

- Pointwise
  - Input: single documents
  - Output: scores or class labels
  - Discriminative model for IR, McRank, ...
- Pairwise
  - Input: document pairs
  - Output: partial order preference
  - Ranking SVM, RankBoost, RankNet, FRank, ...
- Listwise
  - Input: document collections
  - Output: ranked document list
  - LambdaRank, AdaRank, SVM-MAP, RankCosine,...

Pointwise Approach

- Reduce ranking to regression or classification on single documents
- Discriminative Model
  - Treat relevant documents as positive examples, while irrelevant documents as negative examples
  - Learning algorithms
    - Maximum Entropy
    - Support Vector Machines

Document Features

$\sum_{q_i \in Q \cap D} \log(c(q_i, D))$	$\sum_{q_i \in Q \cap D} (\log(\frac{\|C\|}{c(q_i, C)}))$
$\sum_{i=1}^{n} \log(1 + \frac{c(q_i, D)}{\|D\|})$	$\sum_{i=1}^{n} \log(1 + \frac{c(q_i, D)}{\|D\|} idf(q_i))$
$\sum_{q_i \in Q \cap D} \log(idf(q_i))$	$\sum_{i=1}^{n} \log(1 + \frac{c(q_i, D)}{\|D\|} \frac{\|C\|}{c(q_i, C)})$

where $c(w, D)$ represents the raw count of word w in document D, C represents the collection, n is the number of terms in the query, $|\cdot|$ is the size-of function and $idf(\cdot)$ is the inverse document frequency.

- Vector space model (or term vector model) uses a vector of indexed words to represent a document.
- Each dimension corresponds to a separate term
- If a term (keyword, phrase, etc.) occurs in the document, its value in the vector is non-zero.
- The dimensionality of the vector is the number of words in the vocabulary.

Relevancy Ranking

Relevancy rankings of documents in a keyword search can be calculated, using the assumptions of document similarities theory, by comparing the deviation of angles between each document vector and the original query vector where the query is represented as same kind of vector as the documents. In practice, it is easier to calculate the cosine of the angle between the vectors instead of the angle:

$$\cos\theta = \frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{||\mathbf{v}_1|| ||\mathbf{v}_2||} \tag{1}$$

A cosine value of zero means that the query and document vector are orthogonal and have no match (i.e. the query term do not exist in the document being considered). See cosine similarity for further information.

Term Frequency

The **term count** in the given document is simply the number of times a given term appears in that document. This count is usually normalized to prevent a bias towards longer documents (which may have a higher term count regardless of the actual importance of that term in the document) to give a measure of the importance of the term ti within the particular document d_j. Thus we have the **term frequency**, defined as follows.

$$\mathrm{tf}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{1}$$

where $n_{i,j}$ is the number of occurrences of the considered term (t_i) in document d_j, and the denominator is the sum of number of occurrences of all terms in document d_j.

Inverse Document Frequency

The **inverse document frequency** is a measure of the general importance of the term (obtained by dividing the number of all documents by the number of documents containing the term, and then taking the logarithm of that quotient).

$$\mathrm{idf}_i = \log \frac{|D|}{|\{d : t_i \in d\}|} \tag{1}$$

with

- $|D|$: total number of documents in the corpus
- $|\{d : t_i \in d\}|$: number of documents where the term t_i appears (that is $n_{i,j} \neq 0$). If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to use $1 + |\{d : t_i \in d\}|$ Then

$$\text{tf-idf}_{i,j} = \mathrm{tf}_{i,j} \times \mathrm{idf}_i \tag{2}$$

A high weight in tf–idf is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms. The tf-idf value for a term will always be greater than or equal to zero.

Maximum Entropy (ME) Model

- Principle of Maximum Entropy is to model all that is known and assume nothing about that which is unknown.
- Choose a model consistent with all facts, but otherwise as uniform as possible.

ME Probability function is defined as:

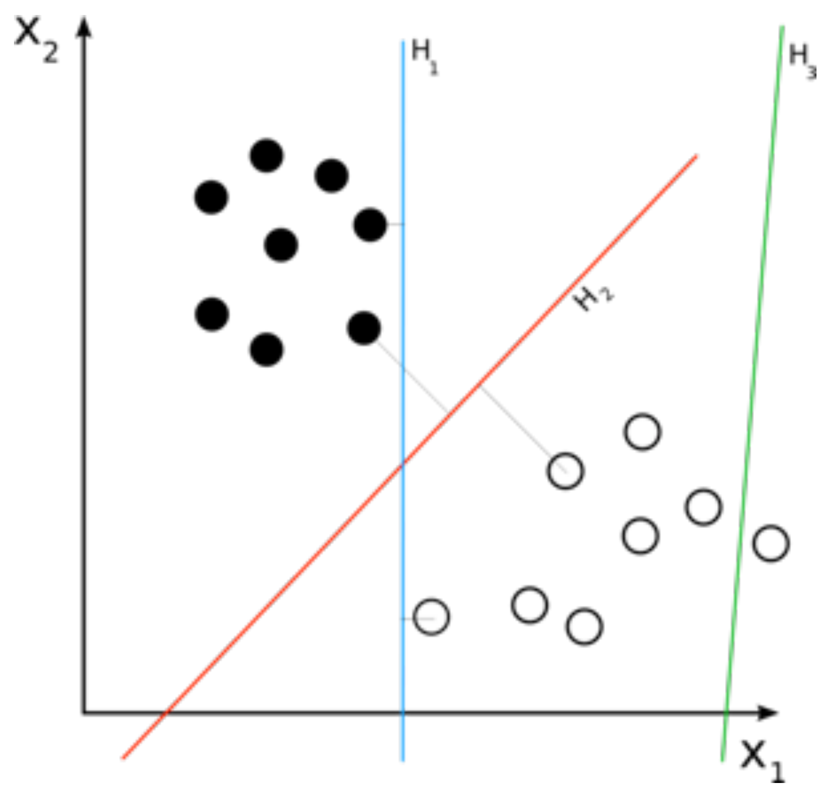
$$P(R|D,Q) = \frac{1}{Z(Q,D)} \exp\left(\sum_{i=1}^{n} \lambda_{i,R} f_i(D,Q)\right) \tag{1}$$

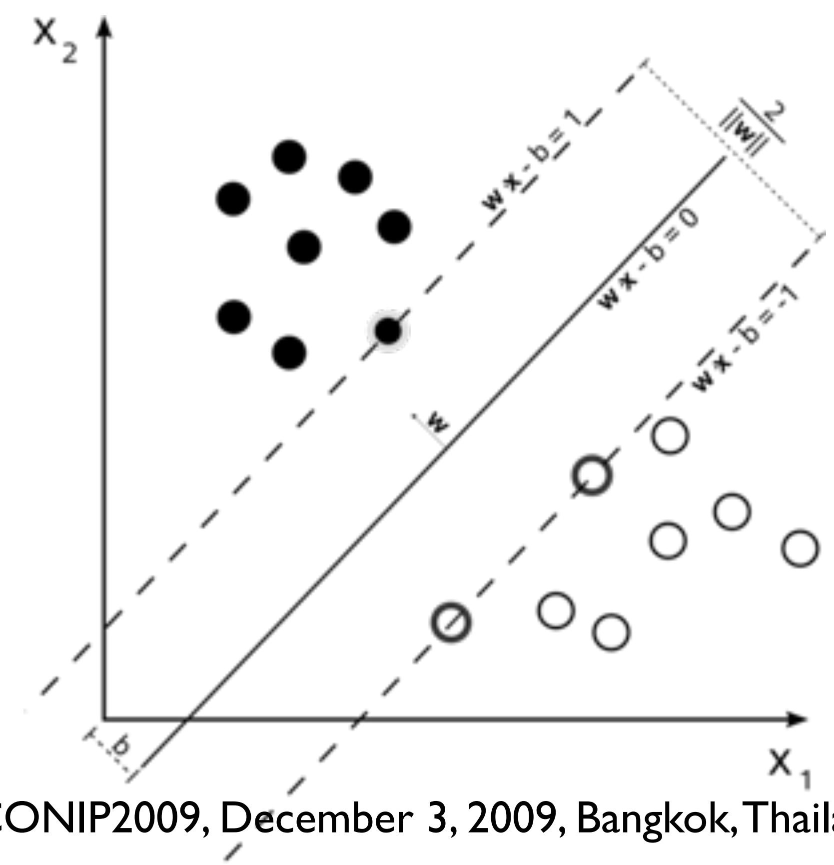
where $Z(Q,D)$ is a normalizing constant, $f_i(D,Q)$ are the feature functions of the document with weights $\lambda_{i,R}$ and n is the number of features. One can use the log-likelihood ratio as the scoring function:

$$\log \frac{P(R|D,Q)}{P(\bar{R}|D,Q)} = \sum_{i=1}^{n} (\lambda_{i,R} - \lambda_{i,\bar{R}}) f_i(D,Q) \tag{2}$$

Support Vector Machine

- A support vector machine constructs a hyperplane or set of hyperplanes in a high-dimensional space, which can be used for classification, regression or other tasks.
- A good separation is achieved by the hyperplane that has the largest distance to the nearest training datapoints of any class.





SVM Formalization

We are given some training data, a set of points of the form

$$\mathcal{D} = \{(\mathbf{x}_i, c_i) | \mathbf{x}_i \in \mathcal{R}^p, c_i \in \{-1, 1\}\}_{i=1}^n \tag{1}$$

where the c_i is either 1 or -1, indicating the class to which the point $\mathbf{x}_i$ belongs. Each $\mathbf{x}_i$ is a p-dimensional real vector. We want to find the maximum-margin hyperplane which divides the points having $c_i = 1$ from those having $c_i = -1$. Any hyperplane can be written as the set of points $\mathbf{x}$ satisfying

$$\mathbf{w} \cdot \mathbf{x} - b = 0, \tag{2}$$

where $\cdot$ denotes the dot product. The vector $\mathbf{w}$ is a normal vector: it is perpendicular to the hyperplane. The parameter $\frac{b}{||\mathbf{w}||}$ determines the offset of the hyperplane from the origin along the normal vector $\mathbf{w}$.

We want to choose the $\mathbf{w}$ and b to maximize the margin, or distance between the parallel hyperplanes that are as far apart as possible while still separating the data. These hyperplanes can be described by the equations

$$\mathbf{w} \cdot \mathbf{x} - b = 1, \tag{3}$$

and

$$\mathbf{w} \cdot \mathbf{x} - b = -1, \tag{4}$$

SVM Formalization

By using geometry, we find the distance between these two hyperplanes is $\frac{2}{||\mathbf{w}||}$, so we want to minimize $||\mathbf{w}||$. As we also have to prevent data points falling into the margin, we add the following constraint: for each i either

$$\mathbf{w} \cdot \mathbf{x} - b \geq 1 \text{ for } \mathbf{x}_i \tag{1}$$

of the first class or

$$\mathbf{w} \cdot \mathbf{x} - b \leq 1 \text{ for } \mathbf{x}_i \text{ of the second.} \tag{2}$$

This can be rewritten as:

$$c_i(\mathbf{w} \cdot \mathbf{x} - b) \geq 1 \text{ for all } 1 \leq i \leq n. \tag{3}$$

We can put this together to get the optimization problem:

$$\min_{\mathbf{w},b} \quad ||\mathbf{w}|| \tag{4}$$

$$\text{subject to} \quad c_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1 \text{ for any } i = 1, \ldots, n. \tag{5}$$

SVM

Thus if $\mathbf{f}(D, Q)$ is the vector of features, then the discriminant function is given by

$$g(R|D, Q) = \mathbf{w} \cdot \phi(\mathbf{f}(D, Q)) + b, \tag{1}$$

where

- $\mathbf{w}$ is the weight vector in kernel space that is learnt by the SVM from the training exmaples,
- $\cdot$ denotes inner product
- b is a constant
- ϕ is the mapping from input space to kernel space

The equation $g(R|D, Q) = 0$ represents the equation for the hyperplane in the kernel space.

The value of the discriminant function $g(R|D, Q)$ for an arbitrary document D and a query Q is proportional to the perpendicular distance of the document's augmented feature vector $\phi(\mathbf{f}(D, Q))$ from the separating hyper-plane in the kernel space.

Pairwise Approach

- No longer assume absolute relevance
- Reduce ranking to classification on document pairs w.r.t. the same query
- RankNet
  - Use Neural Network as model, and gradient descent as algorithm, to optimize the cross-entropy loss.
  - Evaluate on single documents: output a relevance score for each document w.r.t. a new query.

Ranking with Neural Nets

- Don't need to learn ordinal regression (mapping points to actual rank values); just need to map features to reals
- Train system on pairs (where first point is to be ranked higher or equal to second)
- However must evaluate on single points
- Use cross entropy cost => probabilistic model
- Use gradient descent

RankNet: Notes

- 5 human judged levels of relevance (“bad”, ... , “perfect”)
- A net with (number of features) inputs and one output
- Sort documents by the score that their feature vectors (which are computed from query + doc + other data)
- Compute NDCG on a set-aside validation set, keep the net that gives the best validation NDCG

RankNet Conclusions

- RankNet is simple to train
- RankNet is fast in test phase
- RankNet gives good results
- For pair-based probability costs (e.g., click rates!) RankNet is very well suited to the problem.
- However, the cost function used is not NDCG: the latter is optimized only indirectly, using a validation set.

Listwise Approach

- Instead of reducing ranking to regression or classification, perform learning directly on document list.
  - Directly optimize IR evaluation measure
    - AdaRank, SVM-MAP, SoftRank, LambdaRank, RankGP, ...
  - Define listwise loss functions
    - RankCosine, ListNet, ListMLE, ...

Privacy and Trust in Social Network

Irwin King
Department of Computer Science and Engineering
The Chinese University of Hong Kong
http://wiki.cse.cuhk.edu.hk/irwin.king/home

Privacy and Trust Tradeoff

- Privacy
- Need legal rights
- Reveal more data to trustworthy people

- Trust
- Provide access rights
- Gain trust through open sensitive data

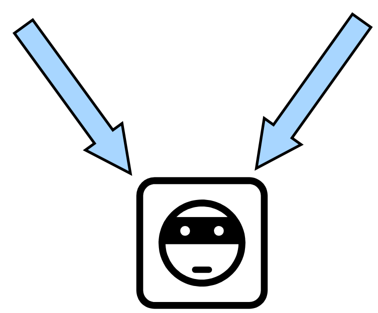
Motivation

Published table

Age	Zip.	Salary
17	12k	1000
19	13k	1010
20	14k	1020
24	16k	50000
29	21k	16000
34	24k	24000
39	36k	33000
45	39k	31000

Voter registration list

Name	Age	Zip.
Andy	17	12k
Bill	19	13k
Ken	20	14k
Jane	23	15k
Nash	24	16k
Joe	29	21k
Sam	34	24k
Linda	39	36k
Mary	45	39k



An adversary

Fact: 87% of Americans can be uniquely identified by {Zipcode, gender, date-of-birth}.

k-anonymity

[Sweeney, 2001]

	Age	Zip.	Salary
Andy	17	12k	1000
	19	13k	1010
	20	14k	1020
	24	16k	50000
	29	21k	16000
	34	24k	24000
	39	36k	33000
	45	39k	31000

(a) The microdata

Group ID	Age	Zip.	Salary
1	[17,24]	[12k,16k]	1000
1	[17,24]	[12k,16k]	1010
1	[17,24]	[12k,16k]	1020
1	[17,24]	[12k,16k]	50000
2	[29,34]	[21k,24k]	16000
2	[29,34]	[21k,24k]	24000
3	[39,45]	[36k,39k]	33000
3	[39,45]	[36k,39k]	31000

(b) Generalization

A group

Not sure about the salary of Andy now!

- *k*-anonymity
  - Divide tuples into groups
  - Each group has at least *k* tuples

Problem with *k*-anonymity

[Machanavajjhala, 2001]

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

Microdata

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	130**	< 30	*	Heart Disease
2	130**	< 30	*	Heart Disease
3	130**	< 30	*	Viral Infection
4	130**	< 30	*	Viral Infection
5	1485*	≥ 40	*	Cancer
6	1485*	≥ 40	*	Heart Disease
7	1485*	≥ 40	*	Viral Infection
8	1485*	≥ 40	*	Viral Infection
9	130**	3*	*	Cancer
10	130**	3*	*	Cancer
11	130**	3*	*	Cancer
12	130**	3*	*	Cancer

A 4-anonymous table

What about we know a person's Zip Code = 13053 and Age = 31?

In this case, we can conclude his/her disease is Cancer.

l-diversity

[Machanavajjhala, 2001]

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

Microdata

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	1305*	≤ 40	*	Heart Disease
4	1305*	≤ 40	*	Viral Infection
9	1305*	≤ 40	*	Cancer
10	1305*	≤ 40	*	Cancer
5	1485*	> 40	*	Cancer
6	1485*	> 40	*	Heart Disease
7	1485*	> 40	*	Viral Infection
8	1485*	> 40	*	Viral Infection
2	1306*	≤ 40	*	Heart Disease
3	1306*	≤ 40	*	Viral Infection
11	1306*	≤ 40	*	Cancer
12	1306*	≤ 40	*	Cancer

A 3-diverse table

- *l*-diversity
    - Divide tuples into groups
    - Each group has at least *l* different sensitive values

(*k*, e)-anonymity

[Zhang, 2007]

	ID	Quasi-identifiers			Sensitive
tuple ID	name	age	zipcode	gender	salary
1	Alex	35	27101	M	$54,000
2	Bob	38	27120	M	$55,000
3	Carl	40	27130	M	$56,000
4	Debra	41	27229	F	$65,000
5	Elain	43	27269	F	$75,000
6	Frank	47	27243	M	$70,000
7	Gary	52	27656	M	$80,000
8	Helen	53	27686	F	$75,000
9	Jason	58	27635	M	$85,000

Microdata

		Quasi-identifiers			Sensitive
group ID	tuple ID	age	zipcode	gender	salary
1	1	[31-40]	271*	*	$56,000
1	2	[31-40]	271*	*	$54,000
1	3	[31-40]	271*	*	$55,000
2	4	[41-50]	272*	*	$65,000
2	5	[41-50]	272*	*	$75,000
2	6	[41-50]	272*	*	$70,000
3	7	[51-60]	276*	*	$80,000
3	8	[51-60]	276*	*	$75,000
3	9	[51-60]	276*	*	$85,000

A 3-diverse table

Though the salary in group 1 is different, we are sure that Alex's salary is around 55,000

- (*k*, e)-anonymity
  - Each group has at least *k* tuples
  - Difference between the maximum and minimum values must be at least e

Outline

- What is privacy and trust?
- Privacy in social network
  - Basic privacy requirement
  - Privacy in graph
- Trust in social network
- Reference

Possible Attacks on Anonymized Graphs

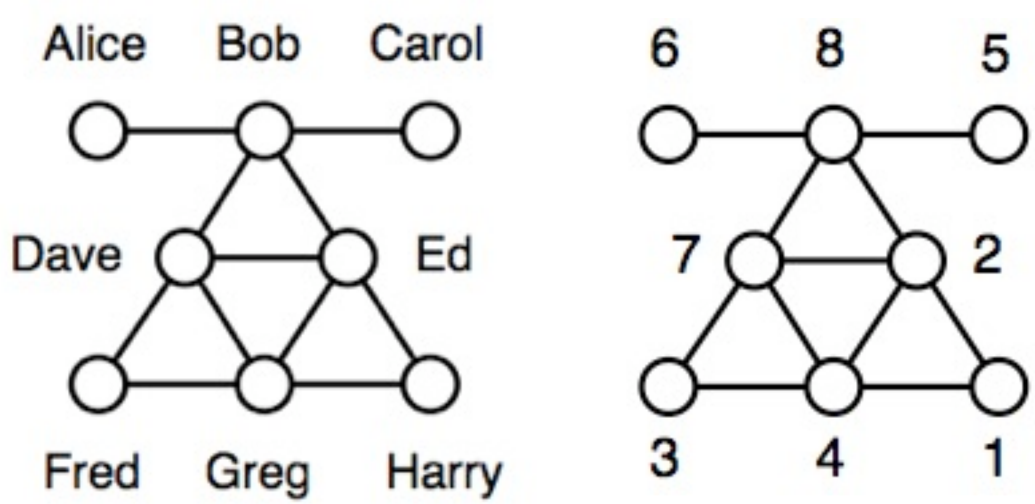
- Attack method [Michael Hay, 2008]
  - Identify by neighborhood information
  - It includes
    - Vertex Refinement Queries
    - Sub-graph Queries
    - Hub Fingerprint Queries

Possible Attacks on Anonymized Graphs

- Attack types [Lars Backstrom, 2008]
  - Active Attacks
    - Create a small number of new user accounts linking with other users before the anonymized graph is generated
  - Passive Attacks
    - Identify themselves in the published graph
  - Semi-passive Attacks
    - Create necessary link with other users

Vertex Refinement Queries

[Michael Hay, 2008]



Alice	6
Bob	8
Carol	5
Dave	7
Ed	2
Fred	3
Greg	4
Harry	1

(a) graph

Node ID	$\mathcal{H}_0$	$\mathcal{H}_1$	$\mathcal{H}_2$
Alice	ϵ	1	$\{4\}$
Bob	ϵ	4	$\{1,1,4,4\}$
Carol	ϵ	1	$\{4\}$
Dave	ϵ	4	$\{2,4,4,4\}$
Ed	ϵ	4	$\{2,4,4,4\}$
Fred	ϵ	2	$\{4,4\}$
Greg	ϵ	4	$\{2,2,4,4\}$
Harry	ϵ	2	$\{4,4\}$

(b) vertex refinements

Equivalence Relation	Equivalence Classes
$\equiv_{\mathcal{H}_0}$	$\{A,B,C,D,E,F,G,H\}$
$\equiv_{\mathcal{H}_1}$	$\{A,C\}$ $\{B,D,E,G\}$ $\{F,H\}$
$\equiv_{\mathcal{H}_2}$	$\{A,C\}\{B\}\{D,E\}\{G\}\{F,H\}$
$\equiv_{A}$	$\{A,C\}\{B\}\{D,E\}\{G\}\{F,H\}$

(c) equivalence classes

H*'s computation is linear in the number of edges in the graph!

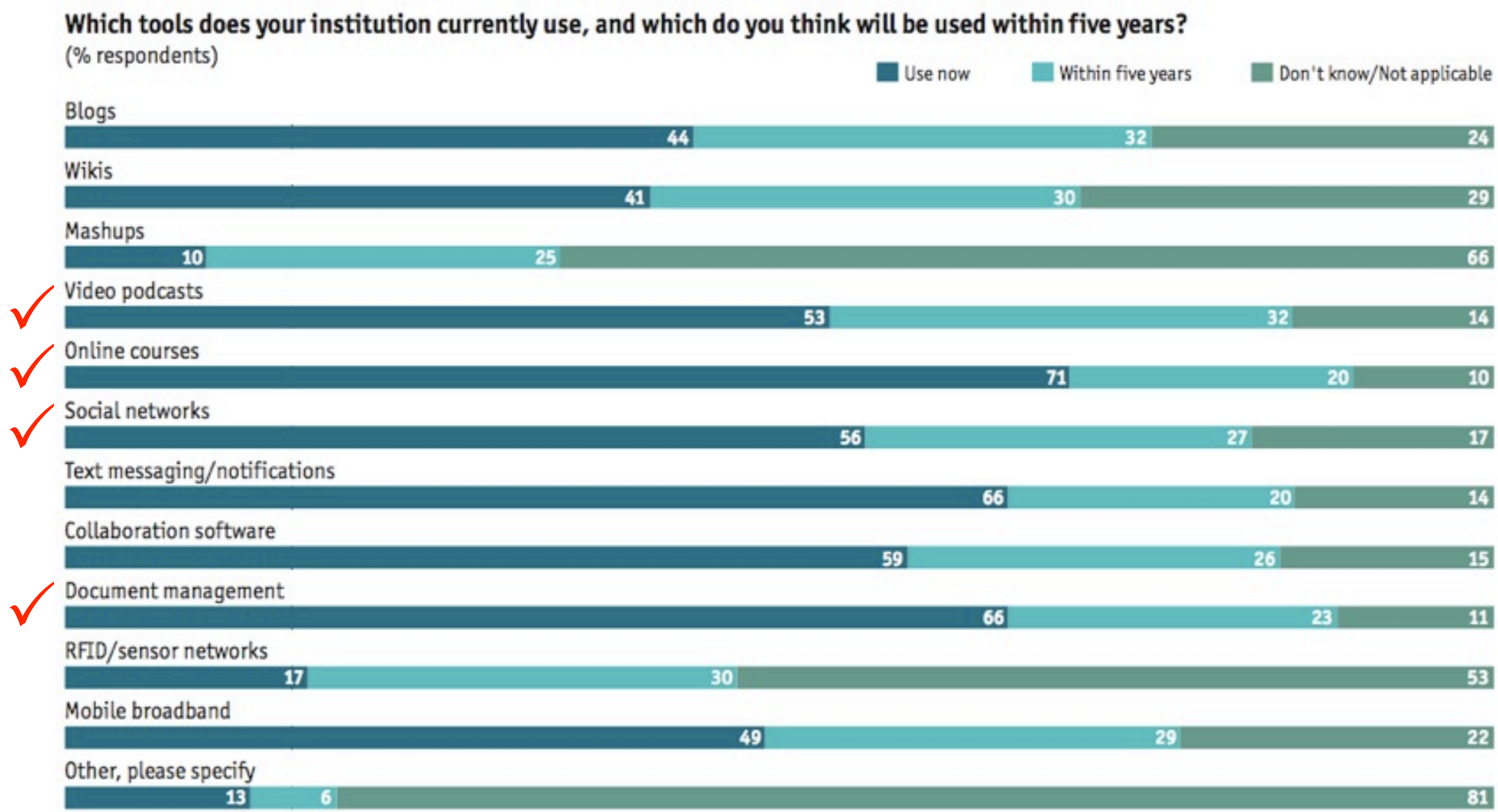
Summary

- Data privacy and security is a real and serious issue
- *k*-Anonymity and *l*-Diversity could help but may not be watertight
- Anonymizing graphs through graph generalization, node partitioning, and graph summarization

References

- L. Sweeney. k-anonymity: a model for protecting privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 2002
- Ashwin Machanavajjhala , Daniel Kifer , Johannes Gehrke , Muthuramakrishnan Venkitasubramaniam, L-diversity: Privacy beyond k-anonymity, TKDD, 2007
- Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian, t-Closeness: Privacy Beyond k-Anonymity and l-Diversity, ICDE, 2007.
- Xiao, X., Tao, Y, Dynamic Anonymization: Accurate Statistical Analysis with Privacy Preservation, SIGMOD, 2008.
- Michael Hay, Gerome Miklau, David Jensen, Don Towsley and Philipp Weis, Resisting Structural Re-identification in Anonymized Social Networks, PVLDB, 2008
- Lars Backstrom, Cynthia Dwork and Jon Kleinberg, Wherefore Art Thou R3579X? Anonymized Social Networks, Hidden Patterns, and Structural Steganography, WWW, 2007
- Kun liu and Evimaria Terzi, Towards Identity Anonymization on Graphs. SIGMOD, 2008
- Bin Zhou and Jian Pei, Preserving Privacy in Social Networks Against Neighborhood Attacks, ICDE, 2008

Economist Intelligent Unit 2008



Concluding Remarks

- Social Computing is here to stay!
- Relations are important!
- Discovering new paradigms by blending different social media and interactions
- Be concerned about computational techniques to search, rank, and mine data and information to achieve collective intelligence/wisdom

Acknowledgments

- Prof. Michael Lyu
- Mr. Patrick Lau
- Mr. Lam Cho Fung
- Mr. Simon Mok
- Mr. Ivan Yau
- Ms. Sara Fok
- Hongbo Deng (Ph.D.)
- Baichuan Li (M.Phil.)
- Zhenjiang Lin (Ph.D.)
- Hao Ma (Ph.D.)
- Mingzhe Mo (M.Phil.)
- Dingyan Wang (M.Phil.)
- Wei Wang (M.Phil.)
- Haiqin Yang (Ph.D.)
- Connie Yuen (Ph.D.)
- Xin Xin (Ph.D.)
- Chao Zhou (Ph.D.)
- Yi Zhu (Ph.D.)

On-Going Research

Machine Learning

- Heavy-Tailed Symmetric Stochastic Neighbor Embedding (NIPS'09)
- Adaptive Regularization for Transductive Support Vector Machine (NIPS'09)
- Direct Zero-norm Optimization for Feature Selection (ICDM'08)
- Semi-supervised Learning from General Unlabeled Data (ICDM'08)
- Learning with Consistency between Inductive Functions and Kernels (NIPS'08)
- An Extended Level Method for Efficient Multiple Kernel Learning (NIPS'08)
- Semi-supervised Text Categorization by Active Search (CIKM'08)
- Transductive Support Vector Machine (NIPS'07)
- Global and local learning (ICML'04, JMLR'04)

On-Going Research

Web Intelligence/Information Retrieval

- A Generalized Co-HITS Algorithm and Its Application to Bipartite Graphs (KDD'09)
- Entropy-biased Models for Query Representation on the Click Graph (SIRIR'09)
- Effective Latent Space Graph-based Re-ranking Model with Global Consistency (WSDM'09)
- Formal Models for Expert Finding on DBLP Bibliography Data (ICDM'08)
- Learning Latent Semantic Relations from Query Logs for Query Suggestion (CIKM'08)
- RATE: a Review of Reviewers in a Manuscript Review Process (WI'08)
- MatchSim: link-based web page similarity measurements (WI'07)
- Diffusion rank: Ranking web pages based on heat diffusion equations (SIGIR'07)
- Web text classification (WWW'07)

On-Going Research

Recommender Systems/Collaborative Filtering

- Learning to Recommend with Social Trust Ensemble (SIRIR'09)
- Semi-Nonnegative Matrix Factorization with Global Statistical Consistency in Collaborative Filtering (CIKM'09)
- Recommender system: accurate recommendation based on sparse matrix (SIGIR'07)
- SoRec: Social Recommendation Using Probabilistic Matrix Factorization (CIKM'08)

Human Computation

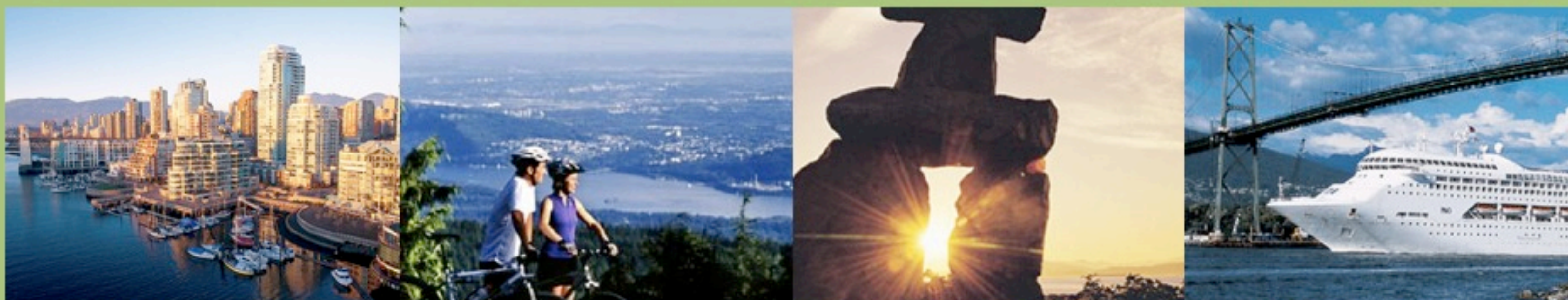
- A Survey of Human Computation Systems (SCA2009)
- Mathematical Modeling of Social Games (SIAG2009)
- An Analytical Study of Puzzle Selection Strategies for the ESP Game (WI'08)
- An Analytical Approach to Optimizing The Utility of ESP Games (WI'08)

Workshop on Social Computing in Education 2009

Home New since last time: 1 file

http://groups.google.com/group/WSCE2009

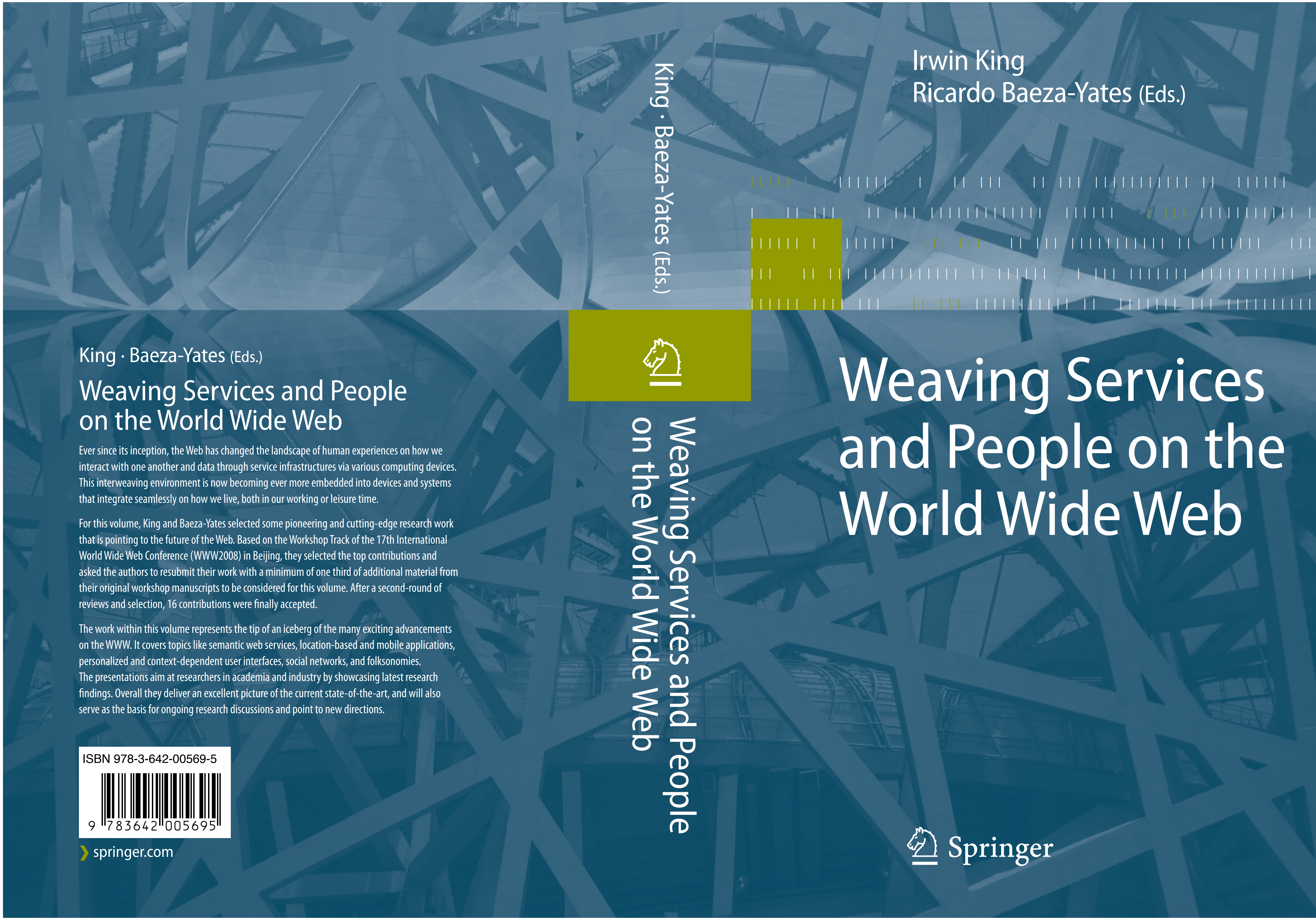
Call for Papers



Workshop on Social Computing in Education (WSCE2009)
in conjunction with SocialComp-09, August 29-31, 2009, Vancouver, Canada

Welcome to the workshop on Social Computing in Education (SCE2009). The workshop is held in conjunction with the SocialComp-09, Vancouver, Canada from August 29-31, 2009.

With the advent of Web 2.0 and related technologies, Social Computing has become a new paradigm in ways we communicate, learn, and educate. Social platforms such as wikis, blogs, twitters, forums, groups, podcasts, mashups, virtual worlds, and sites for social networking, recommender systems, social bookmarking, social news, knowledge sharing, etc. are generating novel ways we acquire, access, manipulate, process, retrieve, present, and visualize information in the teaching and learning space. The social media for education has become dynamic, ubiquitous, distributed, real-time, collaborative, bottom-up, many-to-many, value-based, and personalized. This workshop solicits contributions on using Social Computing and related technologies for education, the emerging applications of Web 2.0 as an educational platform, as well as privacy, risk, security, and policy issues associated in Social Computing for Education 2.0.

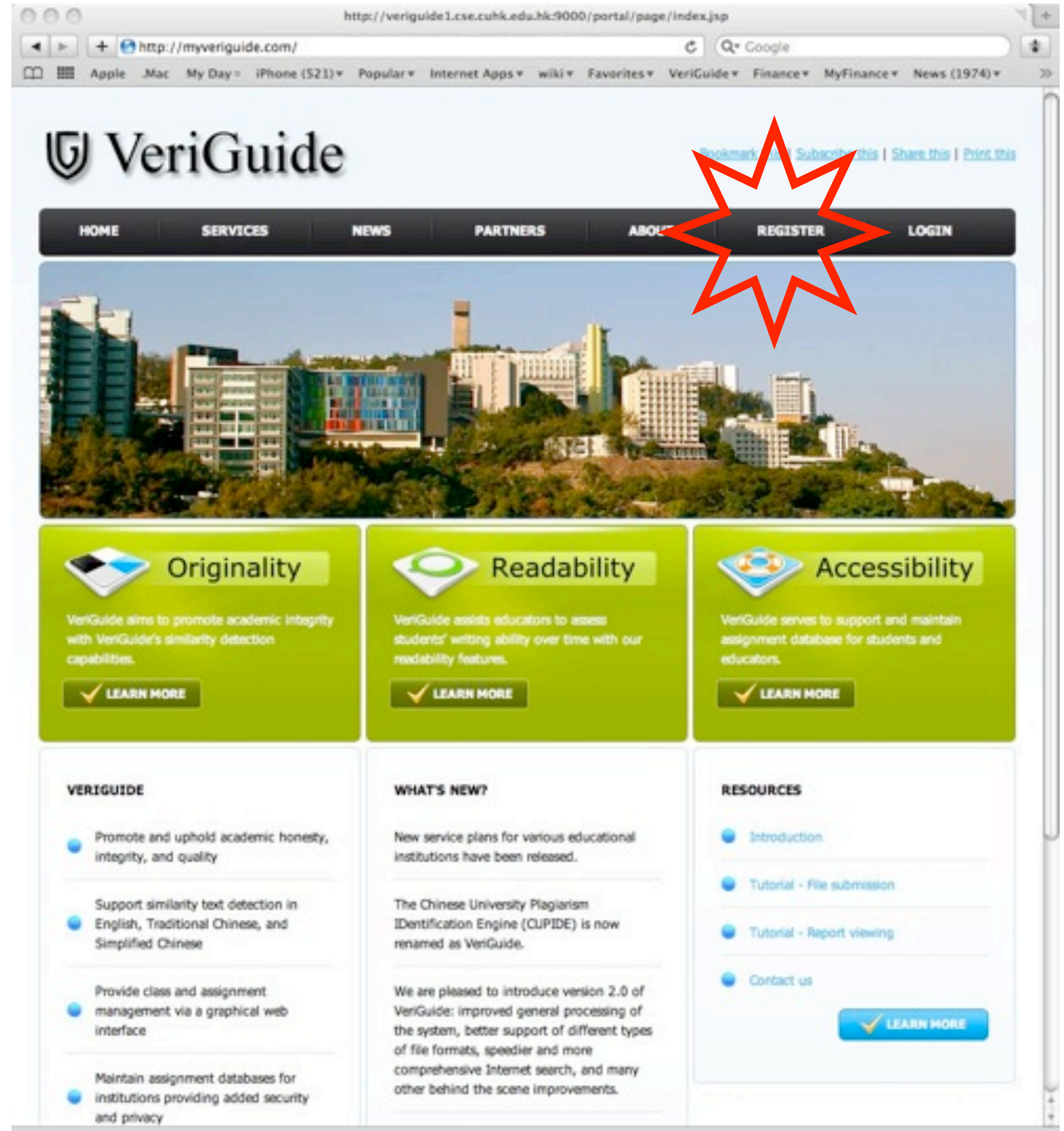

Irwin King
Ricardo Baeza-Yates (Eds.)
Weaving Services and People on the World Wide Web
Springer
King · Baeza-Yates (Eds.)
Weaving Services and People on the World Wide Web
King · Baeza-Yates (Eds.)
Weaving Services and People on the World Wide Web
Ever since its inception, the Web has changed the landscape of human experiences on how we interact with one another and data through service infrastructures via various computing devices. This interweaving environment is now becoming ever more embedded into devices and systems that integrate seamlessly on how we live, both in our working or leisure time.
For this volume, King and Baeza-Yates selected some pioneering and cutting-edge research work that is pointing to the future of the Web. Based on the Workshop Track of the 17th International World Wide Web Conference (WWW2008) in Beijing, they selected the top contributions and asked the authors to resubmit their work with a minimum of one third of additional material from their original workshop manuscripts to be considered for this volume. After a second-round of reviews and selection, 16 contributions were finally accepted.
The work within this volume represents the tip of an iceberg of the many exciting advancements on the WWW. It covers topics like semantic web services, location-based and mobile applications, personalized and context-dependent user interfaces, social networks, and folksonomies.
The presentations aim at researchers in academia and industry by showcasing latest research findings. Overall they deliver an excellent picture of the current state-of-the-art, and will also serve as the basis for ongoing research discussions and point to new directions.
ISBN 978-3-642-00569-5
springer.com

Economist Intelligent Unit 2008

Potential increase in student plagiarism

VeriGuide

- Similarity text detection system
- Developed at CUHK
- Promote and uphold academic honesty, integrity, and quality
- Support English, Traditional and Simplified Chinese
- Handle .doc, .txt, .pdf, .html, etc. file formats
- Generate detailed originality report including readability



VeriGuide Free Trial

IRWIN KING @ WEB INTELLIGENCE & SOCIAL COMPUTING LAB

Trace: » confs » record2008 » home

You are here: home

NAVIGATION

- Home
- Profile
- Research Interests & Projects

ABOUT US

- News | Newsletter
- Research Group | Presentations
- Collaborators
- Contact Us

PUBLICATIONS

1. Conference Papers 2005-Now
2. Journal Articles
3. Books, Edited Books & Proceedings
4. Book Chapters
5. Conference Papers 2000-2004
6. Conference Papers 1994-1999
7. Theses
8. Presentations

PROFESSIONAL ACTIVITIES

1. Professional Achievements
2. Awards
3. Grants
4. Teaching
5. Education Excellence
6. Demos & Software
   I. Finding Experts Demo
   II. MEMPM Matlab Toolbox
7. Conference Activities

Irwin King (金國慶), WISC Lab

Associate Professor, B.Sc. (Caltech), M.Sc., Ph.D. (USC)
SMIEEE (CIS), MACM, MINNS, APNNA

Department of Computer Science and Engineering
The Chinese University of Hong Kong, Shatin, NT, Hong Kong
Phone: +(852) 2609 8398; **Fax**: +(852) 2603 5024
Email: king [at] cse [dot] cuhk [dot] edu [dot] hk

- Associate Editor of IEEE Transactions on Neural Networks (IEEE TNN)
- Associate Editor of IEEE Computational Intelligence Magazine (IEEE CIM)
- Vice-President and Board Member of Asia Pacific Neural Network Assembly (APNNA)
- Chair, Task Force on the Future Directions of Neural Networks (IEEE CIS)
- Chair, SIG and Regional Chapters Committee for Asia and the Pacific, (INNS)
- Director of International Programmes, Faculty of Engineering (ERGIP)
- Member of RGC Engineering Panel, The Hong Kong SAR Government
- Co-Founder, Co-Principal Investigator and Chief Technologist, The VeriGuide Project
- General Co-Chair, Workshop on Social Computing in Education (WSCE2009), in conjunction with SocialComp'09
- General Co-Chair, Workshop on Social Web Search and Mining, in conjunction with CIKM2009
- Program Co-Chair, The first SIGMM Workshop on Social Media (WSM2009) in conjunction with ACM Multimedia 2009 (ACM MM'09), October 19-24, 2009, Beijing China

Research interests: Machine learning, social computing, web intelligence, information retrieval, multimedia information processing

Caltech's motto, "...the truth shall set you free."

News

- **Keynote, Invited Talk, Advisory Committee, Technical Program Committee Member, Reviewer, Panel Chair, Panelist, or Tutorial Speaker at** ICONIP'09, CollaborateCom2009, CIKM2009, ACML'09, ICCCI'09, APSIPA ASC 2009, WI'09, SocialCom-09, SIGIR2009, IJCAI-09, CASoN2009, IWSSIP2009, IJCNN2009, FAW2009,

http://www.cse.cuhk.edu.hk/~king