Tutorial AM-1

Hypertext Data Mining

Soumen Chakrabarti (IIT, Bombay)

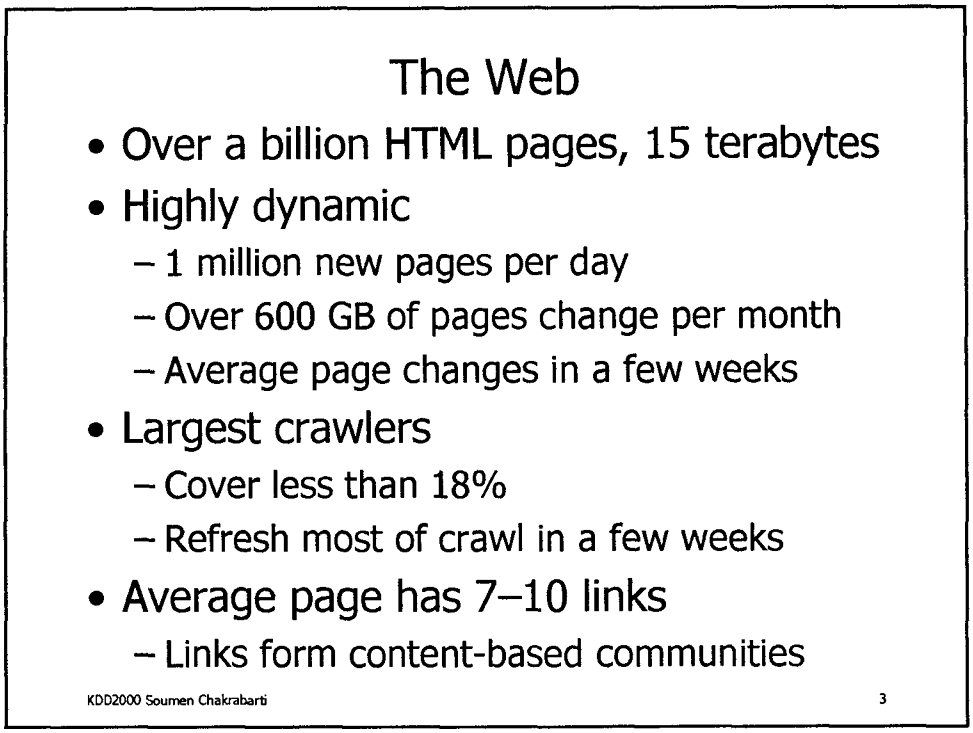
Hypertext Data Mining
(KDD 2000 Tutorial)

Soumen Chakrabarti
Indian Institute of Technology Bombay
http://www.cse.iitb.ernet.in/~soumen
http://www.cs.berkeley.edu/~soumen
soumen@cse.iitb.ernet.in

Hypertext databases

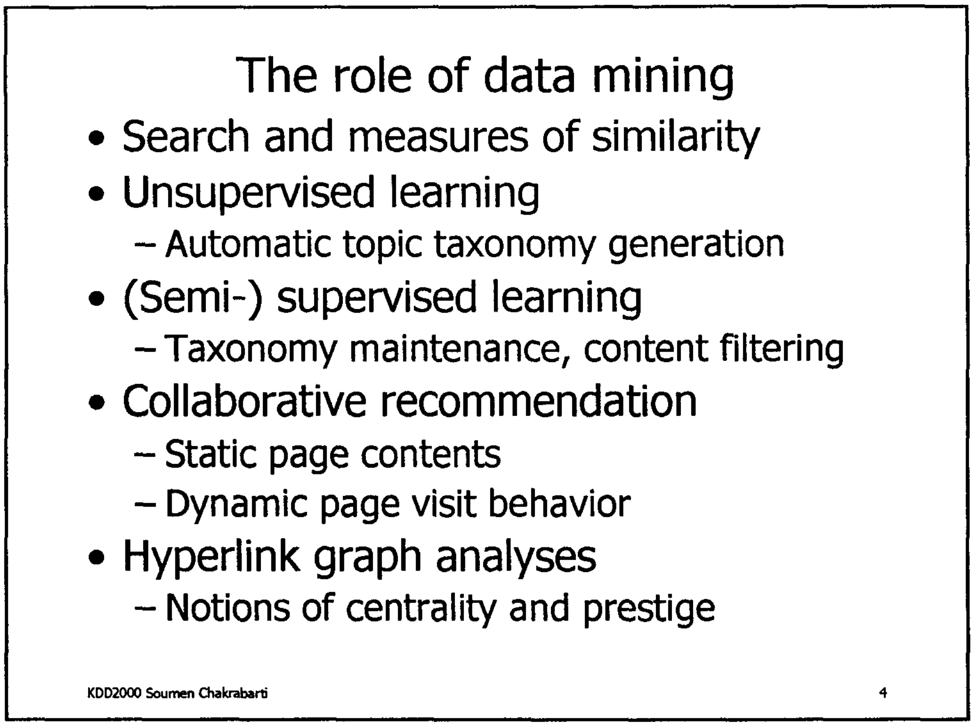
- Academia
  - Digital library, web publication
- Consumer
  - Newsgroups, communities, product reviews
- Industry and organizations
  - Health care, customer service
  - Corporate email

- An inherently collaborative medium
- Bigger than the sum of its parts



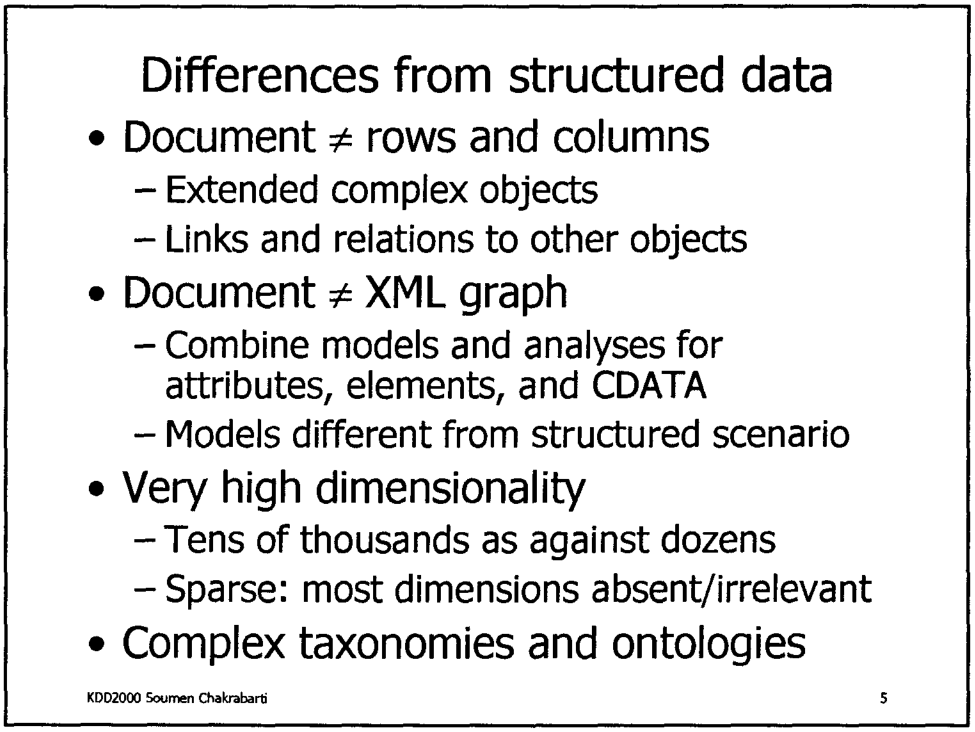
The Web

- Over a billion HTML pages, 15 terabytes
- Highly dynamic
  - 1 million new pages per day
  - Over 600 GB of pages change per month
  - Average page changes in a few weeks
- Largest crawlers
  - Cover less than 18%
  - Refresh most of crawl in a few weeks
- Average page has 7–10 links
  - Links form content-based communities



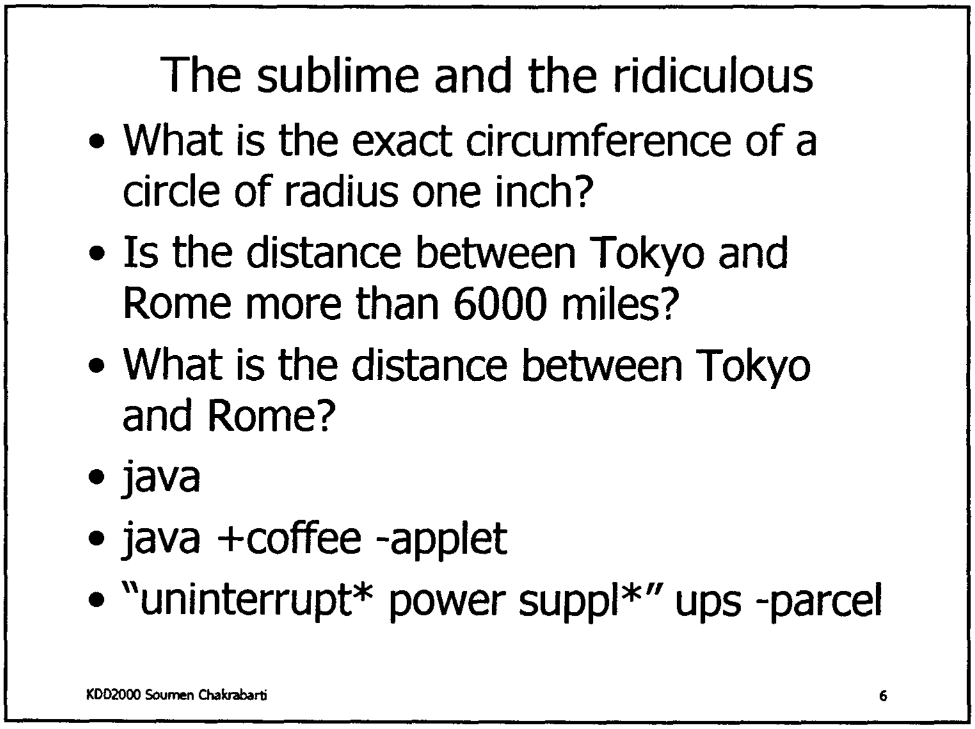
The role of data mining

- Search and measures of similarity
- Unsupervised learning
  - Automatic topic taxonomy generation
- (Semi-) supervised learning
  - Taxonomy maintenance, content filtering
- Collaborative recommendation
  - Static page contents
  - Dynamic page visit behavior
- Hyperlink graph analyses
  - Notions of centrality and prestige



Differences from structured data

- Document ≠ rows and columns
  - Extended complex objects
  - Links and relations to other objects
- Document ≠ XML graph
  - Combine models and analyses for attributes, elements, and CDATA
  - Models different from structured scenario
- Very high dimensionality
  - Tens of thousands as against dozens
  - Sparse: most dimensions absent/irrelevant
- Complex taxonomies and ontologies



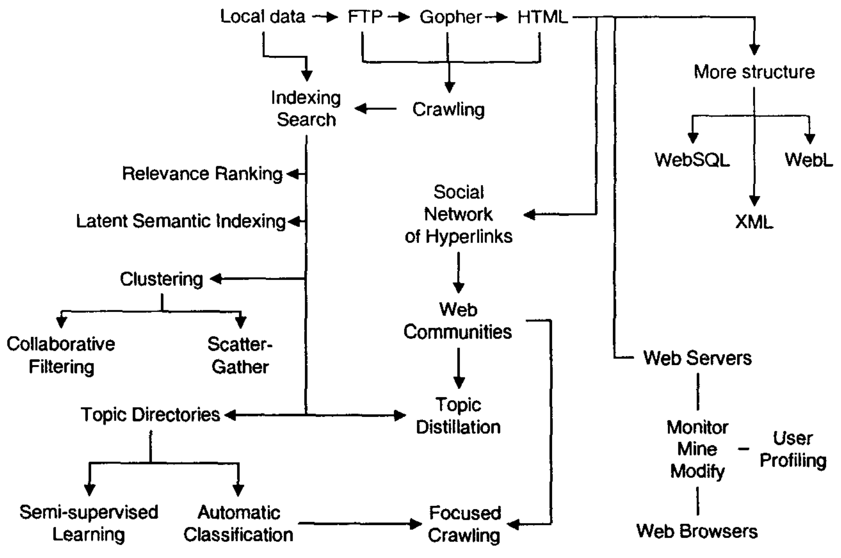
The sublime and the ridiculous

- What is the exact circumference of a circle of radius one inch?
- Is the distance between Tokyo and Rome more than 6000 miles?
- What is the distance between Tokyo and Rome?
- java
- java +coffee -applet
- "uninterrupt* power suppl*" ups -parcel

Search products and services

- Verity
- Fulcrum
- PLS
- Oracle text extender
- DB2 text extender
- Infoseek Intranet
- SMART (academic)
- Glimpse (academic)

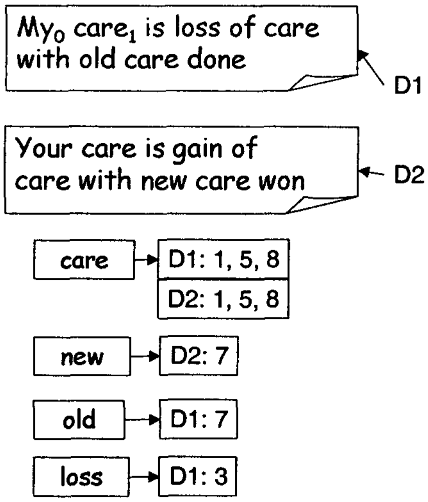
- Inktomi (HotBot)
- Alta Vista
- Raging Search
- Google
- Dmoz.org
- Yahoo!
- Infoseek Internet
- Lycos
- Excite



Basic indexing and search

Keyword indexing

- Boolean search
  - care AND NOT old
- Stemming
  - gain*
- Phrases and proximity
  - "new care"
  - loss NEAR/5 care
  - <SENTENCE>



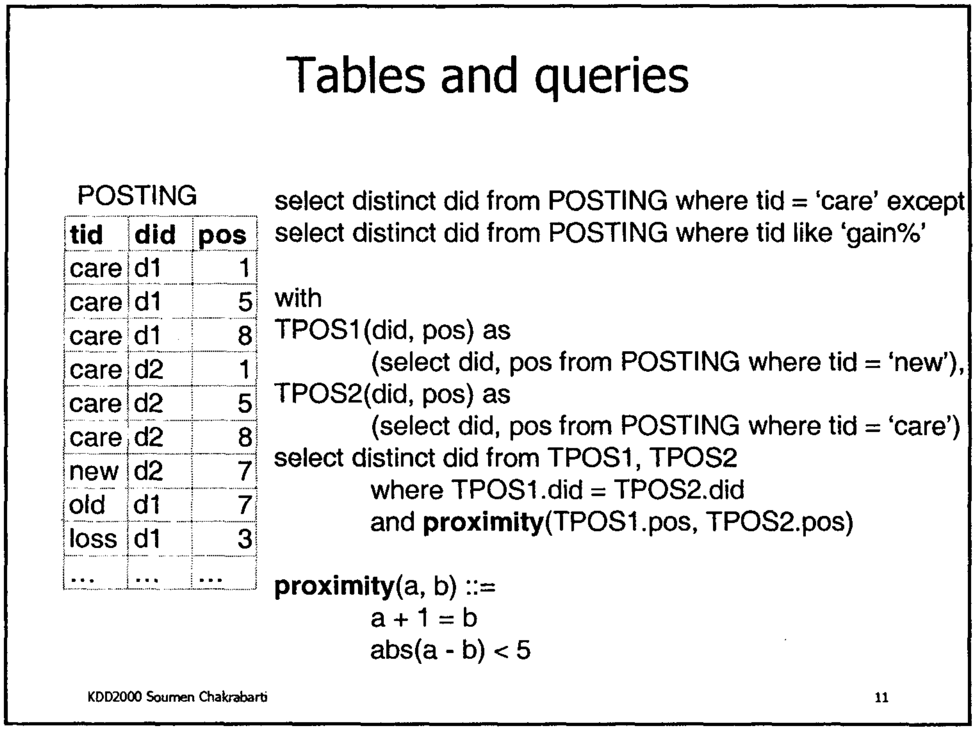
My_0 care_1 is loss of care with old care done

D1

Your care is gain of care with new care won

D2

care	D1: 1, 5, 8 D2: 1, 5, 8
new	D2: 7
old	D1: 7
loss	D1: 3



Tables and queries

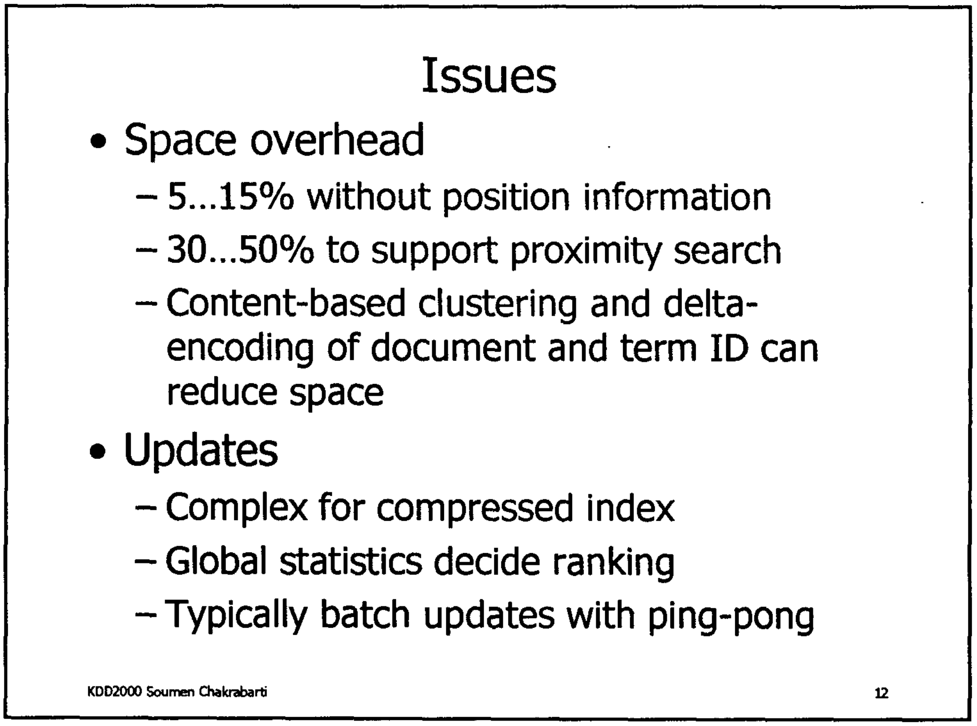
POSTING

tid	did	pos
care	d1	1
care	d1	5
care	d1	8
care	d2	1
care	d2	5
care	d2	8
new	d2	7
old	d1	7
loss	d1	3
...	...	...

```
select distinct did from POSTING where tid = 'care' except
select distinct did from POSTING where tid like 'gain%'

with
TPOS1(did, pos) as
        (select did, pos from POSTING where tid = 'new'),
TPOS2(did, pos) as
        (select did, pos from POSTING where tid = 'care')
select distinct did from TPOS1, TPOS2
        where TPOS1.did = TPOS2.did
        and proximity(TPOS1.pos, TPOS2.pos)

proximity(a, b) ::=
        a + 1 = b
        abs(a - b) < 5
```

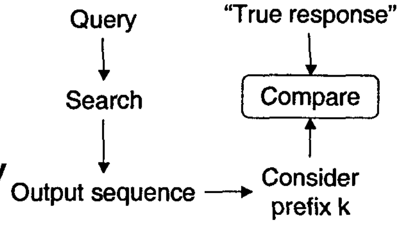


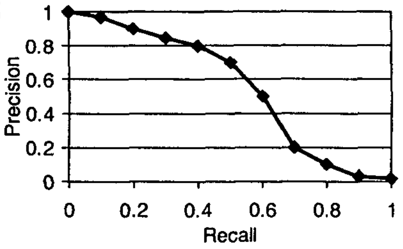
Issues

- Space overhead
  - 5...15% without position information
  - 30...50% to support proximity search
  - Content-based clustering and delta-encoding of document and term ID can reduce space
- Updates
  - Complex for compressed index
  - Global statistics decide ranking
  - Typically batch updates with ping-pong

Relevance ranking

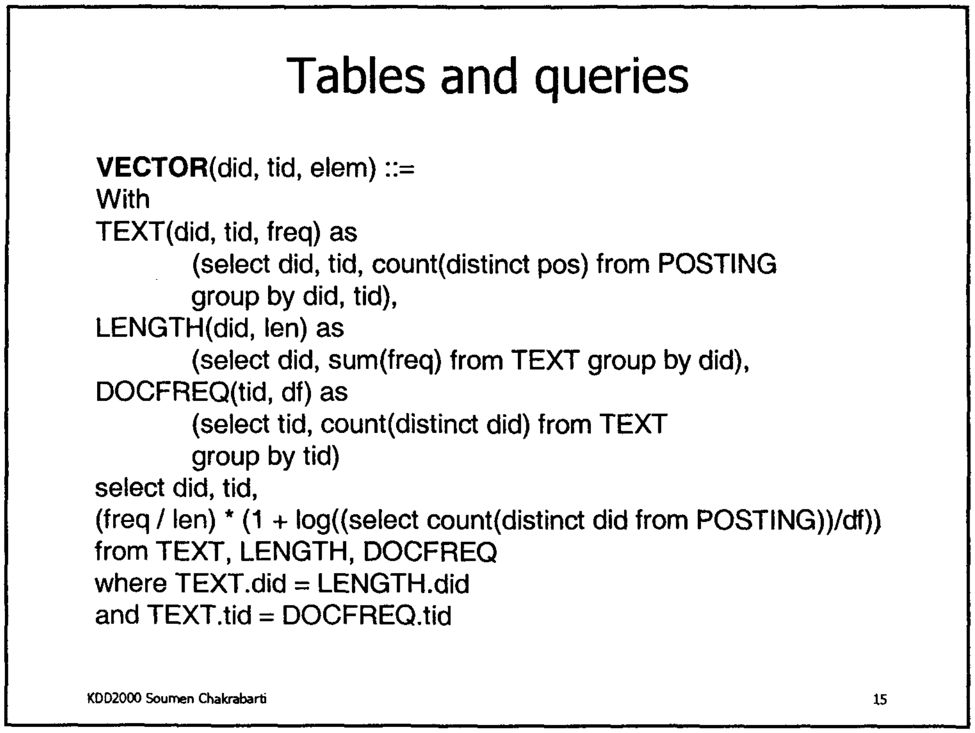
- Recall = coverage
  - What fraction of relevant documents were reported
- Precision = accuracy
  - What fraction of reported documents were relevant
- Trade-off
- 'Query' generalizes to 'topic'





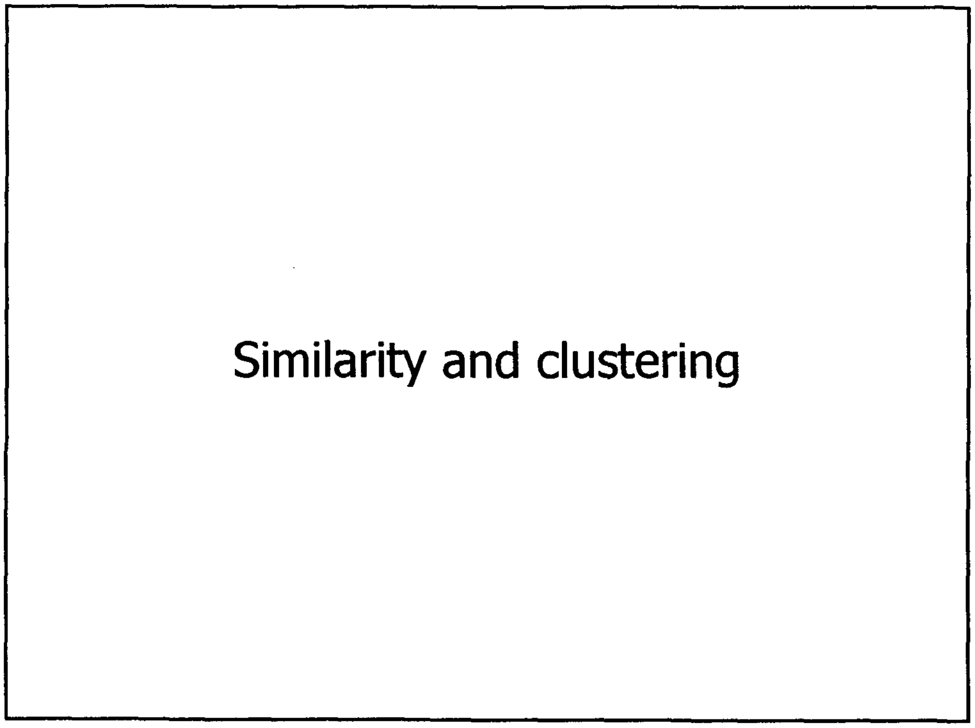
Vector space model and TFIDF

- Some words are more important than others
- W.r.t. a document collection *D*
  - d_+ have a term, d_- do not
  - "Inverse document frequency" $1+\log\frac{d_+ + d_-}{d+}$
- "Term frequency" (TF)
  - Many variants: $\frac{n(d,t)}{\sum_t n(d,t)}, \frac{n(d,t)}{\max_t n(d,t)}$
- Probabilistic models



Tables and queries

```
VECTOR(did, tid, elem) ::=
With
TEXT(did, tid, freq) as
        (select did, tid, count(distinct pos) from POSTING
        group by did, tid),
LENGTH(did, len) as
        (select did, sum(freq) from TEXT group by did),
DOCFREQ(tid, df) as
        (select tid, count(distinct did) from TEXT
        group by tid)
select did, tid,
(freq / len) * (1 + log((select count(distinct did from POSTING))/df))
from TEXT, LENGTH, DOCFREQ
where TEXT.did = LENGTH.did
and TEXT.tid = DOCFREQ.tid
```



Similarity and clustering

Clustering

- Given an unlabeled collection of documents, induce a taxonomy based on similarity (such as Yahoo)
- Need document similarity measure
  - Represent documents by TFIDF vectors
  - Distance between document vectors
  - Cosine of angle between document vectors
- Issues
  - Large number of noisy dimensions
  - Notion of noise is application dependent

Document model

- Vocabulary *V*, term *w*$_i$, document α represented by $c(\alpha)=\{f(w_i,\alpha)\}_{w_i\in V}$
- $f(w_i,\alpha)$ is the number of times w_i occurs in document α
- Most *f*s are zeroes for a single document
- Monotone component-wise damping function *g* such as log or square-root

$$g(c(\alpha))=\{g(f(w_i,\alpha))\}_{w_i\in V}$$

Similarity

$$s(\alpha,\beta)=\frac{\langle g(c(\alpha)),g(c(\beta))\rangle}{\|g(c(\alpha))\|\cdot\|g(c(\beta))\|}$$

$\langle\cdot,\cdot\rangle = \text{inner product}$

Normalized document profile: $$p(\alpha)=\frac{g(c(\alpha))}{\|g(c(\alpha))\|}$$

Profile for document group Γ: $$p(\Gamma)=\frac{\sum_{\alpha\in\Gamma}p(\alpha)}{\left\|\sum_{\alpha\in\Gamma}p(\alpha)\right\|}$$

Top-down clustering

- *k*-Means: Repeat…
  - Choose *k* arbitrary 'centroids'
  - Assign each document to nearest centroid
  - Recompute centroids
- Expectation maximization (EM):
  - Pick *k* arbitrary 'distributions'
  - Repeat:
    - Find probability that document *d* is generated from distribution *f* for all *d* and *f*
    - Estimate distribution parameters from weighted contribution of documents

Bottom-up clustering

$$s(\Gamma)=\frac{1}{|\Gamma|(|\Gamma|-1)}\sum_{\alpha\in\Gamma}\sum_{\beta\neq\alpha}s(\alpha,\beta)$$

- Initially *G* is a collection of singleton groups, each with one document
- Repeat
  - Find Γ, Δ in *G* with max $s(\Gamma\cup\Delta)$
  - Merge group Γ with group Δ
- For each Γ keep track of best Δ
- $O(n^2\log n)$ algorithm with n^2 space

Updating group average profiles

Un-normalized group profile: $\hat{p}(\Gamma)=\sum_{\alpha\in\Gamma}p(\alpha)$

Can show:

$$s(\Gamma)=\frac{\langle\hat{p}(\Gamma),\hat{p}(\Gamma)\rangle-|\Gamma|}{|\Gamma|(|\Gamma|-1)}$$

$$s(\Gamma\cup\Lambda)=\frac{\langle\hat{p}(\Gamma\cup\Delta),\hat{p}(\Gamma\cup\Delta)\rangle-(|\Gamma|+|\Delta|)}{(|\Gamma|+|\Delta|)(|\Gamma|+|\Delta|-1)}$$

$$\langle\hat{p}(\Gamma\cup\Delta),\hat{p}(\Gamma\cup\Delta)\rangle=\langle\hat{p}(\Gamma),\hat{p}(\Gamma)\rangle+\langle\hat{p}(\Delta),\hat{p}(\Delta)\rangle+2\langle\hat{p}(\Gamma),\hat{p}(\Delta)\rangle$$

"Rectangular time" algorithm

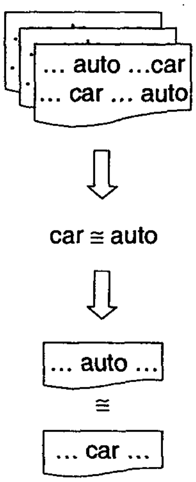
- Quadratic time is too slow
- Randomly sample $O(\sqrt{kn})$ documents
- Run group average clustering algorithm to reduce to *k* groups or clusters
- Iterate assign-to-nearest $O(1)$ times
  - Move each document to nearest cluster
  - Recompute cluster centroids
- Total time taken is $O(kn)$
- Non-deterministic behavior

Issues

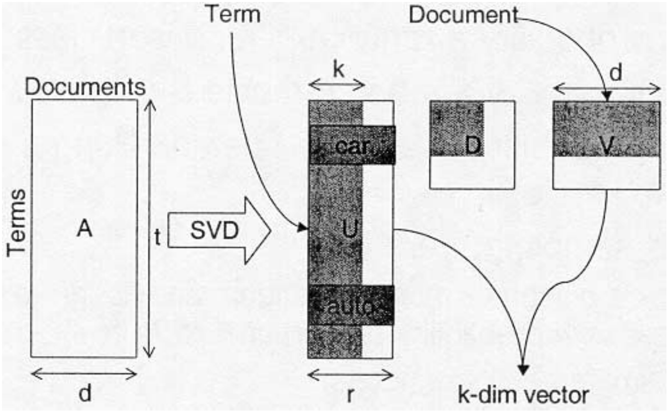
- Detecting noise dimensions
  - Bottom-up dimension composition too slow
  - Definition of noise depends on application
- Running time
  - Distance computation dominates
  - Random projections
  - Sublinear time w/o losing small clusters
- Integrating semi-structured information
  - Hyperlinks, tags embed similarity clues
  - A link is worth a ___?___ words

Extended similarity

- Where can I fix my scooter?
- A great garage to repair your 2-wheeler is at …
- auto and car co-occur often
- Documents having related words are related
- Useful for search and clustering
- Two basic approaches
  - Hand-made thesaurus (WordNet)
  - Co-occurrence and associations

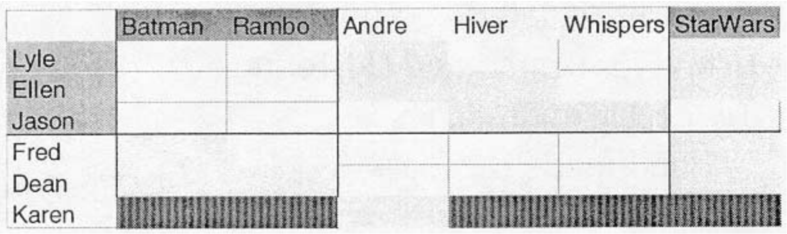

… auto …car
… car … auto
car ≅ auto
… auto …
≅
… car …

Latent semantic indexing



Collaborative recommendation

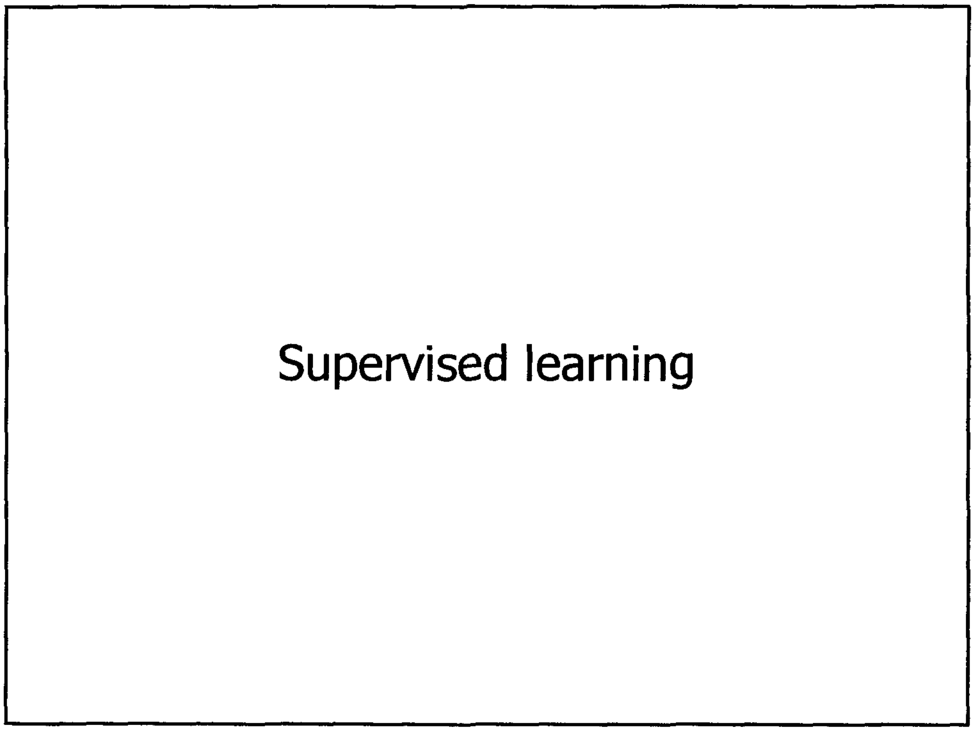
- People=record, movies=features
- People and features to be clustered
  - Mutual reinforcement of similarity
- Need advanced models



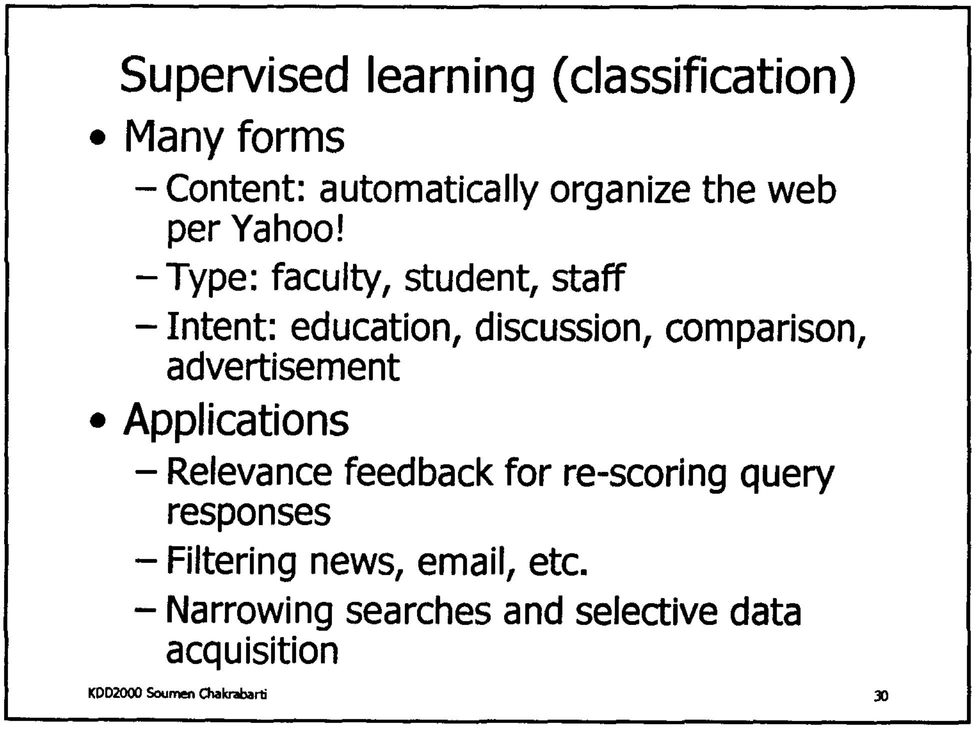
From *Clustering methods in collaborative filtering,* by Ungar and Foster

A model for collaboration

- People and movies belong to unknown classes
- P_k = probability a random person is in class k
- P_l = probability a random movie is in class l
- P_{kl} = probability of a class-k person liking a class-l movie
- Gibbs sampling: iterate
  - Pick a person or movie at random and assign to a class with probability proportional to P_k or P_l
  - Estimate new parameters

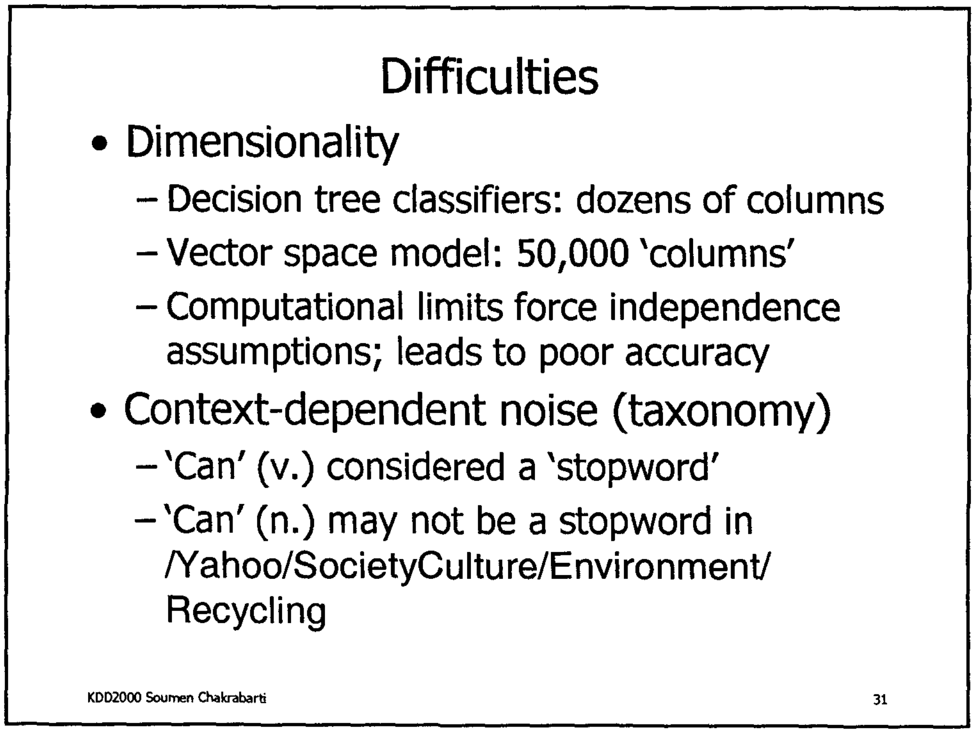


Supervised learning



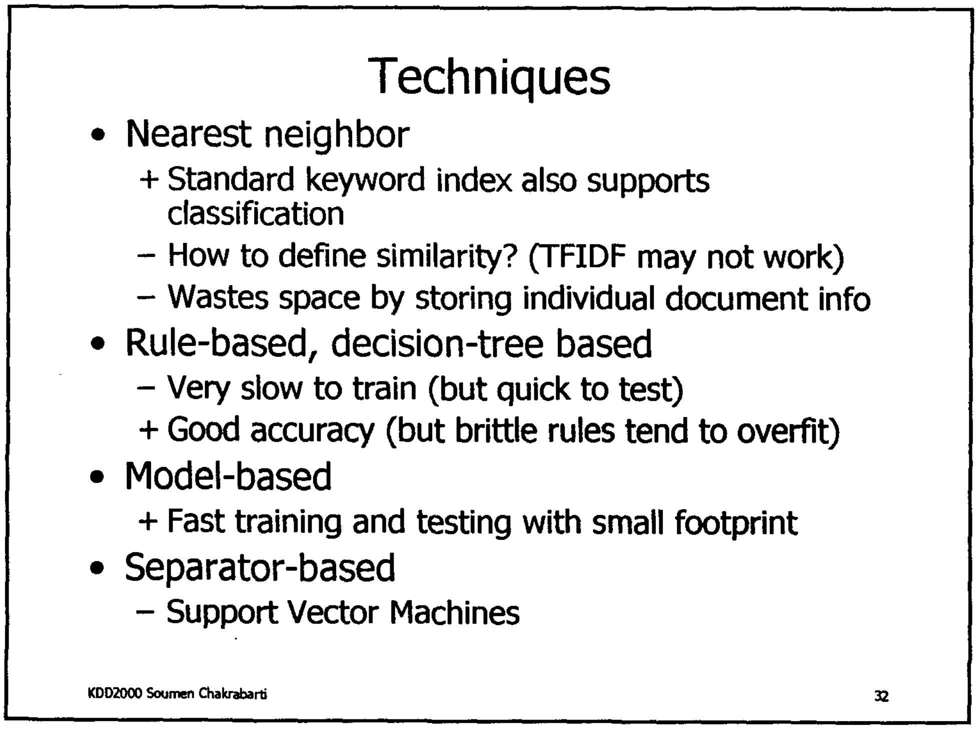
Supervised learning (classification)

- Many forms
  - Content: automatically organize the web per Yahoo!
  - Type: faculty, student, staff
  - Intent: education, discussion, comparison, advertisement
- Applications
  - Relevance feedback for re-scoring query responses
  - Filtering news, email, etc.
  - Narrowing searches and selective data acquisition



Difficulties

- Dimensionality
  - Decision tree classifiers: dozens of columns
  - Vector space model: 50,000 'columns'
  - Computational limits force independence assumptions; leads to poor accuracy
- Context-dependent noise (taxonomy)
  - 'Can' (v.) considered a 'stopword'
  - 'Can' (n.) may not be a stopword in /Yahoo/SocietyCulture/Environment/ Recycling



Techniques

- Nearest neighbor
  - + Standard keyword index also supports classification
  - − How to define similarity? (TFIDF may not work)
  - − Wastes space by storing individual document info
- Rule-based, decision-tree based
  - − Very slow to train (but quick to test)
  - + Good accuracy (but brittle rules tend to overfit)
- Model-based
  - + Fast training and testing with small footprint
- Separator-based
  - − Support Vector Machines

Document generation models

- Boolean vector (word counts ignored)
  - Toss one coin for each term in the universe
- Bag of words (multinomial)
  - Toss coin with a term on each face
- Limited dependence models
  - Bayesian network where each feature has at most *k* features as parents
  - Maximum entropy estimation
- Limited memory models
  - Markov models

"Bag-of-words"

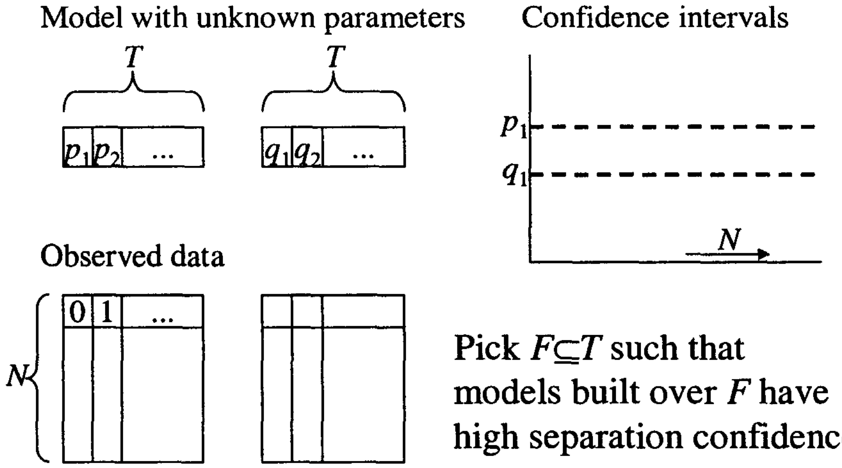
- Decide topic; topic *c* is picked with prior probability $\pi(c)$; $\Sigma_c \pi(c) = 1$
- Each topic *c* has parameters $\theta(c,t)$ for terms *t*
- Coin with face probabilities $\Sigma_t \theta(c,t) = 1$
- Fix document length and keep tossing coin
- Given *c*, probability of document is

$$\Pr[d \mid c] = \binom{n(d)}{\{n(d,t)\}} \prod_{t \in d} \theta(c,t)^{n(d,t)}$$

Limitations

- With the term distribution
  - 100th occurrence is as surprising as first
  - No inter-term dependence
- With using the model
  - Most observed $\theta(c,t)$ are zero and/or noisy
  - Have to pick a low-noise subset of the term universe
  - Have to "fix" low-support statistics
    - Smoothing and discretization
    - Coin turned up heads 100/100 times; what is Pr(tail) on the next toss?

Feature selection

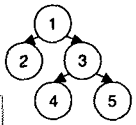


Pick $F \subseteq T$ such that models built over F have high separation confidence

Tables and queries

TAXONOMY

pcid	kcid	kcname
▨	1	▨
1	2	Arts
1	3	Science
3	4	Math
3	5	Physics



EGMAP

did	kcid

TEXT

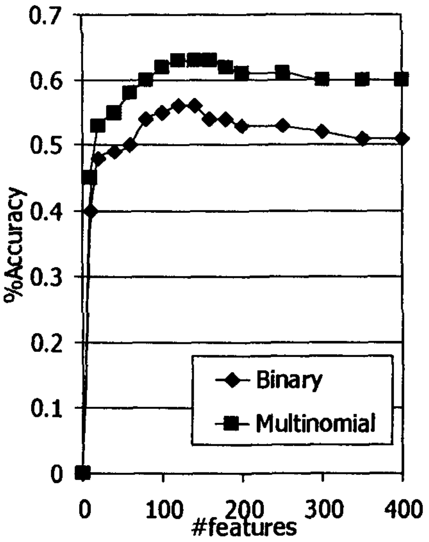
did	tid	freq

```
EGMAPR(did, kcid) ::=
        ((select did, kcid from EGMAP) union all
        (select e.did, t.pcid from
        EGMAPR as e, TAXONOMY as t
        where e.kcid = t.kcid))

STAT(pcid, tid, kcid, ksmc, ksnc) ::=
        (select pcid, tid, TAXONOMY.kcid,
        count(distinct TEXT.did), sum(freq)
        from EGMAPR, TAXONOMY, TEXT
        where TAXONOMY.kcid = EGMAPR.kcid
        and EGMAPR.did = TEXT.did
        group by pcid, tid, TAXONOMY.kcid)
```

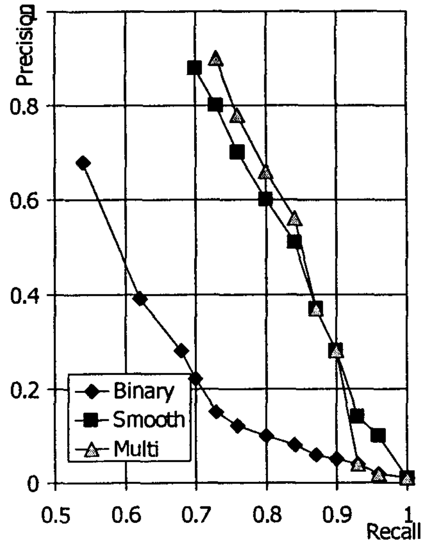
Effect of feature selection

- Sharp knee in error with small number of features
- Saves class model space
  - Easier to hold in memory
  - Faster classification
- Mild increase in error beyond knee
  - Worse for binary model



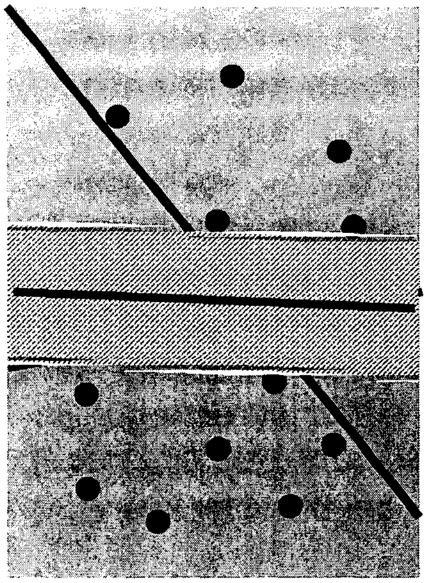
Effect of parameter smoothing

- Multinomial known to be more accurate than binary under Laplace smoothing
- Better marginal distribution model compensates for modeling term counts!
- Good parameter smoothing is critical



Support vector machines (SVM)

- No assumptions on data distribution
- Goal is to find separators
- Large bands around separators give better generalization
- Quadratic programming
- Efficient heuristics
- Best known results



Maximum entropy classifiers

- Observations (d_i,c_i), i = 1…N
- Want model $p(c\,|\,d)$, expressed using features $f_j(c, d)$ and parameters λ_j as

$$p(c\,|\,d) = \frac{1}{Z(d)} \prod_j e^{\lambda_j f_j(c,d)}, Z(d) = \sum_{c'} p(c'|\,d)$$

- Constraints given by observed data

$$\textstyle\sum_{d,c} \tilde{p}(d) p(c\,|\,d) f(d,c) = \sum_{d,c} \tilde{p}(d,c) f(d,c)$$

- Objective is to maximize entropy of p

$$H(p) = -\textstyle\sum_{d,c} \tilde{p}(d) p(c\,|\,d) \log p(c\,|\,d)$$

- Features
  - Numerical non-linear optimization
  - No naïve independence assumptions

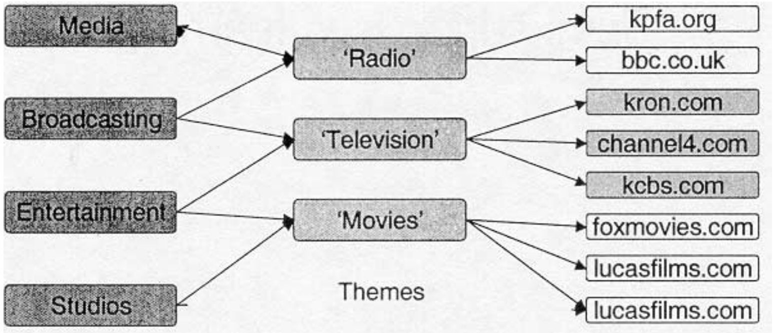
Semi-supervised learning

Exploiting unlabeled documents

- Unlabeled documents are plentiful; labeling is laborious
- Let training documents belong to classes in a *graded* manner Pr(*c*|*d*)
- Initially labeled documents have 0/1 membership
- Repeat (Expectation Maximization 'EM'):
  - Update class model parameters θ
  - Update membership probabilities Pr(*c*|*d*)
- Small labeled set→large accuracy boost

Mining themes from bookmarks

- Clustering with categorical attribute
- Unclear how to embed in a geometry
  - A folder is worth __?__ words?
- Unified model for three similarity clues



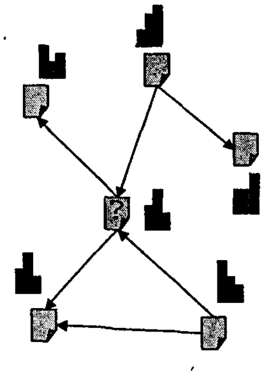
Analyzing hyperlink structure

Hyperlink graph analysis

- Hypermedia is a **social network**
  - Telephoned, advised, co-authored, paid
- Social network theory (cf. Wasserman & Faust)
  - Extensive research applying graph notions
  - **Centrality and prestige**
  - **Co-citation (relevance judgment)**
- Applications
  - Web search: HITS, Google, CLEVER
  - Classification and topic distillation

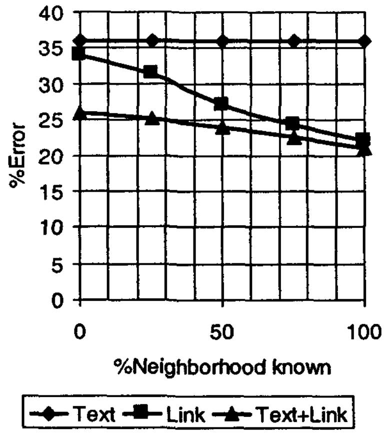
Hypertext models for classification

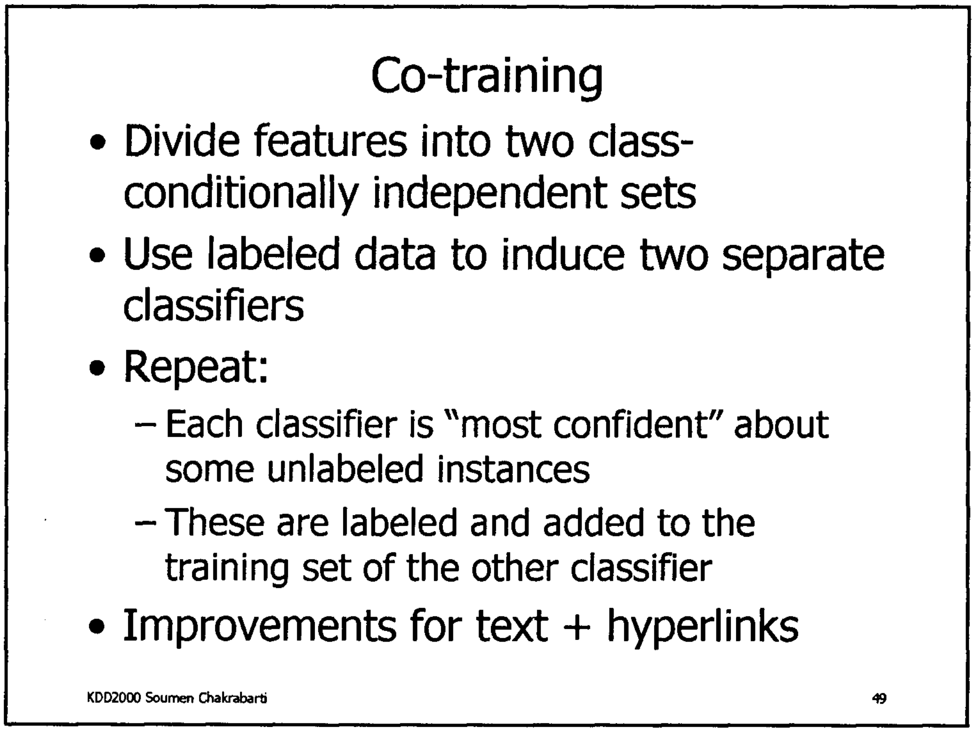
- *c*=class, *t*=text, *N*=neighbors
- Text-only model: Pr[*t*|*c*]
- Using neighbors' text to judge my topic: Pr[*t*, *t*(*N*) | *c*]
- Better model: Pr[*t*, *c*(*N*) | *c*]
- Non-linear relaxation



Exploiting link features

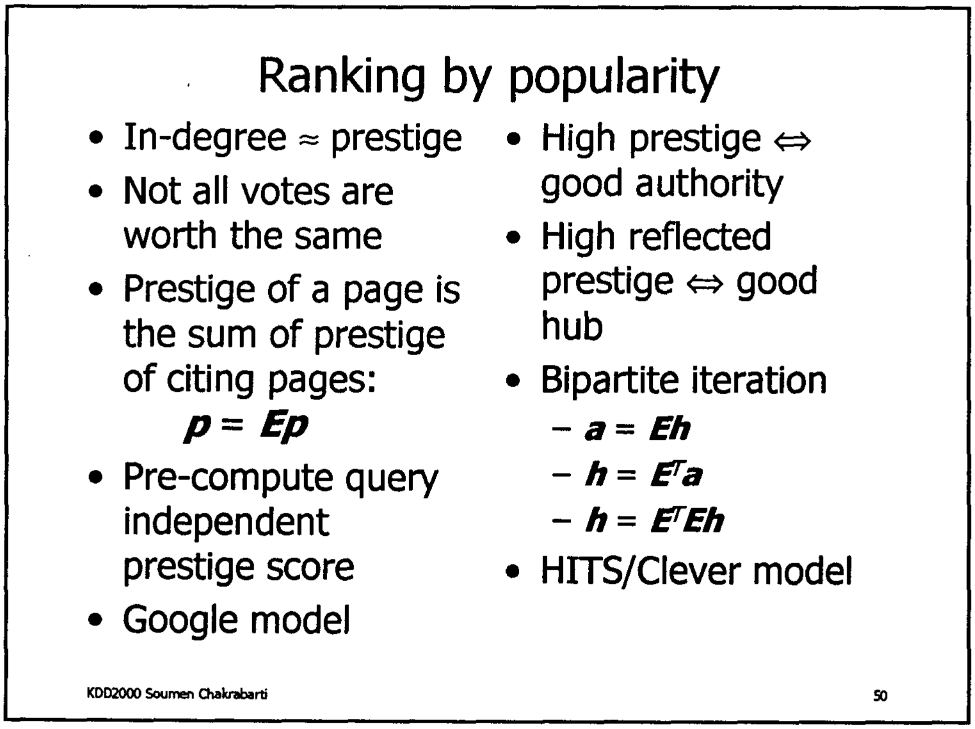
- 9600 patents from 12 classes marked by USPTO
- Patents have text and cite other patents
- Expand test patent to include neighborhood
- 'Forget' fraction of neighbors' classes





Co-training

- Divide features into two class-conditionally independent sets
- Use labeled data to induce two separate classifiers
- Repeat:
  - Each classifier is "most confident" about some unlabeled instances
  - These are labeled and added to the training set of the other classifier
- Improvements for text + hyperlinks



Ranking by popularity

- In-degree ≈ prestige
- Not all votes are worth the same
- Prestige of a page is the sum of prestige of citing pages:

  $\boldsymbol{p} = \boldsymbol{Ep}$
- Pre-compute query independent prestige score
- Google model
- High prestige ⇔ good authority
- High reflected prestige ⇔ good hub
- Bipartite iteration
  - $\boldsymbol{a} = \boldsymbol{Eh}$
  - $\boldsymbol{h} = \boldsymbol{E}^T\boldsymbol{a}$
  - $\boldsymbol{h} = \boldsymbol{E}^T\boldsymbol{Eh}$
- HITS/Clever model

Tables and queries

HUBS

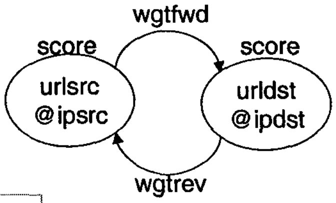
url	score

AUTH

url	score

```
delete from HUBS;
insert into HUBS(url, score)
        (select urlsrc, sum(score * wtrev) from AUTH, LINK
        where authwt is not null and type = non-local
        and ipdst <> ipsrc and url = urldst
        group by urlsrc);
update HUBS set (score) = score /
        (select sum(score) from HUBS);
```

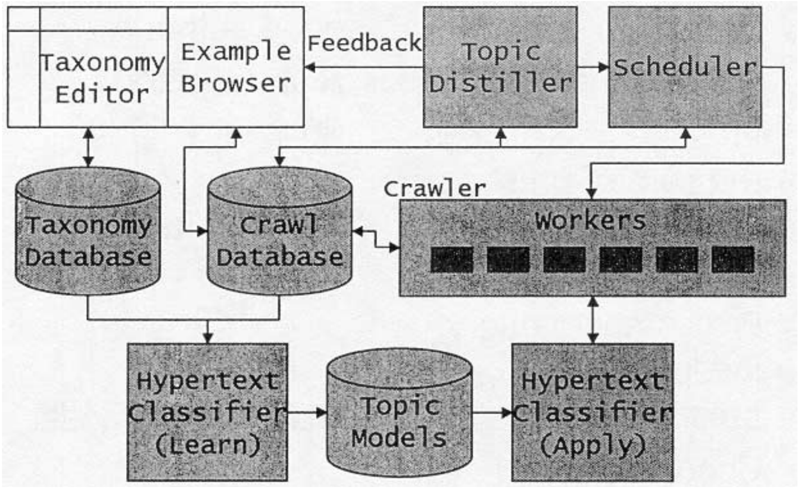
```
update LINK as X set (wtfwd) = 1. /
        (select count(ipsrc) from LINK
        where ipsrc = X.ipsrc
        and urldst = X.urldst)
        where type = non-local;
```



LINK

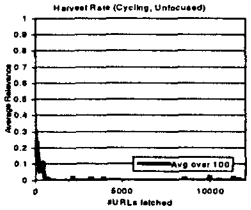
urlsrc	urldst	ipsrc	ipdst	wgtfwd	wtrev	type

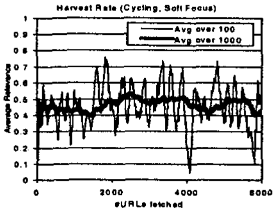
Resource discovery

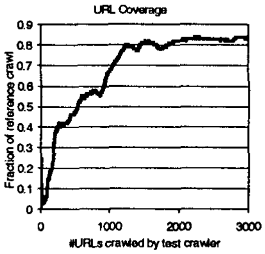


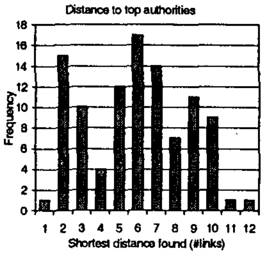
Resource discovery results

- High rate of "harvesting" relevant pages
- Robust to perturbations of starting URLs
- Great resources found 12 links from start set

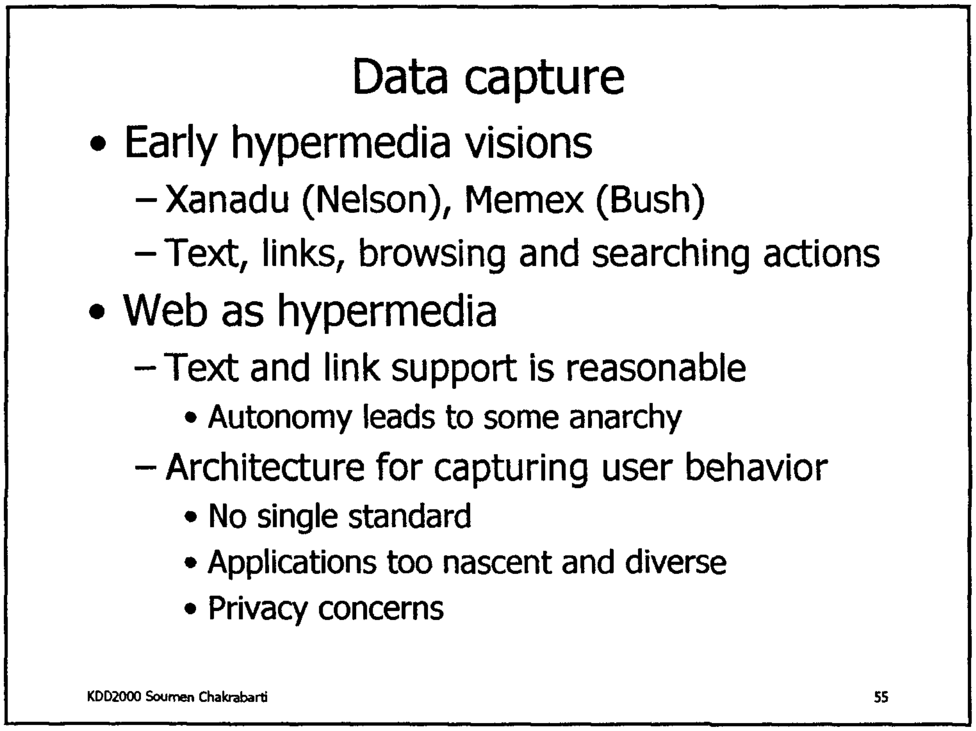






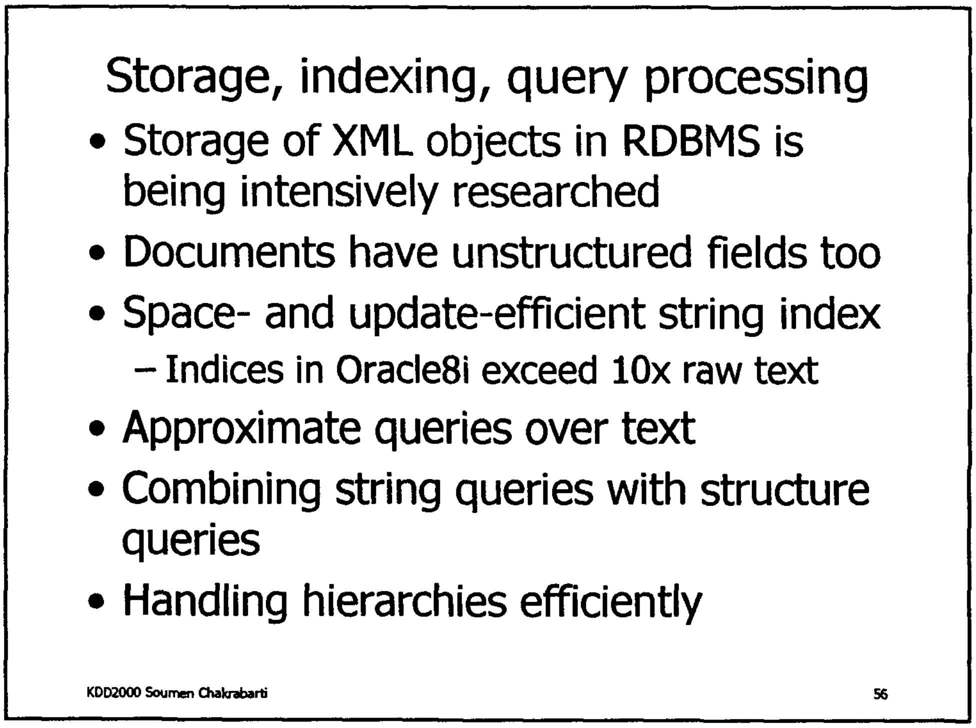


Systems issues



Data capture

- Early hypermedia visions
  - Xanadu (Nelson), Memex (Bush)
  - Text, links, browsing and searching actions
- Web as hypermedia
  - Text and link support is reasonable
    - Autonomy leads to some anarchy
  - Architecture for capturing user behavior
    - No single standard
    - Applications too nascent and diverse
    - Privacy concerns



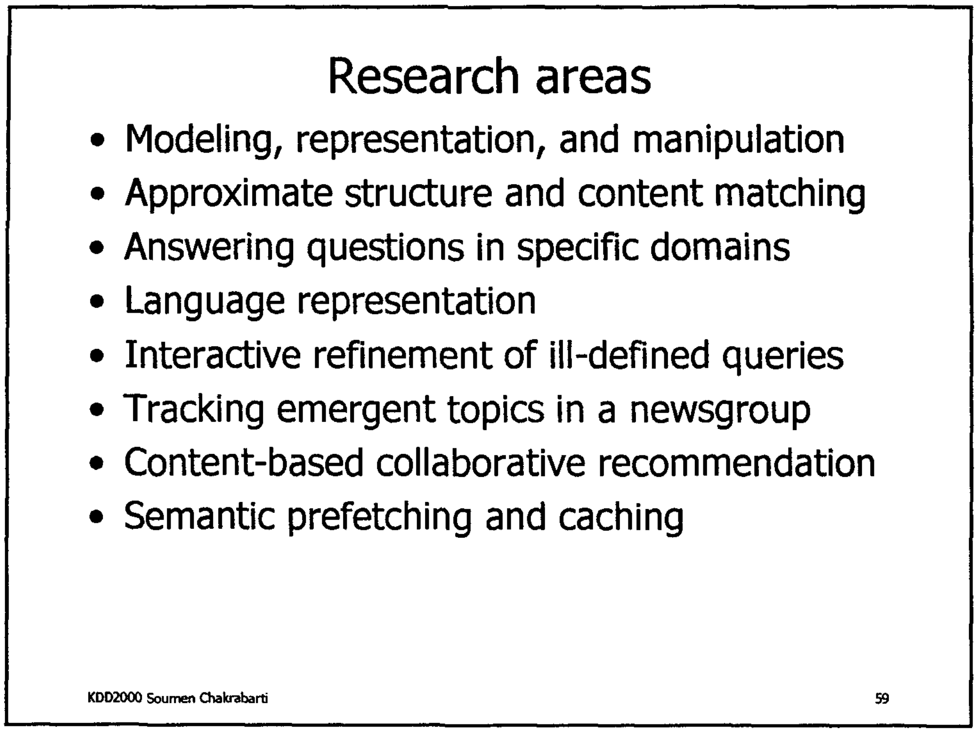
Storage, indexing, query processing

- Storage of XML objects in RDBMS is being intensively researched
- Documents have unstructured fields too
- Space- and update-efficient string index
  - Indices in Oracle8i exceed 10x raw text
- Approximate queries over text
- Combining string queries with structure queries
- Handling hierarchies efficiently

Concurrency and recovery

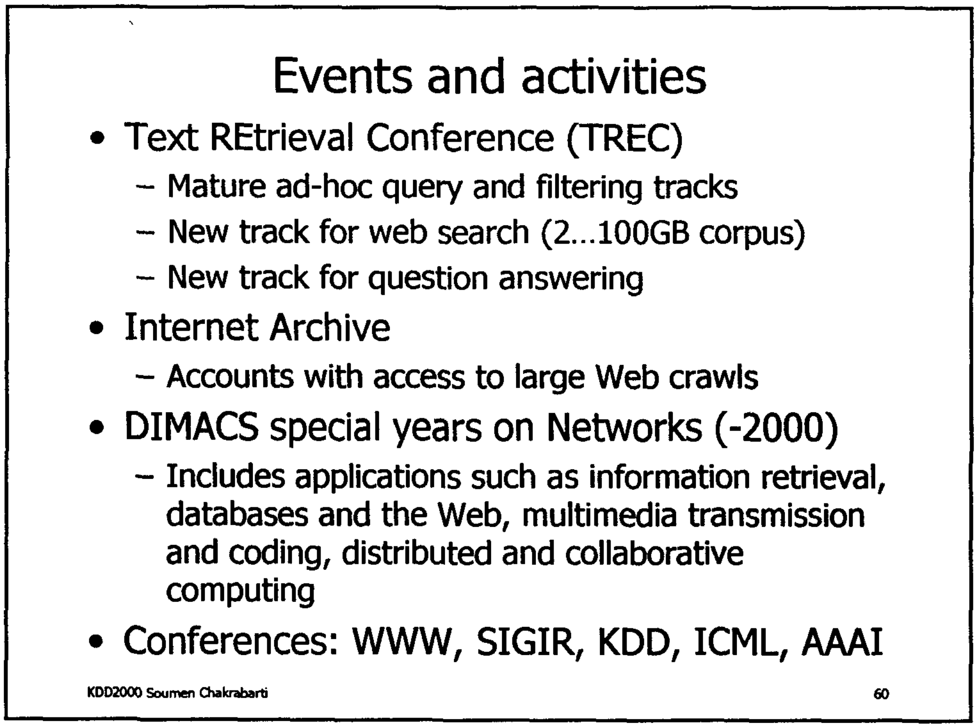
- Strong RDBMS features
  - Useful in medium-sized crawlers
- Not sufficiently flexible
  - Unlogged tables, columns
  - Lazy indices and concurrent work queues
- Advances query processing
  - Index (-ed scans) over temporary table expressions; multi-query optimization
  - Answering complex queries approximately

Resources



Research areas

- Modeling, representation, and manipulation
- Approximate structure and content matching
- Answering questions in specific domains
- Language representation
- Interactive refinement of ill-defined queries
- Tracking emergent topics in a newsgroup
- Content-based collaborative recommendation
- Semantic prefetching and caching



Events and activities

- Text REtrieval Conference (TREC)
  - Mature ad-hoc query and filtering tracks
  - New track for web search (2...100GB corpus)
  - New track for question answering
- Internet Archive
  - Accounts with access to large Web crawls
- DIMACS special years on Networks (-2000)
  - Includes applications such as information retrieval, databases and the Web, multimedia transmission and coding, distributed and collaborative computing
- Conferences: WWW, SIGIR, KDD, ICML, AAAI