Machine Learning in IR: Recent Successes and New Opportunities

ICML-2009 Tutorial

Paul Bennett (pauben@microsoft.com)
Misha Bilenko (mbilenko@microsoft.com)
Kevyn Collins-Thompson (kevynct@microsoft.com)

Microsoft Research

Tutorial Goals

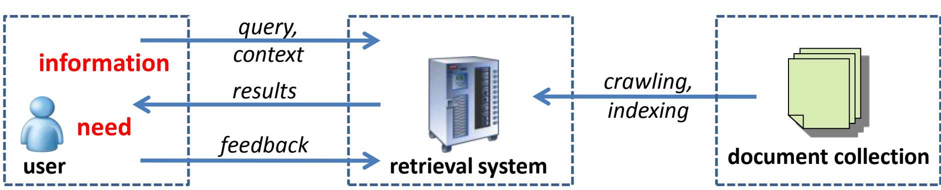
- Provide insight into core ML problems in IR
- Survey recent high-impact ML contributions to IR
- Highlight areas with promising opportunities for ML

Tutorial Overview

1. IR: Background and Challenges for Learning
2. Recent Advances at IR-ML Crossroads
   – Modeling relevance
   – Learning from user behavior
   – Learning to rank
3. Emerging Opportunities for Learning in IR
   – Online advertising
   – Risk-reward tradeoffs for retrieval algorithms
   – Learning complex structured outputs
4. Summary and Bibliography

IR Overview

- Basic IR paradigm: satisfying users' information needs



- Industry-defining applications: search, advertising, recommenders
- Major research areas
  – Modeling and estimating user intent
  – Processing and modeling information from documents
  – Selecting and ranking relevant results, incorporating feedback
- Core IR problems are *modeling and prediction tasks*

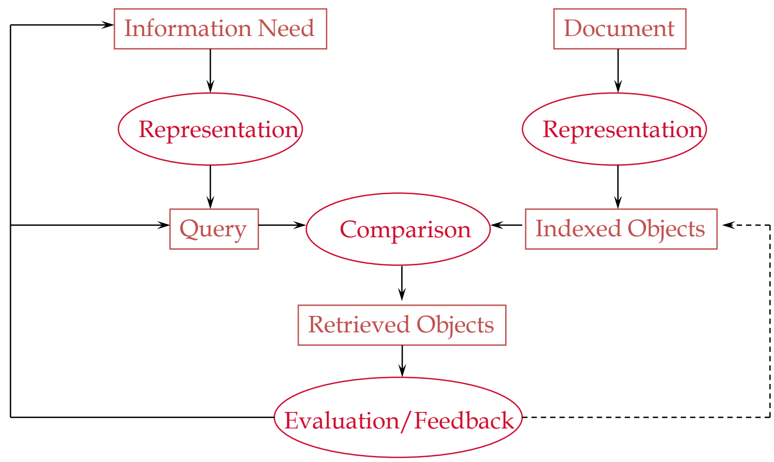
IR Increasingly Relies on ML

- Classic IR: *heuristics* that capture query-document similarity
  - TF-IDF, BM25, Rocchio classification, …
- Last 15 years: using evidence sources *beyond document text*
  - Document structure: hypertext properties, named entity extraction, …
  - Collection structure: annotation of in-links (anchor text), authority, …
  - User behavior data: from past clicks to browsing patterns
- Query and document models are becoming increasingly complex
  - Language, structure, relations, user behavior, time, location, ….
  - Rich applications for generative, discriminative and hybrid approaches
- Heuristics cannot scale, ML is the obvious solution

IR: Cornucopia of ML Problems

- *Classification*: content/query categorization, spam detection, entity recognition, …
- *Ranking*: result selection and ordering
- *Clustering*: retrieval result organization, user need segmentation
- *Semi-supervised learning*: unlabeled data is omnipresent
- *Active learning*: ranking, recommenders
- *Multi-instance learning*: image retrieval
- *Reinforcement learning*: online advertising

Basic IR Processes

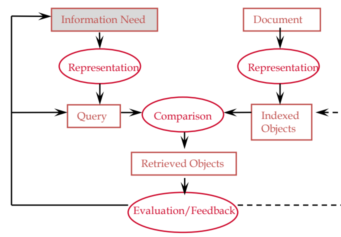


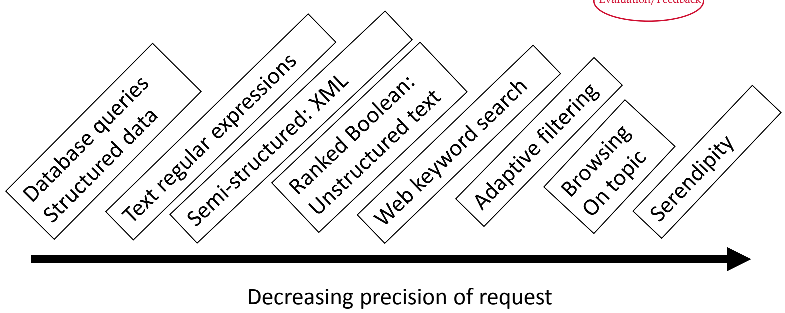
Characteristic IR challenges

- Uncertainty: task, topic, relevance, resources
- Scale: feature space, size, speed tradeoffs
- Evaluation and Feedback: user satisfaction
- Temporal: freshness and drift
- Adversarial: spam and security

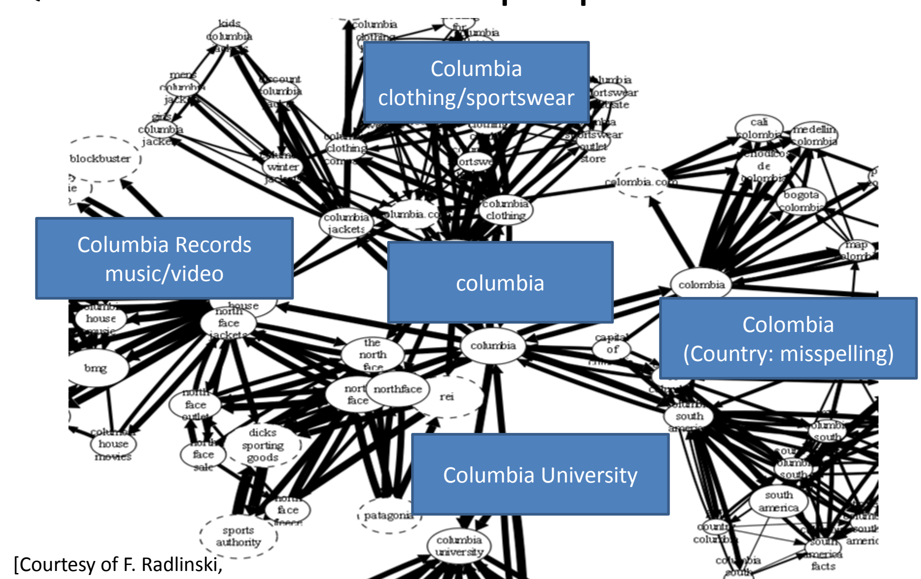
IR challenge: task variation
From precise lookup to random browsing

- Users may not know <u>how</u> to ask for what they need
- Or even <u>what</u> they need…
- Ambiguous intent



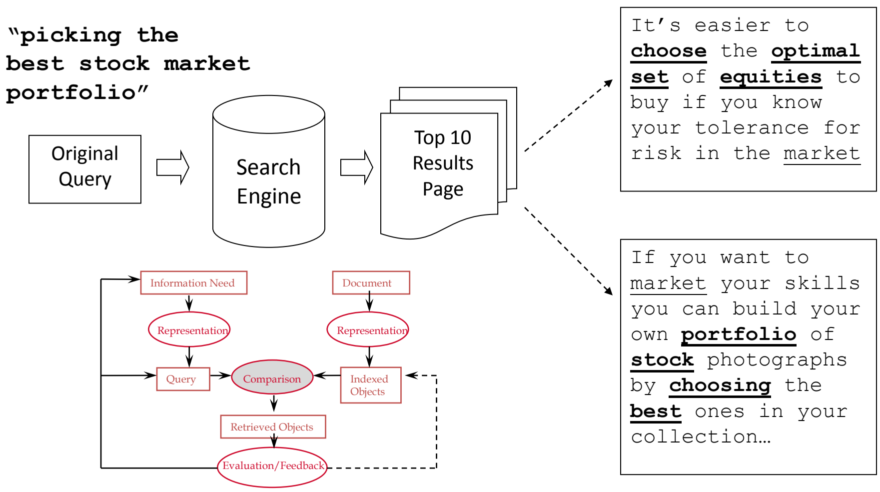


Queries can have multiple potential intents



[Courtesy of F. Radlinski, MSR Cambridge]

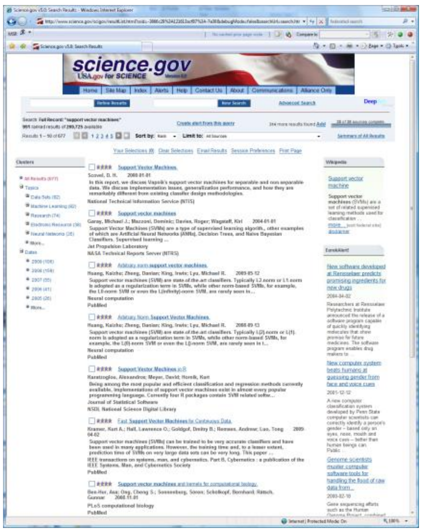
IR challenge: What if query and document terms don’t match? Or match incorrectly?



How can we formalize this vague notion of ‘relevance’ for learning algorithms?

- `System-oriented’ relevance:
  - Overlap in representations of Q and D
- But simple overlap ignores many important factors, such as:
  - Preferences and prior knowledge of user who issued request
  - Task that prompted the request
  - Other documents in collection
  - Previous queries of this or other users
  - External information on the (non) relevance of D
- Mizzarro [1997] surveyed 160 different formulations of relevance for IR tasks

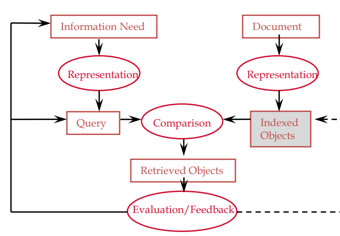
What if information is distributed across many sources?



- Many data sources may be hidden or unavailable to standard Web crawlers
- Not all sources may be co-operative
- Information sources may all be within the same organization or even same search system (tiers, index partitions)

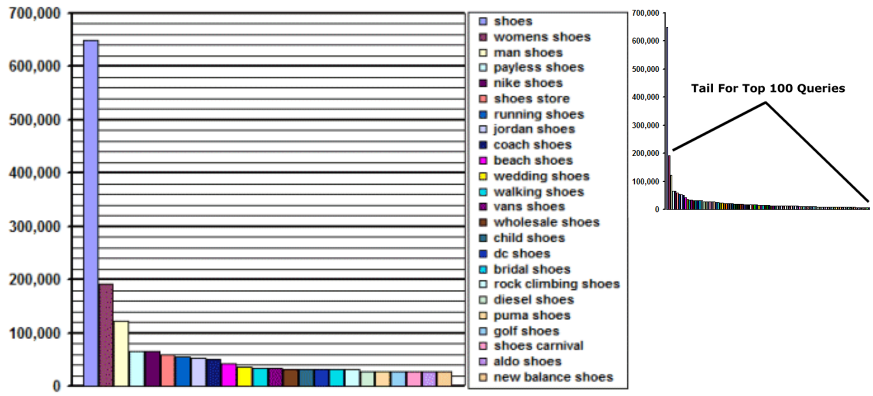
- Science.gov searches 38 databases and 1,950 selected websites.
- 200 million pages of U.S. gov't scientific information, e.g.
  - PubMed
  - NASA Technical Reports
  - National Science Digital Library
  - National Tech. Info. Service

IR challenge: Multiple resources



- How to learn what's in a resource?
  - Query-based sampling [Callan 2000]
- Learning which resources are best for a given query
  - Resource selection [Si 2004]
- There is a cost for accessing a resource
  - Learning when NOT to access a resource
- Merge results returned by different searches
  - Metasearch: learning how to calibrate & combine [Aslam & Montague 2001]
  - Information extraction and integration: Extract relevant name from one place, relevant location from another, … [Neves, Fox, Yu 2005]

The 'long tail' of a search log



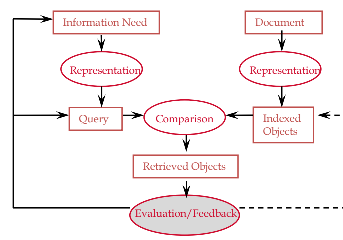
Danny Sullivan, Search Engine Watch, Sep. 2, 2004. http://searchenginewatch.com/3403041

IR challenge: scale

- Must scale over users/collections/dimensionality
- High throughput, real-time requirements of online systems: test time vs training time complexity
  - e.g. typical 250ms cutoff, with timeouts for subsystem dependencies much shorter.
- Huge number of potential features
  - Unstructured data
  - Ambiguity, subtlety, complexity of human language

IR challenge: Evaluation, ground-truth and feedback uncertainty

- Uncertain/noisy evidence:
  - Implicit feedback
    - Click data, user behavior
  - Pseudo-relevance feedback
  - Explicit feedback
    – "Find similar", "More like this"
- Formal relevance assessments
  – Missing or limited data, assessor disagreement
- Covered in detail later for evaluation and user modeling



IR challenge: Adversarial issues

- Continuous, evolving `war' between providers and spammers
- Search: Artificial ranking increases to attract visitors
  - Link farms [Eiron, McCurley, Tomlin 2004; Du, Shi & Zhao 2007]
  - Keyword stuffing [Ntoulas, Najork, Manasse & Fetterly, 2006]
  - Cloaking and redirection [Wu and Davison 2005]
- Ads: aggregators, bounce rate [Sculley et al. 2009], click bots
- Majority of issues at crawl & index time

IR challenge: Temporal issues

- Web is dynamic: keeping pace with changing content, siz e, topology, and use
  - Freshness [Lewandowski 2008]
  - Modeling page updates [Adar et al. 2009] and user revisitation [Adar, Teevan, Dumais 2008]
  - Crawling strategies must optimize for multiple goals, including:
    - Optimize allocation of bandwidth, computing resources
    - Re-visitation frequency for freshness
    - Politeness
    - Parallelization: coordinating distributed crawlers

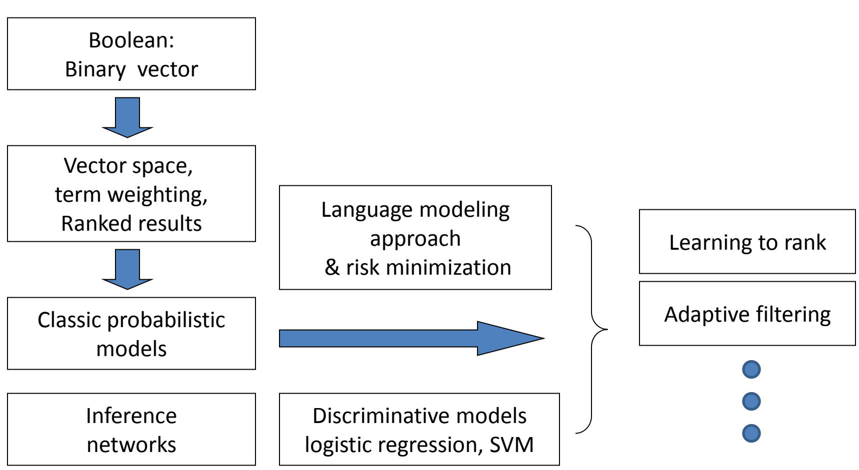
Tutorial Overview

1. IR: Background and Challenges for Learning
2. Recent Advances at IR-ML Crossroads
   – Modeling relevance
   – Learning from user behavior
   – Learning to rank
3. Emerging Opportunities for Learning in IR
   – Online advertising
   – Risk-reward tradeoffs for retrieval algorithms
   – Learning complex structured outputs
4. Summary and Bibliography

Outline: Modeling relevance

- Background on text representation and probabilistic retrieval models
- Generative *vs.* discriminative methods
- Focus applications:
  - Language modeling for retrieval, query model smoothing
  - Query performance prediction
  - Adaptive filtering

Highly simplified summary of IR retrieval model development



Text representat n: heuristic tf.idf weights combine freq d informativeness

idf: Rare/informative

- Each term i in docume *tf.idf* weight

tf: Frequent

$$= tf_{i,d} \cdot \log \frac{N}{df_i}$$

$tf_{i,d}$ = frequency of term i in document d

N = total number of documents in collection

df_i = the number of documents that contain term i

- Increases with the number of occurrences *within* a doc
- Increases with rarity of the term *across* the whole corpus

- tf.idf approximates a Fisher kernel for the Dirichlet Compound Multinomial [Elkan 2005].

Combining tf.idf with the classical vector space model

- Vector space scoring function is very general:

$$sim(u,v) = \cos\theta = \frac{\vec{u} \circ \vec{v}}{|\vec{u}||\vec{v}|} = \frac{\sum_{i=1}^{n} u_i \cdot v_i}{\sqrt{\sum_{i=1}^{n} u_i^2}\sqrt{\sum_{i=1}^{n} v_i^2}}$$

- Retrieval becomes a k-nearest-neighbor problem in a high-dimensional feature space
- Relevance is measured by distance from query

Probabilistic IR methods provide a principled foundation for reasoning under uncertainty

- Underlying problems:
  1. Ranking documents
  2. Traditional IR: Doc/query matching is semantically imprecise.

Can we use probabilities to quantify our uncertainties?

- Step 1: Assign probability of relevance to each document
- Step 2: Rank documents: highest probability get highest rank
- We observe a user's query Q, and often not much else, in addition to document D
- Probability Ranking Principle: rank documents *in order of probability of relevance* to the information need

Classical probabilistic retrieval model

[Robertson & Sparck-Jones 1976]

- Treats retrieval as a kind of Naïve Bayes classification problem with relevant/non-relevant classes
- Binary independence model (BIM):
  - Only presence/absence of terms is used: no term frequency
  - Terms are treated as independent

1. Assign feature weights to query terms
   - How does each term contribute to relevance?
2. Score documents
   - Add weights ("votes") of the query terms it contains

$$w_i = \log \frac{(r_i + 0.5)(N - R - n_i + r_i + 0.5)}{(R - r_i + 0.5)(n_i - r_i + 0.5)}$$

[n_i , r_i]= count of [docs, relevant docs] containing term t_i

Okapi: Adding term frequency via the two-poisson model

- Two-poisson: A document is 'about' a concept (term) or not
  - 'Elite' terms are terms that the document is about
  - Replace presence/absence with query-term eliteness
    - Eliteness isn't known directly but can be estimated from statistical models
- Okapi / BM25 weighting:
  - One of the most effective current weighting schemes
  - Estimate eliteness weights from observed term counts
    - RSJ weight, TF factor, correction for document length

BM25

$$\sum_{t \in Q,D} \ln \frac{N - df + 0.5}{df + 0.5} \times \frac{(k_1 + 1) \cdot tf}{k_1(1-b) + b \cdot (\frac{dl}{avgdl}) + tf} \times \frac{(k_3 + 1) \cdot qtf}{k_3 + qtf}$$

$$k_1 \in [1.0, 2.0],\ b \text{ usually } 0.75,\ k_3 \in [0, 1000]$$

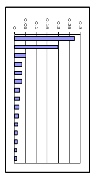
"k_1, b and k_3 are parameters which depend on the nature of the queries and possibly on the database; k_1 and b default to 1.2 and 0.75 respectively, but smaller values of b are sometimes advantageous; in long queries k_3 is often set to 7 or 1000 (effectively infinite)"
[Robertson and Walker, 1999].

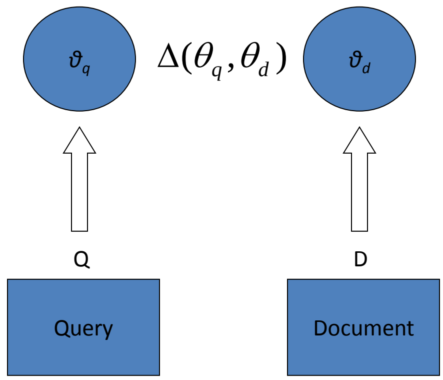
The Language Modeling Approach to Information Retrieval

- Queries and documents are samples from language models whose parameters must be estimated

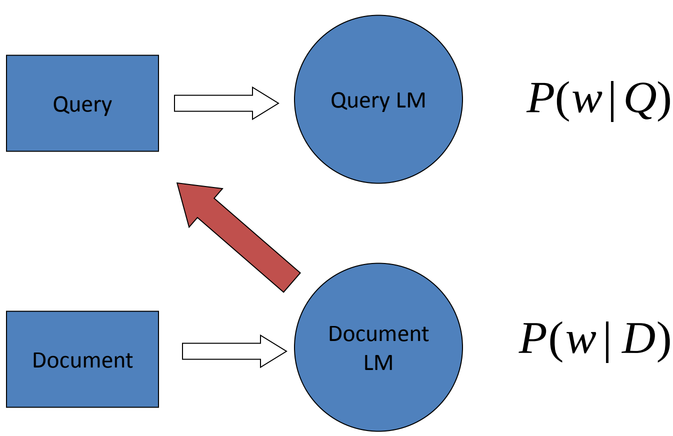
w	$p(w \mid \mathbf{q})$
virus	0.275
ebola	0.197
hoax	0.051
viruses	0.034
outbreak	0.034
fever	0.033
disease	0.024
haemorrhagic	0.023
gabon	0.022
infected	0.019
aids	0.016
security	0.014
monkeys	0.013
hiv	0.011
zaire	0.011

$\mathbf{q}$ = ebola virus (Web)

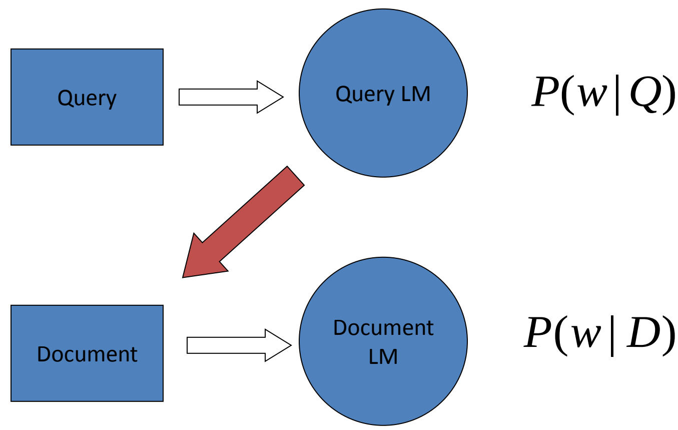




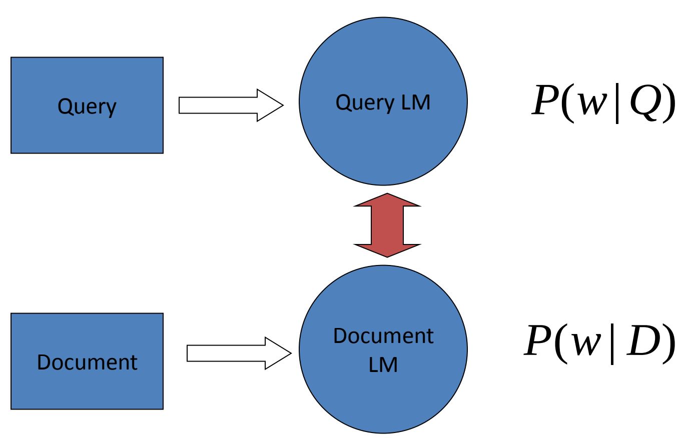
Language model retrieval: Query likelihood



Language model retrieval: Document likelihood



Language model retrieval: Model comparison



Regularization in the LM approach

[Zhai & Lafferty 2001]

- Goal: Provide estimates for missing or rare terms.
- Approaches are constrained by efficiency for retrieval

Jelinek-Mercer: Interpolate MLE $P(t \mid M_d)$ with collection MLE

$$p_s(w \mid d) = \lambda \cdot p(w \mid d) + (1-\lambda)\, p(w \mid C)$$

Dirichlet: Conjugate prior for multinomial distribution

$$p_s(w \mid d) = \frac{c(w,d) + \mu \cdot p(w \mid C)}{|d| + \mu}$$

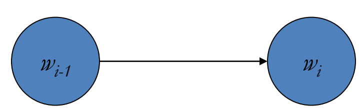
Two-stage: JM + Dirichlet for both short and long queries

$$p_s(w \mid d) = (1-\lambda)\frac{c(w,d) + \mu \cdot p(w \mid C)}{|d| + \mu} + \lambda\, p(w \mid C)$$

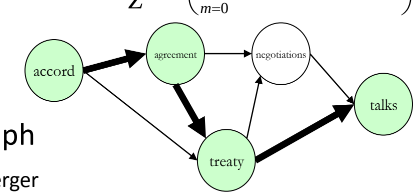
- These ignore dependencies between terms

Semantic smoothing: Exploiting semantic dependencies between words or phrases

- Relations between two terms are defined by link functions
- Link functions $\lambda_1 \ldots \lambda_m$, mixture parameters ϑ_m
  - Synonyms
  - Morphology (Stems)
  - Free association data
  - Co-occurrence
    - Top documents
    - External Web corpus
- Random walk on translation graph
- Statistical translation models [Berger & Lafferty 1999]



$$p(w_i \mid w_{i-1}) = \frac{1}{Z}\exp\left(\sum_{m=0}^{L} \theta_m(i)\lambda_m(w_i, w_{i-1})\right)$$



[Collins-Thompson & Callan, 2005]

Document Scoring via Bayesian Decision Theory

Loss function

Query model uncertainty

Document model uncertainty

Special c… …nt loss

$$r(d \mid q, C, U, \vec{S}) = \int_{\theta_Q} \int_{\theta_D} \underline{\Delta(\theta_Q, \theta_D)}\, \underline{p(\theta_Q \mid q, U)}\, \underline{p(\theta_D \mid d, \vec{S})}\, d\theta_D\, d\theta_Q$$

Handling <u>topic uncertainty</u>: <u>expectation</u> over all possible models
Handling <u>relevance uncertainty</u>: specify concrete loss function

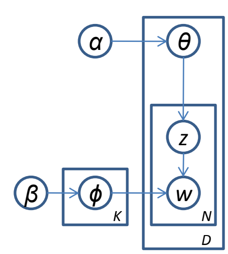
Document sources *S* for documents in collection *C*, user *U*. Present a subset of documents *D.*

Source: C. Zhai, Risk Minimization and Language Modeling in Text Retrieval, PhD Dissertation, CMU, 2002.

Latent Dirichlet Allocation [Blei, Ng, Jordan. 2001] and other generative topic models

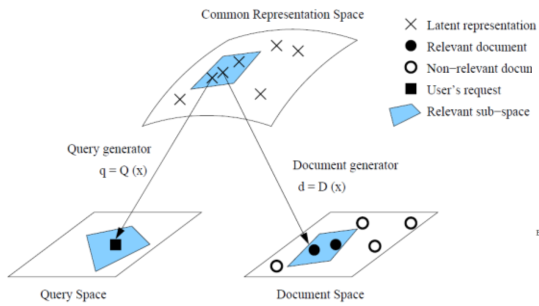
- Specify *K*: number of topics, D: docs in corpus
- Learning *α, β* gives information about the corpus:
  - *α* : Semantic diversity of docs
  - *β* : How similar topics are
  - *ϑ* : Prob. of each topic in each document
- Topics and words can vary in generality
- LDA: Bursty in topics, but not in words
- Hybrid: DCM-LDA [Doyle & Elkan, ICML 2009]
  - Captures topic + word burstiness
- Advantage: Reasonable assumptions, interpretable parameters
- Disadvantage: Not good at handling outliers

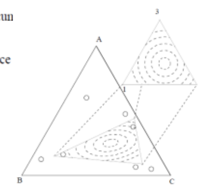
$\vartheta \sim$ Dirichlet(α)
$\phi \sim$ Dirichlet(β)
$z \sim$ Multinomial(ϑ)
$w \sim$ Multinomial(ϕ)



Generative Relevance Model

[Lavrenko 2004]





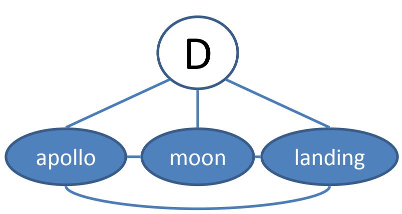
LDA

GRM

- Differences from LDA:
  - Effective at capturing outliers / rare events
  - Few structural assumptions, easy & efficient to train (Free parameter λ controls estimator variance)
- Beyond text: Learn models of text + image correspondence via shared relevance/semantic space

Markov Random Field retrieval scoring

[Metzler & Croft 2005]



- Undirected graphical model
- Edges capture dependency assumptions
- Arbitrary features
- Linear scoring function
- Prefers documents containing features that reflect dependencies present in query

$$P_{G,\Lambda}(Q,D) = \frac{1}{Z_\Lambda}\prod_{c\in C(G)}\psi(c;\Lambda)$$

$$P_{G,\Lambda}(Q,D) \overset{rank}{=} \sum_{c\in T_{QD}}\lambda_c f_c(c)+\sum_{c\in O_{QD}}\lambda_c f_c(c)+\sum_{c\in U_{QD}}\lambda_c f_c(c)+\sum_{c\in D}\lambda_c f_c(c)$$

Individual terms

LM or BM25 weights

Ordered query terms

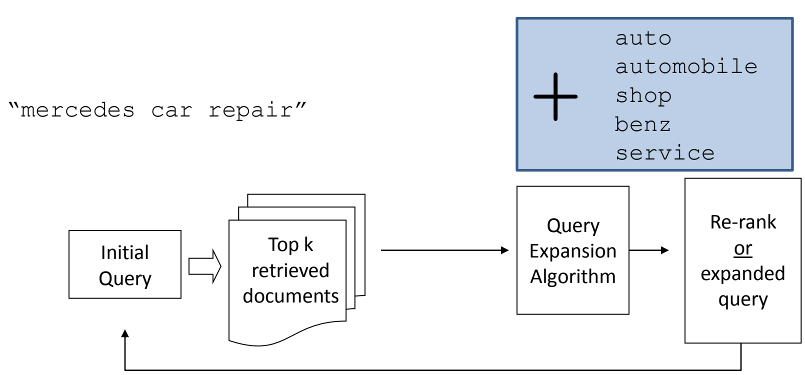
Unordered query terms

Document

Discriminative and hybrid models

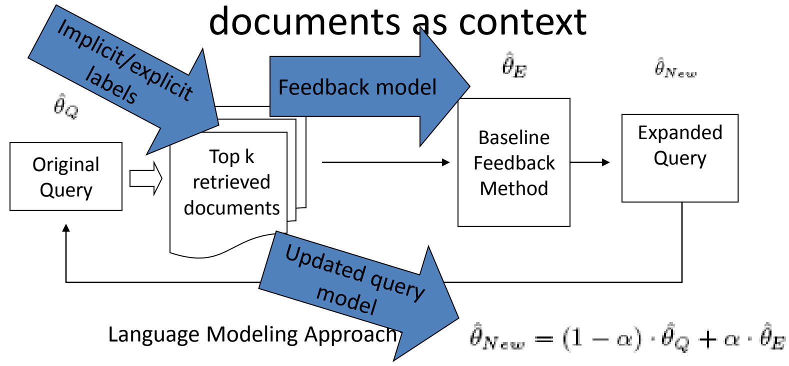
- Logistic regression [Gey 1994]
- Linear feature-based models
  - Linear discriminant model [Gao et al. '05]
  - MaxEnt [Cooper 1993, Nallapati 2004]
  - Markov Random Field model [Metzler and Croft '05]
- Challenge: many negative, few positive examples
- Learning methods
  - Direct maximization [Metzler and Croft 2007]
  - Perceptron learning [Gao et al. 2005]
  - RankNet [Burges et al. 2005]
  - SVM-based optimization
    - Precision at *k* [Joachims 2005]
    - NDCG [Le and Smola 2007]
    - Mean Average Precision [Yue et al. 2007]

Goal: Use <u>context</u> to learn a more complete representation of the user's information need



<u>Applications</u>:
- Query expansion and alterations
- Applying search personalization for re-ranking
- Matching short text snippets (queries, ads, …)

Learning from evidence related to top-ranked documents as context



- Unsupervised learning: implicit relevance (pseudo-feedback)
- Semi-supervised learning methods [Zhu 2006]
  - Propagate explicit document labels through similarity graph
- Image retrieval: selective use of very large numbers of features, on-line learning
  - e.g. Boosting [Tieu and Viola, 2000]

Query performance prediction

Given query, collection and possibly initial results predict:

1. Query difficulty: what is likely precision of the top-*k* docs?
   - Work harder or involve user if poor results predicted
2. Resource selection: When is a collection NOT likely to satisfy a query?
   - Federated search: save access costs, reduce noise
3. Expansion risk: When is query expansion likely to be effective?
   - Big win if we could accurately predict when and how to perform for any given query

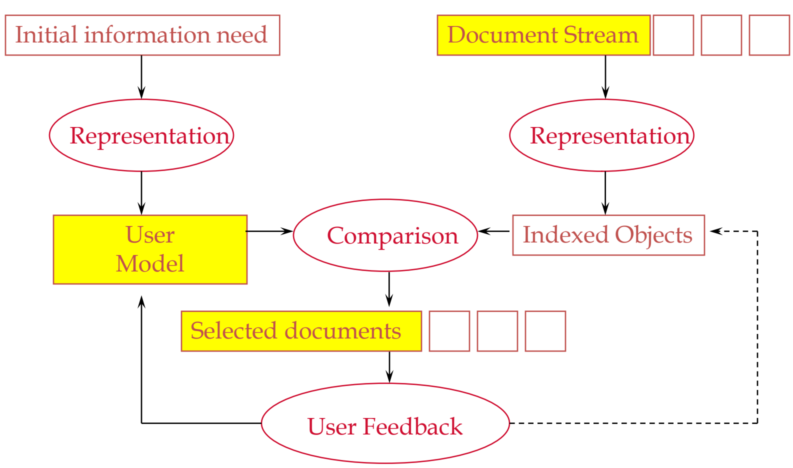
Learning to predict query difficulty

- Classifier using features based on agreement between result sets from initial query and subqueries [Yom-Tov et al. 2005]

- Pre-retrieval predictors [He & Ounis 2004]
- Query clarity: divergence of top-ranked LM from general collection [Cronen-Townsend, Zhou, Croft 2004]
- Sensitivity to query and document perturbation [Vinay et al. 2006]
- Divergence of multiple scoring functions [Aslam & Pavlu 2007]

- Typical Kendall-tau with average precision: 0.10 - 0.50
- Promising early results, but further improvements needed
- Core problems:
  - Estimating prediction confidence
  - Selective allocation of computing resources

Adaptive filtering system



Adaptive filtering systems require more dynamic retrieval & user models

- Traditional IR systems:
  - Relatively static collection, ranking
- Filtering systems:
  - Handle a dynamic stream of new documents, and make yes/no decisions about when to alert user to important new information
  - Based on implicit or explicit feedback
  - Evolving user profile which is updated frequently
  - Exploration vs exploitation (active learning)
- Evaluation: TREC Filtering Track with adaptive filtering task
- Early systems [Survey: Faloutsos & Oard, 1995]
  - Exemplar documents create an implicit standing query
  - New documents treated as queries, compared against exemplar
- Problem: Learn user profiles efficiently from very limited data.

Adaptive filtering: Active learning

[Zhang 2005]

With existing training data D = {(x_1, y_1), …, (x_k, y_k)} with scores y_i, labels y_i

Exploitation: Make the user happy now:

$$U_1(x \mid D) = \int_\theta \sum_y A_y \cdot p(y \mid x, \theta)\ p(\theta \mid D)$$

Exploration: Ask user for feedback now to increase future happiness:

$$U_2(x \mid D) = \sum_y p(y \mid x, D) \cdot Loss(D \cup \{x, y\}) - Loss(D)$$

Overall utility combines both:

$$U(x \mid D) = U_1(x \mid D) + n_{FUTURE} U_2(x \mid D)$$

$$\text{Deliver to user if } U(x \mid D) \geq 0$$

Adaptive filtering : Bayesian framework [Zhang 2005]

- Constrained MLE: integrate expert heuristic algorithm (Rocchio) as Bayesian prior for logistic regression
  - Find Rocchio decision boundary
  - Prior: Find LR MLE with same decision boundary as Rocchio
- Model complexity controlled by amount of training data
- Better than either Rocchio or logistic regression alone
- Beyond relevance:
  - Novelty, readability, authority [Zhang, Callan, Minka 2004]

Tutorial Overview

1. IR: Background and Challenges for Learning
2. **Recent Advances at IR-ML Crossroads**
   - Modeling relevance
   - **Learning from user behavior**
   - Learning to rank
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs for retrieval algorithms
   - Learning complex structured outputs
4. Summary and Bibliography

A Machine Learning Approach

- Get some data labeled with the ground truth
  - Force the user to give feedback?
  - Expert Judges?
  - Implicit Feedback?
- Train a model
- We're done
  - Better Performance?
    - More data
    - New features
    - New learning algorithms
  - Iterate until performance reaches desired level

IR's Focus on the User

- The user is central in information retrieval.
- Evaluation Design
  - Construct hypothesis about what matters to the user.
  - Formulate a way to test hypothesis.
  - Study users to find where hypothesis breaks down.
- Getting at user satisfaction requires revision of data-driven performance metric as well as features and models.

IR + Machine Learning (Better Performance)

- Construct hypothesis about *how to predict* what matters to the user.
- Formulate a measure to optimize.
- Train a model.
- Look for errors in model -> improve model.
- Look for mismatch in measure -> target new measure, design new approach

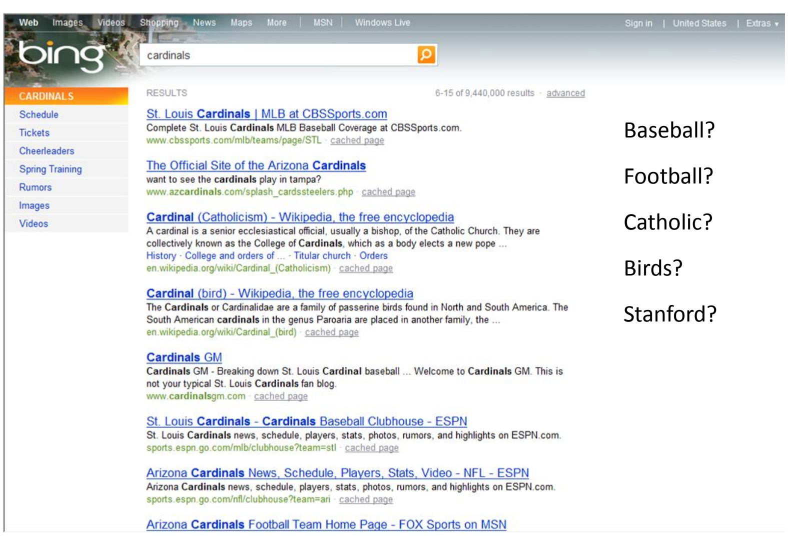
IR + Machine Learning for Data Mining (Better ground truth, features)

- Construct hypothesis about what matters to the user.
- Formulate a measure to optimize.
- Formulate a hypothesis regarding connection between data and measure to optimize.
- Mine for patterns that match hypothesis -> add as feature for ranker, convert to ground truth
- Mine for patterns that violate hypothesis. -> target new measure
- This section will present a series of examples focused on web search that fall into these paradigms. General lessons apply to any IR task.

Hypothesis: Search is simply many classification tasks.

- Each information need is really a “concept” as in standard machine learning.
- For each concept, some items are relevant and others are not relevant.
- We know how to approach this:
  - Take query, document pairs and give them to a human relevance expert to label them as relevant and not relevant to the query.
  - Optimize a measure of accuracy over these.

Ambiguous Queries



Baseball?

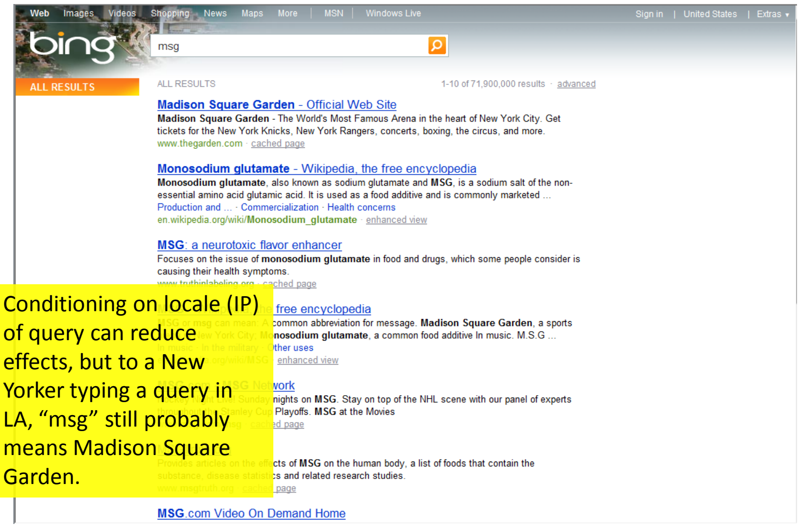
Football?

Catholic?

Birds?

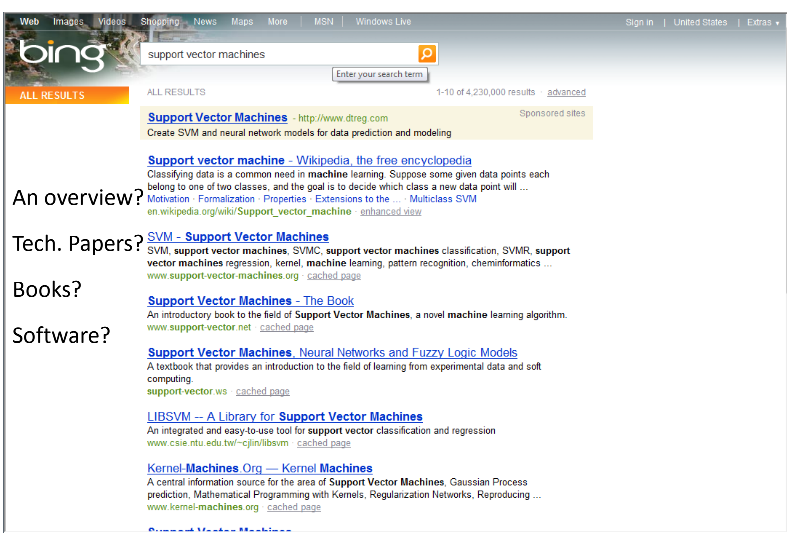
Stanford?

Locale & Ambiguity



Conditioning on locale (IP) of query can reduce effects, but to a New Yorker typing a query in LA, “msg” still probably means Madison Square Garden.

Ambiguity by Result Type



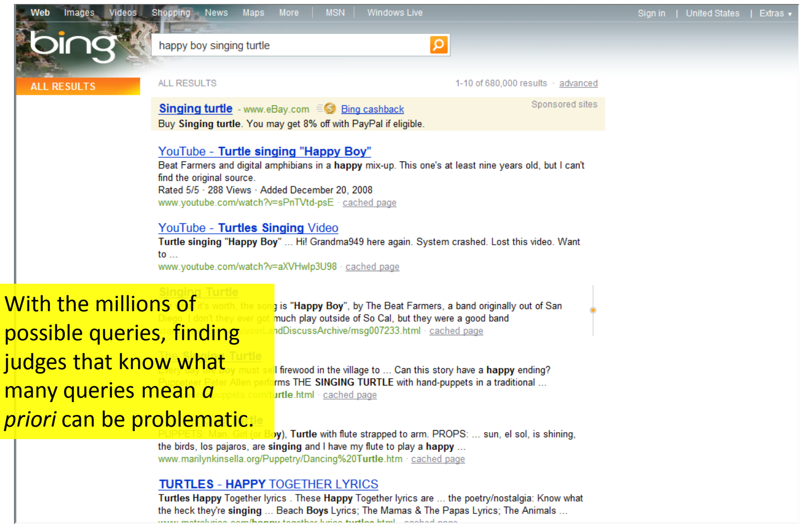
An overview?

Tech. Papers?

Books?

Software?

The Long Tail & Ambiguity



With the millions of possible queries, finding judges that know what many queries mean *a priori* can be problematic.

"Expert" Judging Issues

- Ambiguity – in many forms
  - A query is an ambiguous representation of an underlying information need. Only the issuer of a query knows the actual information need.
- "Relevance"
  - Not only do we need to know the information need, we need to know the user's definition of relevance for this query.
  - Topical? Authoritativeness? Quality? Reading Level? Conditional on other results (novel, diverse viewpoints)?

More Expert Judging Issues

- I think I can get experts trained closely enough to reflect the average user, but there's still …
- Calibrating judges
  - Want the interpretation of a score to be the same across queries.
  - Different judges for the same query
- New content
  - How are new documents judged for relevance on a query?
  - If judges are more likely to be consistent if all judging occurs at the same time for a given query, does new content mean relabeling all documents for that query?
- Changed Content
  - Documents on the web, desktop, intranet can change frequently.
  - Does relevance need to be rejudged every time content changes?

Current IR Collection-Building

- Which queries?
  - Sample from logs.
- How many queries?
  - The proportion of variance in estimated system performance attributable to differences in the query set vs. system differences is highly dependent on the number of queries (Carterette *et al.,* SIGIR 2008).
  - Make number of queries very big.
- Which documents?
  - Top by current system, *pooled* from several systems, top by content method (*e.g.* BM25), random
- Desire minimal labeling effort (cost) for ranking retrieval systems by performance or estimating performance
  - Carterette *et al.* (ECIR 2009) present overview and study of current methods.
  - Related to *active learning* for improving the system rather than evaluating itself. New developing area (Aslam *et al.,* SIGIR 2009).

Learning From User Behavior

- Okay, collection-building is hard. We care about users – so focus on that!
- Instead of explicit judgments, model or optimize for implicit measures using behavior (Kelly & Teevan, SIGIR Forum ‘05; Fox *et al.*, TOIS ‘05; White *et al.,* SIGIR ’05; Boyen *et al.,* IBIS@AAAI ‘96).
  - Queries, clicks, dwell time, next page, interactions w/browser
  - Session level: reformulations, abandonments, etc.
- Pros: behavior changes with content as well, user’s idea of relevance drives behavior, ton of data

Interpreting a Click

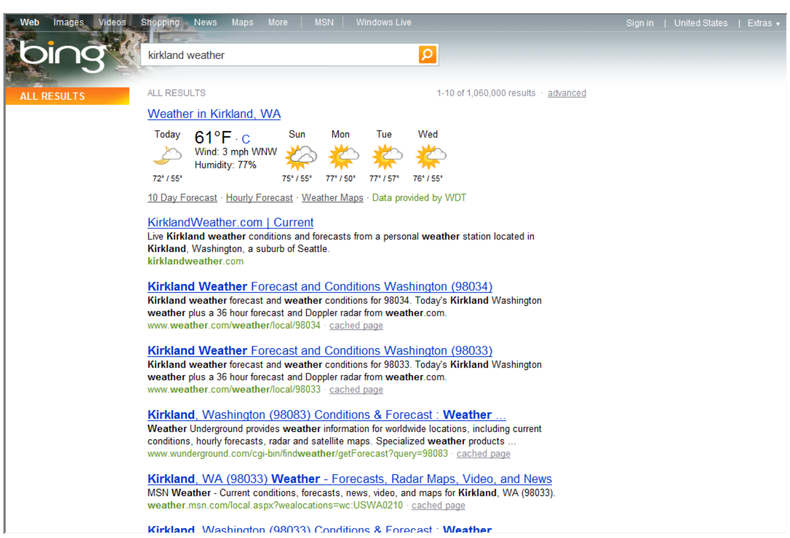
- Hypothesis: A click is a judgment that the clicked item is relevant.
- Rank Bias – the more highly ranked an item, the more likely it is to get a click regardless of relevance.
  - When order is reversed, higher ranked items still typically get more clicks. (Joachims *et al*, SIGIR ’05).
- Clicks are not an absolute judgment of relevance.
  - Although we can debias in various ways (Agichtein *et al.*, SIGIR ’06)
- Eye-tracking studies show users tend to have seen at least everything above a click and perhaps a position below it (Joachims *et al*, SIGIR ’05).
- Hypothesis: A click is a preference for clicked item to all those above and one below it.

Modeling Clicks as Preferences

- Click > Skip Above, Click > Earlier Click, Click > Skip Previous, Click First > No-Click Second
- Reversing the ranking satisfies many derived preferences.
- Add constraint that weights learned by ranking SVM are positive (higher minimum value limits ranking to diverge more slowly from original ranking).

[Radlinski & Joachims, KDD ’05]

No Click → Not Relevant?



Click ⟶ Relevant?

Other Common Kinds of User Behavior

- Abandonment – user does not click on a search result.
  - Usually implies irrelevant?
- Reformulation – Users may reformulate a new query instead of clicking on a lower relevant result.
  - Reformulation implies irrelevant?
- Backing Out – Users may go back to the search page and click another relevant result.
  - Last click is most relevant?
  - Information gathering queries?

Last Click as Relevance Vote

- Hypothesis: user continues until they find a document that satisfies their needs.
- Goal: separate click as relevance from presentation order effects.
- Predict clicks on urls presented in order B,A when trained from A,B order (Craswell *et al.*, WSDM '08)
- Possible explanatory models
  - Baseline – symmetric probability
  - Mixture – click with probability based on relevance or blind clicking based on rank
  - Examination – examine with probability based on rank and if examined, click with probability based on relevance
  - Cascade – Click on document, *d*, based on probability of relevance, r_d, and continue with next lower document with probability, $(1 - r_d)$.
- Active area is extending simplified assumptions of Cascade model.

[Craswell *et al.*, WSDM '08]

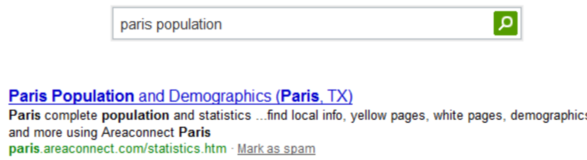
Online Learning to Optimize Rankings

- Goal – minimize abandonments.
- Online learning for repeated queries.
  - Run *k* multi-armed bandits.
  - The *k*th one is responsible for determining value of each document at *k*th position given chosen above.
  - If click on position *k*, *kth* MAB gets payoff to update values.
  - Computing OPT offline is equivalent to set cover and is NP-hard.
  - Bounds get $(1 – 1/e)$ OPT – sublinear(T)
- Assumptions of a single click on first relevant item and that a click always occurs when a relevant item is displayed.

[Radlinski *et al.*, ICML '08]

Risking Brand

- Should you display potentially irrelevant items to determine if they are relevant?

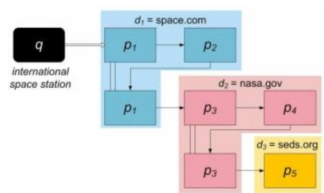


- Everything is fine until someone ends up with a honeymoon in Paris, TX.
- More importantly, displaying irrelevant items runs the risk of lowering user perception of the search engine's overall quality.
- Potentially more susceptible to spamming as well.
- Could use as a technique to collect a gold standard ranking.
- Open Area
  - Models that learn risk and reward and integrate that into a risk/reward tradeoff framework.
  - Identifying/Predicting low risk scenarios for exploring relevance.
    - Simple one is when predicted query performance is low.

Session Information

Sessions and Browsing

- Clearly, a click for a single query is too short term.
- Use overlap in queries from the same session, clicked results, *etc.* to build a lightweight profile of the user's current goal.
  - Relational learning approach to tailoring next query's results based on earlier queries (Mihalkova & Mooney, ECML '09).
- Mining Browsing Patterns (Bilenko & White, WWW '08)
  - A user browses to other relevant pages starting with pages reached from a query.
  - Use that browse path to infer relevance to the original query.



Personalization

- The same query means different things to different people.
- The same results therefore have different relevance value to two issuers of the same query.

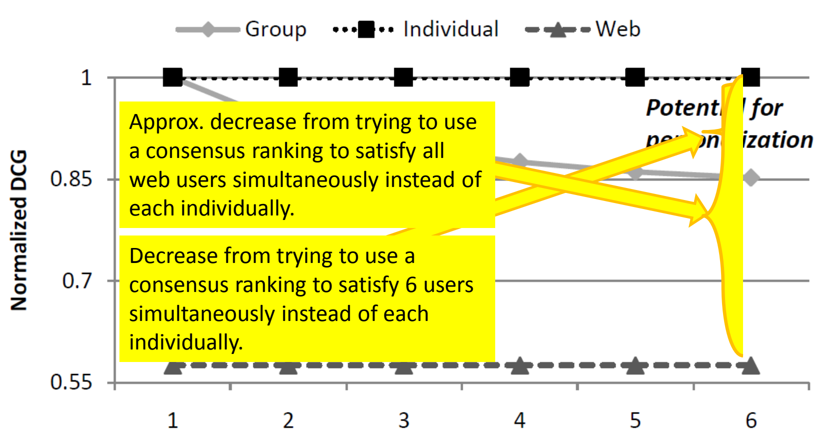
- Hypothesis: many forms of ambiguity would disappear if we could condition on the user.

Two Views of Relevance for One Query

Web Result	Gain A	Gain B	A+B
usa.canon.com/consumer/controller?act=ProductCatIndexAct&fcategoryid=111	1	0	1
cameras.about.com/od/professionals/tp/slr.htm	1	1	2
cameras.about.com/od/camerareviews/ig/Digital-SLR-Camera-Gallery/index.htm	0	1	1
amazon.com/Canon-Digital-Rebel-XT-f3-5-5-6/dp/B0007QKN22	0	0	0
amazon.com/Canon-40D-10-1MP-Digital-Camera/dp/B000V5P90K	0	0	0
en.wikipedia.org/wiki/Digital_single-lens_reflex_camera	1	0	1
en.wikipedia.org/wiki/DSLR	1	2	3
olympusamerica.com/e1/default.asp	0	0	0
olympusamerica.com/e1/sys_body_spec.asp	0	0	0
astore.amazon.com/photograph-london-20	0	0	0
	User A	User B	Avg
Normalized DCG	0.52	0.23	0.38

From Teevan *et al.* (2009)

The Improvement Possible via Personalization



From Teevan *et al.* (2009)

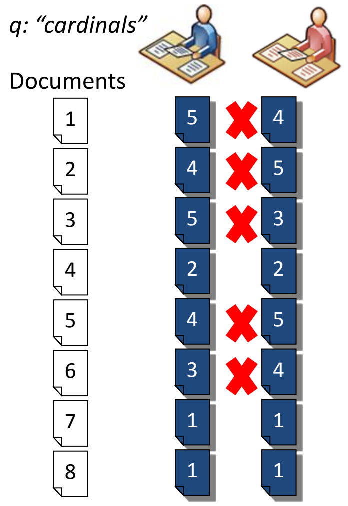
Predicting when to Personalize

- Personalization can help significantly, but when should it be applied?
  - All the time?
    - Data sparsity challenge for building a profile to cover all queres.
    - Often people search "outside" of their profiles.
  - When the query matches the user's profile?
    - How should the profile be built? Topically? Demographic? Locale?
  - What types of models are best for identifying what properties of users, queries, and results should be used to tie parameters?
- Predicting when to personalize is likely to have a high payoff if done with a high accuracy.
- Early results indicate reasonable accuracy can be attained via machine learning (Teevan *et al.*, SIGIR 2008).
- Open area for machine learning researchers to contribute more methods and approaches.

Towards a Better Ground Truth

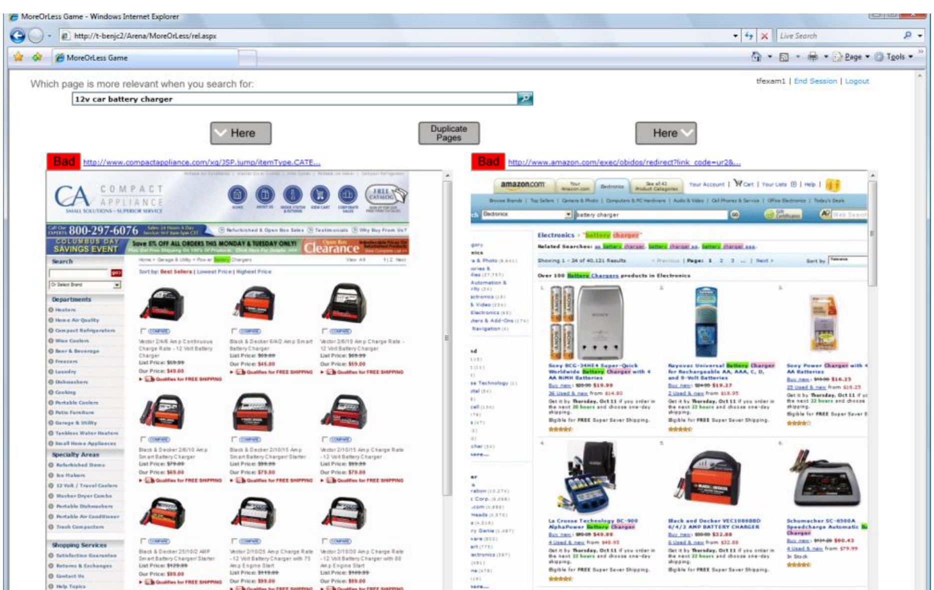
- Many problems have to do with the ambiguity that arises between an information need and its representation.
  - Allow more expressive queries.
  - Give the judges more context.
- Some disagreements in judging might be due to noise.
  - Get better judgments with less noise.
- A search engine is used by many users and not just one. So the real problem is to get a consensus ranking.
  - More (cheaper) judgments to average out individual views of relevance and determine a consensus.
- Many problems come from asking an "expert" instead of the user that issued the query.
  - Elicit feedback from the user by making it have a higher payoff (e.g. personalization) or lower cost for the user.

Expert Judge Disagreement



- Different Labeling Processes
  - Noise that arises from cognitive process and not "true disagreements".
  - Typically a tradeoff between amount of information from a label task and complexity of the task (increased cost and noise).
- Ranking instead of absolute labels?

Pairwise Preferences



What kind of label?

- Binary relevance
  - Most well-studied and understood – especially when relevance of documents is independent from each other.
  - Can fail to capture important levels of distinction to a user.
- Absolute degrees of relevance (Järvelin & Kekäläinen, SIGIR '00)
  - Provides distinction lacking in queries.
- Preferences (relative degrees) (Carterette *et al.,* ECIR '08)
  - More reliable and can assess quality of ranking for a given query but lacks distinction between queries where system performs well (best result is awesome) and those where performance is poor (best result is horrible).
- Relevance by "nugget" aspects (Clarke *et al.,* SIGIR '08)
  - More fine-grained but unclear yet if approach is applicable at scale.
- Different label types provide opportunities for new and hybrid models.

The Human Computation Approach

- If relevance judgments are expensive, then find a cheaper way to get the same thing. Then get *MANY* of them to find consensus.
- ESP game (von Ahn & Dabbish, CHI '04) – Tagging images for indexing.
  - Useful for retrieval but not a relevance judgment (perhaps implied).
- Picture This (Bennett *et al.,* WWW '09) – Preference judgments for image search.
  - Actual relevance judgments given as relative preferences.
  - Relies on assumption that population of raters is drawn from same distribution as searchers.
- Use of human computation for relevance judgments.
  - How many times to relabel in context of Mechanical Turk (Sheng & Provost, KDD '08).
  - Selecting the most appropriate expert (Donmez *et al.,* KDD '09).

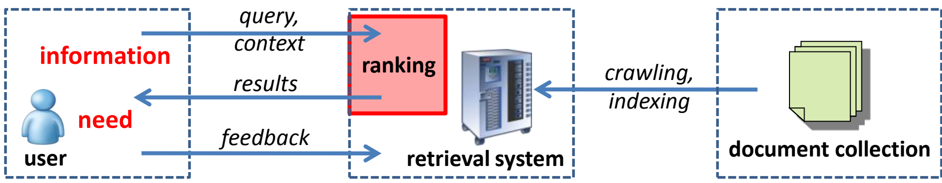
Learning from User Behavior Summary

- Reality -- use both implicit and explicit judgments as a source of information.
  - A common approach is explicit as ground truth and clicks as a feature.
  - Other approaches where optimization targets clicks, reformulations, abandonments, *etc*. (cf. Das Sarma *et al.,* KDD '08).
- Emerging models optimize joint criteria over both or the attention of a user (Dupret *et al.*, QLA@WWW '07; Chapelle & Zhang, WWW '09; Guo *et al.*, WWW '09).
- Primary lesson:
  - User interaction with a set of results is more structured than click as a vote for the most relevant item.
  - Opportunities for rich structured learning models and data mining.

Tutorial Overview

1. IR: Background and Challenges for Learning
2. **Recent Advances at IR-ML Crossroads**
   - Modeling relevance
   - Learning from user behavior
   - **Learning to rank**
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs for retrieval algorithms
   - Learning complex structured outputs
4. Summary and Bibliography

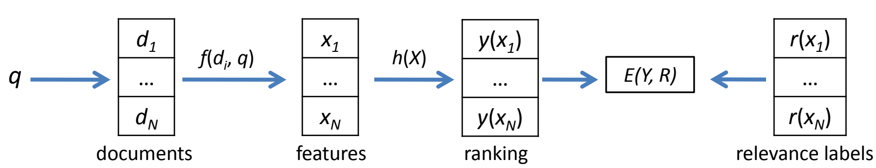
Ranking: Core Prediction Problem in IR



- Context-dependent vs. context-independent ranking
  - *Context-dependent*: relevance w.r.t. information need (query, page, ...)
    - Search result ranking, advertisement selection, frontpage headlines
  - *Context-independent*: absolute relevance (HITS, PageRank, etc.)
    - Static document ranking, crawl queue scheduling, index tiering

Ranking as a Learning Problem

- Given query q and document collection $\{d_1, \ldots, d_N\}$
  - Input: query-document instances $X=\{x_1, \ldots, x_N\}$, $x_i = f(d_i, q)$, $x_i \in \mathbb{R}^d$
  - Output: ranking $Y=\{y(x_1), \ldots, y(x_N)\}$: permutation of X by ranker $h(x)$
  - Evaluation (loss) function: $E(Y, R)$, $R=\{r(x_1), \ldots, r(x_N)\}$: true relevance of x_i



- (Semi-)Supervised Setting
  - Labeled data: query-document-(relevance) instances: $\{ (q,x,r(x)) \}$

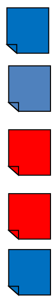
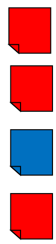
Practical Considerations (I)

- Features capture diverse evidence sources
  - *Query-document*: contents and metadata relevance (BM25, title, anchor, …)
  - *Document*: contents, link structure, popularity, age, …
  - *Query*: length, frequency, named entities, categories/topics, …
  - *Behavioral data*: historical information from logs (clickthrough, dwell time, …)
  - Transformations of all of the above
- Subset of documents to be ranked is provided by the index
  - Indexing must solve syntactic issues (spelling, stemming, synonymy)
- *Discriminative methods* are more appropriate due to strong feature correlations and unavoidable bias in training data

Practical Considerations (II)

- Exhaustive labeling is impossible: distribution is *always skewed*
- TREC: pooling = judges label all documents from each system
- Web: judges label all top-rated documents, plus some lower-ranked documents (e.g., sampled from candidate subset or web usage data)
- Labeling issues (covered earlier)
  - Ambiguity in user intent
  - Query sampling for dataset construction
  - Disagreements between judges
  - Use of implicitly labeled data (clicks, dwell times, query reformulations)

Ranking Evaluation: Binary Labels

Query 1	P@k	Query 2	P@k
	1.0		0
	1.0		0
	0.667		0.333
	0.5		0.25
	0.6		

- Documents are relevant or irrelevant
- MAP: Mean Average Precision:
  - Averaged across positions and queries
  - $MAP = \frac{1}{2}\left(\frac{1}{5}\left(1+1+\frac{2}{3}+\frac{2}{4}+\frac{3}{5}\right) + \frac{1}{4}\left(0+0+\frac{1}{3}+\frac{1}{4}\right)\right) \approx 0.45$
- MRR: Mean Reciprocal Rank
  - Reciprocal of 1st relevant result position
  - $MRR = \frac{1}{2}\left(\frac{1}{1}+1/3\right) \approx 0.67$

[Robertson & Zaragoza '07]

Ranking Evaluation: Ordinal Labels

- Suppose there are 4 relevance levels: Excellent, Good, Fair, Bad

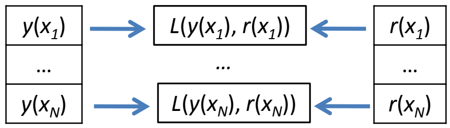
Query 1

- NDCG: Normalized Discounted Cumulative Gain
- Gain for document d_i: $G(d_i) = 2^{r(d_i)} - 1$
- Discount at position i: $D(i) = \log(i+1)$
- Discounted Cumulative Gain: $DCG(k) = \sum_{i=1..k} \frac{G(i)}{D(i)}$
- Normalization: $Z(i) = \max DCG(i)$
- Putting it together: $NDCG(k) = \frac{1}{Z(k)} \sum_{i=1..k} \frac{2^{r(d_i)}-1}{\log(i+1)}$
- Sensitive to top-ranked results
- Correlates with user satisfaction studies

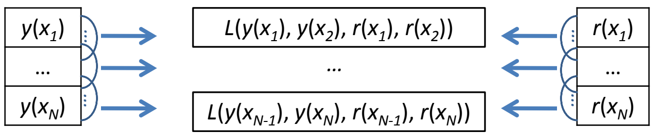
[Järvelin & Kekäläinen '02]
[Bompada *et al* '07]

Learning To Rank: Approach Families

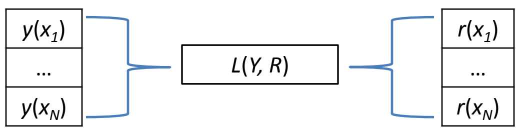
- *Pointwise*: loss is computed for each document independently



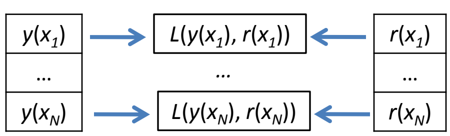
- *Pairwise*: loss is computed on pairs of documents



- *Structural*: optimize loss directly

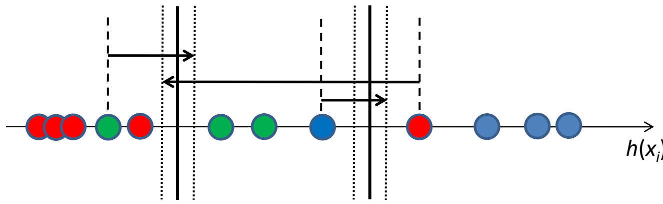


Pointwise Approaches



- Learn from each document *in isolation*
- Standard reductions
  - Regression: relevance/loss is a real-valued variable
  - Classification: relevance is categorical
  - Ordinal regression

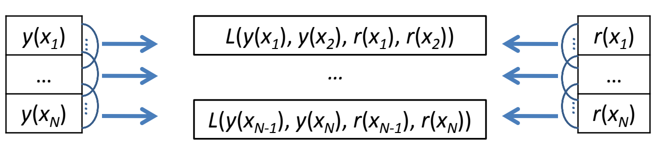
Pointwise L2R: Ordinal Regression



- Loss is based on *thresholds* separating the classes
- Minimization based on *margin/regret*
  - Variants include sum of margins, fixed margin, different constraint setting

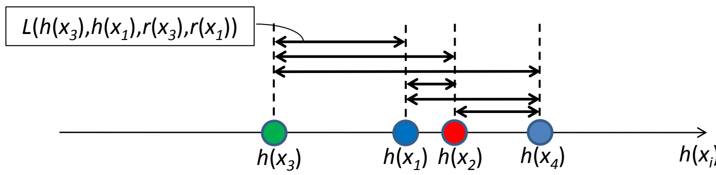
[Krammer & Singer '01]
[Shashua & Levin '02]
[Chu & Keerthi '05]

Pairwise Approaches



- Pointwise approaches ignore relative positions of documents
- Alternative: view ranking as *pairwise classification*
- Pairwise agreements = AUC (for binary labels)
- Natural reduction for incorporating preference training data

Pairwise L2R: Pairwise Loss



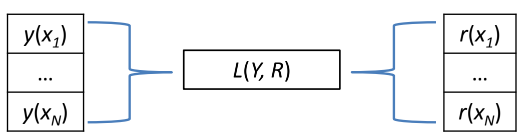
- Overall loss is aggregated over pairwise predictions

  $y_{ij}=r(x_i)- r(x_j)$ $\quad\quad h_{ij}=h(x_i)- h(x_j)$

  – Ranking SVM (for binary $r(x_i)$): hinge loss: $L=|1-y_{ij}h_{ij}|_+$
  – RankBoost (for binary $r(x_i)$): exp loss L = exp $(-y_{ij}h_{ij})$
  – RankNet: cross-entropy (log-loss) based on P(y(x_i)»y(x_j))=exp(h_{ij})/(1+exp(h_{ij}))
  – LambdaRank: directly model *dL/dh* to optimize NDCG

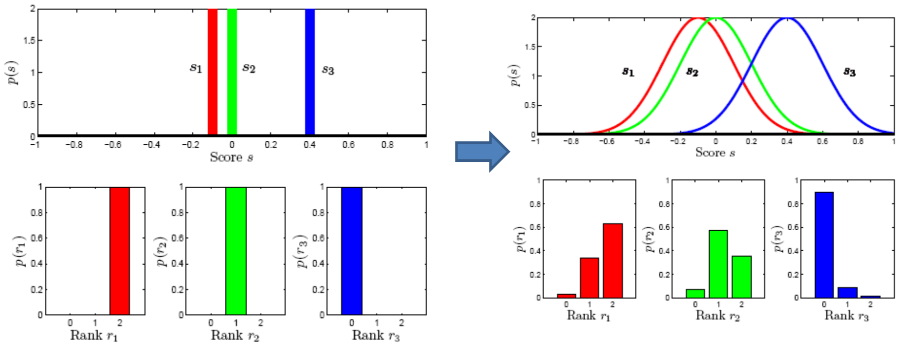
[Cohen *et al.* ‘98]
[Herbrich *et al.* ‘00]
[Freund *et al.* ’03]
[Joachims ‘05]
[Burges *et al.* ‘05]
[Burges *et al.* ‘06]
[Cao *et al.* ‘07]

Structural Approaches



- Goal: optimization of *actual evaluation metric*
- Problem: metrics are not differentiable w.r.t. model parameters
  – MAP, MRR, NDCG are all discontinuous in document scores
- Solutions fall into two families
  – Optimizing a smoothed approximation of the metric (e.g., SoftRank)
  – Optimizing an upper bound of the metric (e.g., ListNet)

Structural L2R: SoftRank



- Key insight: treat predictions as random variables
- Distribution over ranks is obtained by drawing from prediction RVs
- SoftNDCG: $G = G_{max}^{-1} \sum_{j=1}^{N} g(j) D(r_j) \rightarrow \mathcal{G} = G_{max}^{-1} \sum_{j=1}^{N} g(j) E[D(r_j)]$
- BoltzRank (ICML-09): directly approximate the full ranking

[Taylor *et al.* '08]
[Volkovs & Zemel '09]

Learning to Rank: Summary

- Core prediction problem in IR
- Evaluation functions are an active area of IR research
  - User satisfaction is *not* measured via a precision-recall curve
- Ill-behaved objectives → interesting ML problems
- Open problem: can we *learn* what is the right objective

Tutorial Overview

1. IR: Background and Challenges for Learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance
   - Learning from user behavior
   - Learning to rank
3. **Emerging Opportunities for Learning in IR**
   - **Online advertising**
   - Risk-reward tradeoffs for retrieval algorithms
   - Learning complex structured outputs
4. Summary and Bibliography

ML+IR New Opportunities: Online Advertising

- Platform task: select ads to maximize utility
  - *User utility*: relevance
  - *Publisher utility*: yield
  - *Platform utility*: revenue
  - *Advertiser utility*: ROI
- Monetization models
  - *CPM*: cost-per-impression
  - *CPC*: cost-per-click
  - *CPA*: cost-per-action
- Search (CPC/CPA), Contextual (CPC/CPA), Display (CPM)

Ranking for Advertising

- CPC monetization: need to maximize expected revenue:
  $$E[R(ad_i)] = p(click|ad_i) \cdot CPC(ad_i)$$
- CPC depends on auction type; in 2nd price auctions $CPC(ad_i) \le bid(ad_i)$
- *Click probability (CTR) estimation* is the core prediction problem
- Very high-dimensional, very sparse:
  - Features: evidence from context (query/page), ad, user, position, …
  - Billions of queries/pages, hundreds of millions of users, millions of advertisers
  - Clicks are rare
- Ranking is a combinatorial problem with many externalities
  - Co-dependencies between multiple advertisements
  - Optimizing budget allocation for advertisers

Fraud and Quality: Learning Problems

- *Content Ads*: publishers directly benefit from fraudulent clicks
- *Search Ads*: advertisers have strong incentives to game the system
  - Manipulating CTR estimates (for self and competitors)
  - Bankrupting competitors
- *Arbitrage*: aggregators redirect users from one platform to another
- *"Classic" fraud*: fake credit cards

Extraction and Matching

- Advertisers bid on some keywords, but *related* keywords often appear in queries or pages
- Identifying all relevant advertisements is universally beneficial
  - Users: more relevant ads
  - Advertisers: showing ads on more queries/pages → higher coverage
  - Platform: higher competition between advertisements increases CPCs
- Broad match: given query *q*, predict CTR for ads on keyword $k \approx q$
- Different notion of relevance than in search
  - *q*=[*cheap canon G10*] *k*=[*Nikon P6000*]

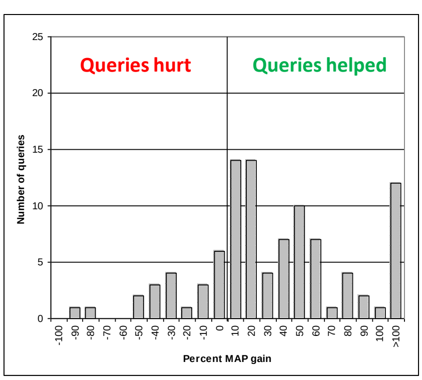
Learning for Personalized Advertising

- Modeling *user attributes and interests* increases monetization
  - Key for social network monetization
- *Demographic prediction* based on behavioral history
  - Large fraction of display advertising sold based on demographics
- *Clustering* and *segment mining*: from macro- to micro-segments
  - Identifing "urban car shoppers", "expecting parents who refinance", …
- Biggest challenges: *privacy* and *scale*
  - Scale: distributed learning via MapReduce

Tutorial Overview

1. IR: Background and Challenges for Learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance
   - Learning from user behavior
   - Learning to rank
3. **Emerging Opportunities for Learning in IR**
   - Online advertising
   - **Risk-reward tradeoffs for retrieval algorithms**
   - Learning complex structured outputs
4. Summary and Bibliography

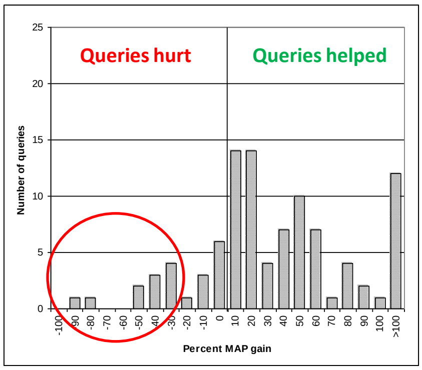
Current query expansion methods work well on average…



Query expansion:
Current state-of-the-art method

Mean Average Precision gain: +30%

…but exhibit high variance across individual queries



Query expansion:
Current state-of-the-art method

This is one of the reasons that even state-of-the-art algorithms are impractical for many real-world scenarios.

Current information retrieval algorithms still have basic problems

- They ignore evidence of risky scenarios & data uncertainty
  - e.g. query aspects not balanced in expansion model
  - Traditionally optimized for average performance, ignoring variance
  - Result: unstable algorithms with high downside risk

Current information retrieval algorithms still have basic problems

- They ignore evidence of risky scenarios & data uncertainty
  - e.g. query aspects not balanced in expansion model
  - Traditionally optimized for <u>average</u> performance, ignoring variance
  - Result: unstable algorithms with high downside risk
- It is hard to integrate multiple task constraints for increasingly complex estimation problems:
  - Personalization, computation constraints, implicit/explicit relevance feedback, …

Current information retrieval algorithms still have basic problems

- They ignore evidence of risky scenarios & data uncertainty
  - e.g. query aspects not balanced in expansion model
  - Traditionally optimized for <u>average</u> performance, ignoring variance
  - Result: unstable algorithms with high downside risk
- It is hard to integrate multiple task constraints for increasingly complex estimation problems:
  - Personalization, computation constraints, implicit/explicit relevance feedback, …
- We need a better algorithmic framework

Example: Ignoring aspect balance increases algorithm risk

Hypothetical query: ‘merit pay law for teachers’

court	0.026
appeals	0.018
federal	0.012
employees	0.010
case	0.010
education	0.009
School	0.008
union	0.007
seniority	0.007
salary	0.006

legal aspect is modeled…

BUT

education & pay aspects thrown away..

A better approach is to jointly optimize selection of terms as a set

Hypothetical query: ‘merit pay law for teachers’

court	0.026
appeals	0.018
federal	0.012
Employees	0.010
case	0.010
education	0.009
school	0.008
union	0.007
seniority	0.007
salary	0.006

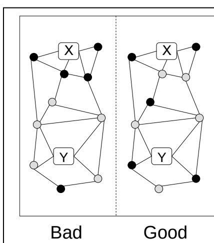
Select terms as a set, not individually, for a more balanced query model

Secret weapons

1. Cast model estimation as constrained optimization
   - Allows rich sets of constraints to capture domain knowledge, reduce risk, and encode structure
   - Efficient convex (LP, QP) or sub-modular formulations
2. Account for uncertainty in data by applying <u>robust optimization</u> methods
   - Define an uncertainty set *U* for the data
   - Then minimize worst-case loss or regret over *U*
   - Often has simple analytical form or can be approximated efficiently

Example of a query expansion constraint on a word graph

- Graph nodes are words
- Related words are colored black (likely relevant) or white (likely not relevant)



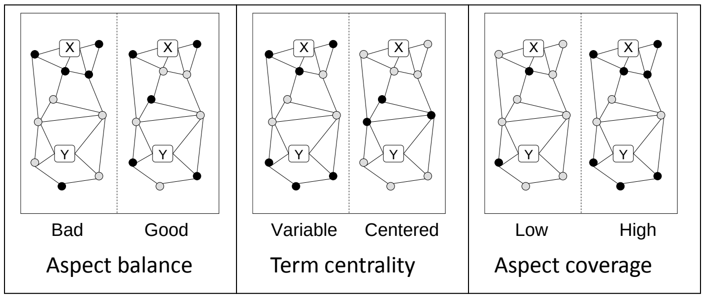
Aspect balance

Two-term query: "X Y"

[Collins-Thompson, NIPS 2008]

Objectives and constraints for query expansion

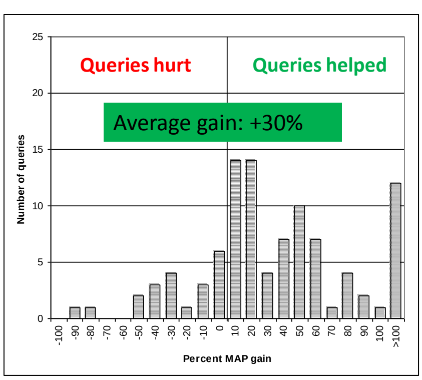
[Collins-Thompson 2008]



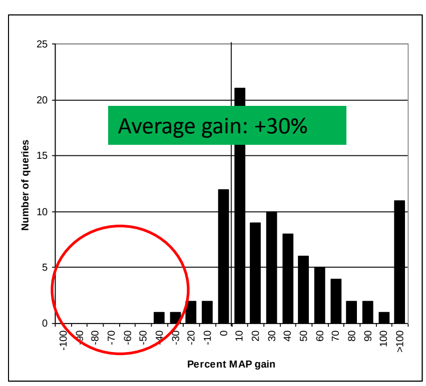
QMOD algorithm

minimize	$-c^T x + \frac{\kappa}{2} x^T \Sigma_\gamma x$	*Term relevance, centrality, risk*
subject to	$Ax \leq \mu + \varsigma_\mu$	*Aspect balance*
	$g_i^T x \geq \varsigma_i, \quad w_i \in Q$	*Aspect coverage*
	$l_i \leq x_i \leq u_i, \; w_i \in Q$	*Query term support*
	$0 \leq x \leq 1$	

We obtain robust query algorithms that greatly reduce worst-case performance while preserving large average gains



Query expansion:
Current state-of-the-art method



Robust version

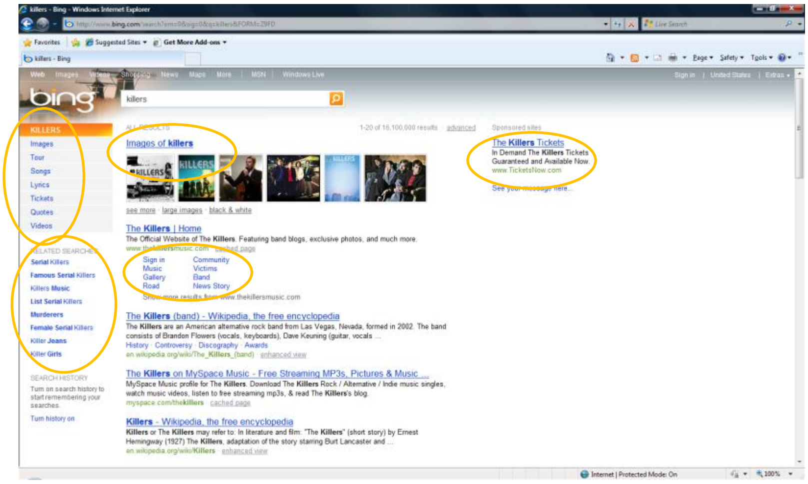
Future directions

- Broad applicability in information retrieval scenarios
  - Query expansion, query alteration, when to personalize, resource selection, document ranking, …
- Learn effective feasible sets for selective operation
- New objective functions, approximations, computational approaches for scalability
- Structured prediction problems in high dimensions with large number of constraints

Tutorial Overview

1. IR: Background and Challenges for Learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance
   - Learning from user behavior
   - Learning to rank
3. **Emerging Opportunities for Learning in IR**
   - Online advertising
   - Risk-reward tradeoffs for retrieval algorithms
   - **Learning complex structured outputs**
4. Summary and Bibliography

IR: Beyond the Ranked List



Structure Increasingly Important in Information Retrieval

- Structured presentation breaks common evaluation and learning paradigms in many ways.
  - Is a click on an indented link the same?
  - What is the "position" of the link following a link with indented links?
  - Is a search page with better ads more relevant than one without?
  - How should heterogeneous media types be displayed together?
  - How are query suggestions evaluated? Is diversification in query suggestions less risky?

- Can a value be placed on each component or is a Reinforcement Learning approach need that apportions blame/credit.

Redundancy, Novelty, Diversity

- Presenting the same information repeatedly is bad.
  - Same link in a list seems obviously bad.
  - Confirming sources?
- Presenting new information is good.
  - With respect to search results, session, a profile?
  - New versus authoritative tradeoffs?
- Both fall under broader scope of diversification:
  - Information content of results
  - Diversify in types of results
  - Types of suggested queries
  - Types of sources (e.g. small and large news outlets)

Maximal Marginal Relevance

(Goldstein & Carbonell, SIGIR '98)

- Given a similarity function $sim(d, d')$ and a relevance function $rel(d, q)$ greedily add documents to *D* to maximize:

$$\lambda \cdot rel(d, q) - (1 - \lambda) \max_{d' \in D} sim(d, d')$$

- Trades off relevance to query with novelty of the document with respect to the more highly ranked documents.

Subtopic Retrieval

(Zhai *et al.*, SIGIR '03)

- When results belong to subtopics or "aspects" (cf. TREC Interactive Track Report '98 – '00), assume the goal is to cover all subtopics as quickly as possible.
- Evaluation measures
  - S-recall(k)
    - (num correct topics retrieved at level k) / (num of all topics)
  - S-precision at recall r: minRank(OPT,r) / minRank(r)
    - Generalizes standard precision and recall.
    - Hard to compute S-precision (equivalent to set-cover).
    - Argue for it as way to normalize difficulty of query.
  - Also cost component for penalizing redundancy.
- Greedy reranking where novelty is based on topic language models.

Learning Complex Structured Outputs

- Chen & Karger, SIGIR '07
  - Ranking conditioning on items above not being relevant, *P(*d_2 *relevant* | d_1 not *relevant, query)*
- Swaminathan *et al.,* MSR-TR '08
  - Often don't know topics, cover words as a proxy.
- Yue & Joachims, ICML '08
  - Using Structural SVMs to learn which words are important to covers.
- Gollapudi *et al.,* WSDM '08
  - Greedy minimization of a submodular formulation based on relevance and utility to user. Assumption that conditional relevance of documents to a query is independent.
- Gollapudi *et al.,* WWW '09
  - 8 desired axioms for diversification (e.g. strength of relevance, strength of similarity), impossibility results for all 8, and investigation of some instantiations

Open Questions Related to Diversity

- What is a good ontology for topical diversification?
- How about for other dimensions (diversity in opinion, result type, etc.)?
- How can an ontology be directly derived from user logs?
- Diversifying Ad Rankings
  - By query intent?

Tutorial Overview

1. IR: Background and Challenges for Learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevances
   - Learning from user behavior
   - Learning to rank
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs for retrieval algorithms
   - Learning complex structured outputs
4. **Summary and Bibliography**

IR Summary

- Basic IR paradigm: satisfying users' information needs

- At its core, IR studies *retrieval-related prediction tasks*
- Much of IR is driven by focus on measurement and improvement against ***user*** satisfaction.

IR Increasingly Relies on ML

- General shift from heuristics to formal probabilistic models.
- More recent shift to discriminative models where previous models serve as input features.
- Salient computational features:
  - Massive amounts of documents.
  - Nearly infinite variety in expressing an information need.
  - Huge amount of user-generated data.

Select future directions

- Methods that use multiple sources of relevance: clicks, expert judgments, human computation labels, …
  - Optimization criterion?
  - Theory for ground truth that cannot be equally trusted.
  - Measures for the usefulness of a label
- Prediction tasks
  - Predicting clicks (on a result, an ad, a query suggestion, …)
  - Predicting when to personalize
  - Predicting query performance
- Risk & Reward
  - Identifying value of components in structured retrieval.
  - Learning and dealing with varying risk/reward tradeoffs (e.g. diversifying suggested queries rather than results).
- The IID assumption and active learning
  - If active learning is used to drive label collection, will the resulting collection be biased for use as an evaluation collection?
  - Can evaluation be debiased using standard methods?

Pointers to Data Resources

- LETOR
  - Learning to rank data: http://research.microsoft.com/en-us/um/beijing/projects/letor/index.html
- TREC
  - Data available from various focused tracks over the years: http://trec.nist.gov/
- Collection of Relative Preferences over Documents
  - http://ciir.cs.umass.edu/~carteret/BBR.html
- Preference Collection for Image Search
  - http://go.microsoft.com/?linkid=9648573
- Netflix
  - Movie recommendations, http://www.netflixprize.com/
- AOL
  - Query log released publicly. See an IR practitioner near you for copies cached before original distribution was removed.

Thanks!

Paul Bennett (pauben@microsoft.com)
Misha Bilenko (mbilenko@microsoft.com)
Kevyn Collins-Thompson (kevynct@microsoft.com)

Bibliography (cont'd)

- General IR
  - *Introduction to Information Retrieval* (2008), Manning, Raghavan, & Schütze.
  - *Search Engines: Information Retrieval in Practice* (2009), Croft, Metzler, & Strohman.
- Seminal IR Work
  - Probability Ranking Principle
    - Robertson, S.E. The probability ranking principle in IR. Journal of Documentations, 33, 4 (1977), 294-304.
  - TFIDF
    - Salton, G., Buckley, C. Term weighting approaches in automatic text retrieval. *Information Processing and Management.* 24 (1988), 513—523.
  - Okapi BM25
    - Robertson, S.E., Walker, S., Hancock-Beaulieu, M. Large test collection experiments on an operational, interactive system: Okapi at trec. *Information Processing and Management.* 31 (1995), 345—360.

Bibliography (cont'd)

- Seminal IR Work (cont.)
  - Maximal Marginal Relevance
    - Carbonell, J., Goldstein, J. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR '98*. pp. 335–336. 1998.
  - NDCG
    - Järvelin, K., Kekäläinen, J. IR evaluation methods for retrieving highly relevant documents. In *SIGIR '00*. pp. 41-48. 2000.
  - HITS
    - Kleinberg, J. Authoritative sources in a hyperlinked environment. *JACM.* 46, 5 (1999), 604–632.
  - PageRank
    - Brin, S., Page, L. The anatomy of a large-scale hypertextual Web search engine. In *WWW ''98. pp. 107-117. 1998.*
  - Language Model Based Approach
    - Ponte, J., Croft, W.B. A language modeling approach to information retrieval. In *SIGIR* '98. pp. 275–281.
    - Lafferty, J., Zhai, C. Document language models, query models, and risk minimization for information retrieval. In *SIGIR* '01. pp. 111-119. 2001.

Bibliography (cont'd)

- Evaluation
  - Carterette et al., SIGIR 2008
  - Carterette et al., ECIR 2009
- Implicit measures
  - Kelly & Teevan, SIGIR Forum 2005
  - Fox *et al.,* TOIS 2005
  - White *et al.,* SIGIR 2005
  - Boyen *et al.* IBIS@AAAI 1996
- Learning To Rank
  - Aslam *et al.,* SIGIR 2009

Bibliography (cont'd)

- Interpreting Clicks
  - Joachims *et al*, SIGIR 2005
  - Agichtein *et al*., SIGIR 2006
  - Radlinski & Joachims, KDD 2005
  - Craswell *et al*., WSDM 2008
- Learning from User Logs
  - Radlinski *et al.,* ICML 2008
  - Bilenko & White, WWW 2008
  - Mihalkova & Mooney, BSCIW@NIPS 2008
  - Teevan *et al*., SIGIR 2008
  - Teevan *et al*., 2009
  - Das Sarma, KDD 2008
  - Dupret *et al*., QLA@WWW 2007
  - Chapelle & Zhang, WWW 2009
  - Guo *et al.* WWW 2009

Bibliography (cont'd)

- Label Type
  - Järvelin & Kekäläinen, SIGIR 2000
  - Carterette *et al.,* ECIR 2008
  - Clarke *et al.* SIGIR 2008
  - von Ahn & Dabbish, CHI 2004
  - Bennett *et al.,* WWW 2009
  - Sheng & Provost, KDD 2008
  - Donmez *et al.,* KDD 2009
- Diversity
  - Zhai, Cohen, Lafferty, SIGIR 2003
  - Swaminathan *et al.,* MSR-TR 2008
  - Yue & Joachims, ICML 2008
  - Gollapudi *et al.,* WSDM 2008
  - Gollapudi *et al.,* WWW 2009

Bibliography (cont'd)

- Learning to rank: Tutorials
  - Tie-Yan Liu
    - http://www2009.org/pdf/T7A-LEARNING%20TO%20RANK%20TUTORIAL.pdf
  - Yisong Yue & Filip Radlinski
    - http://www.yisongyue.com/talks/LearningToRank_NESCAI08_part1.ppt
    - http://radlinski.org/paper.php?p=LearningToRank_NESCAI08.pdf
- Learning to rank: evaluation measures
  - Järvelin & Kekäläinen, TOIS 2002
  - Robertson & Zaragoza, IR 2007
  - Bombada et al., SIGIR 2007
  - Sakai & Kando, IR 2008
  - Yilmaz et al., SIGIR 2008
- Learning to rank: pointwise approaches
  - Krammer & Singer, NIPS 2001
  - Shashua & Levin, NIPS 2002
  - Chu & Keerthi, ICML 2005
  - Chu & Ghahramani, ICML 2005

Bibliography (cont.)

- Learning to rank: pairwise approaches
  - Cohen et al., JAIR 1999
  - Herbrich et al., Advances in Large Margin Classifiers, 1999
  - Freund et al., JMLR 2003
  - Joachims, ICML 2005
  - Burges et al., ICML 2005
  - Burges et al., NIPS 2006
  - Tsai et al., ICML 2007
- Learning to rank: structural approaches
  - Xu & Li, SIGIR 2007
  - Cao et al., ICML 2007
  - Taylor et al., WSDM 2008
  - Qin et al., NIPS 2008
  - Xia et al., ICML 2008
  - Guiver & Snelson, SIGIR 2008
  - Volkovs & Zemel, ICML 2009
  - Lan et al., ICML 2009

Bibliography (cont'd)

- Online Advertising
  - Keyword Auctions
    - Edelman et al., American Economic Review 2007
    - Varian, International J. of Industrial Organization 2007
    - Athey & Ellison, 2008
  - CTR estimation, Search Ads
    - Lacerda et al., SIGIR 2006
    - Richardson et al., WWW 2007
    - Chakrabarti et al., WWW 2008
    - Broder et al., CIKM 2008
  - Matching and Extraction, Content Ads
    - Yih et al., WWW 2006
    - Broder et al., SIGIR 2007
    - Gupta et al., KDD 2009
  - Fraud and Quality
    - Gunawardana & Meek, WWW workshop 2008
    - Sculley et al., KDD 2009

Bibliography (cont'd)

- History of relevance and IR models
  - M.E. Maron and J. L. Kuhns. (1960) On relevance, probabilistic indexing, and information retrieval. Journal of the ACM 7:216-244.
  - Mizarro, S. Relevance: The whole history. *Journal of the American Society for Information Science 48*, 9 (1997), 810.832.
- Classical probabilistic IR model and extensions
  - S.E. Robertson and K. Spärck Jones, Relevance weighting of search terms. *Journal of the American Society for Information Science* **27**, 129-46 (1976). http://www.soi.city.ac.uk/~ser/papers/RSJ76.pdf
  - S.E. Robertson. (1990) On term selection for query expansion. Journal of Documentation. 46, 359-364. http://www.soi.city.ac.uk/~ser/papers/on_term_selection.pdf
  - Robertson, S. E. and Walker, S. (1999). Okapi/keenbow at TREC-8. In Voorhees, E. M. and Harman, D. K., editors, *The Eighth Text REtrieval Conference (TREC 8)*. NIST Special Publication 500-246.
  - C. Elkan. *Deriving TF-IDF as a Fisher kernel* (pdf). *Proceedings of the International Symposium on String Processing and Information Retrieval* (SPIRE'05), Buenos Aires, Argentina, November 2005, pp. 296-301.

Bibliography (cont'd)

- Relevance feedback/query expansion
  - J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. SIGIR 2001. pp. 111-119.
  - K. Collins-Thompson. "Estimating robust query models with convex optimization". *Advances in Neural Information Processing Systems 21* (NIPS), 2008. pg. 329-336.
  - K. Collins-Thompson. "Robust model estimation methods for information retrieval". Ph.D. thesis, Carnegie Mellon University, 2008.

Bibliography (cont'd)

- Query difficulty and query performance prediction
  - Yom-Tov, E., Fine, S., Carmel, D., and Darlow, A. 2005. Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval. In *Proceedings of SIGIR* 2005. ACM, New York, NY, 512-519.
  - S. Cronen-Townsend, Y. Zhou , W.B. Croft, Predicting query performance, Proceedings of SIGIR 2002, Tampere, Finland.
  - V. Vinay, Ingemar J. Cox, Natasa Milic-Frayling, Kenneth R. Wood On ranking the effectiveness of searches. Proceedings of SIGIR 2006. Seattle. Pg 398-404.
  - Javed A. Aslam, Virgiliu Pavlu: Query Hardness Estimation Using Jensen-Shannon Divergence Among Multiple Scoring Functions. ECIR 2007: 198-209

Bibliography (cont'd)

- Language modeling for IR
  - J.M. Ponte and W.B. Croft. 1998. A language modeling approach to information retrieval. In *SIGIR 21.* pp. 275-281.
  - A. Berger and J. Lafferty. 1999. Information retrieval as statistical translation. *SIGIR 22*, pp. 222–229.
  - Workshop on Language Modeling and Information Retrieval, CMU 2001. http://sigir.org/forum/S2001/LM.pdf
  - The Lemur Toolkit for Language Modeling and Information Retrieval. Open-source toolkit from CMU/Umass. LM and IR system in C(++) http://www.lemurproject.org/~lemur/
  - C. Zhai. 2008. Statistical language models for information retrieval: a critical review. Foundations and Trends in Information Retrieval Vol. 2, No. 3.
  - V. Lavrenko. A Generative Theory of Relevance. Doctoral dissertation. Univ. of Massachusetts Amherst, 2004.
  - Metzler, D. and Croft, W.B., "A Markov Random Field Model for Term Dependencies," *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (SIGIR 2005), 472-479, 2005
- Topic models
  - D. Blei, A. Ng, M. Jordan (2001) Latent dirichlet allocation. NIPS 14. pp 601-608.
  - G. Doyle and C. Elkan (2009) Accounting for Burstiness in Topic Models. ICML 2009.

Bibliography (cont'd)

- Federated search / distributed IR / meta-search
  - Callan, J. (2000). Distributed information retrieval. In W.B. Croft, editor, *Advances in Information Retrieval*. (pp. 127-150). Kluwer Academic Publishers.
  - L. Si. (2006). Federated Search of Text Search Engines in Uncooperative Environments. Doctoral dissertation. Carnegie Mellon University.
  - F. A. D. Neves, E. A. Fox, and X. Yu. Connecting topics in document collections with stepping stones and pathways. In CIKM, pages 91–98, 2005.
  - Aslam, J. A. and Montague, M. 2001. Models for metasearch. In *Proceedings of SIGIR 2001*(New Orleans, Louisiana, United States). SIGIR '01. ACM, New York, NY, 276-284.
- Adaptive filtering
  - C. Faloutsos and D. W. Oard. A survey of information retrieval and filtering methods. Technical report, Univ. of Maryland, College Park, 1995.
  - Y. Zhang. (2005) Bayesian Graphical Models for Adaptive Filtering. Doctoral dissertation, Carnegie Mellon University.
  - Y. Zhang, J. Callan, and T. Minka. (2002) Novelty and redundancy detection in adaptive filtering. In Proceedings of SIGIR 2002.

Bibliography (cont'd)

- Adversarial IR
  - da Costa Carvalho, A. L., Chirita, P., de Moura, E. S., Calado, P., and Nejdl, W. 2006. Site level noise removal for search engines. In *Proceedings of* WWW '06. ACM, New York, NY, 73-82.
  - Baoning Wu and Brian D. Davison: "Cloaking and Redirection: A Preliminary Study". Workshop on Adversarial Information Retrieval on the Web, Chiba, Japan, 2005.
  - Du, Y., Shi, Y., and Zhao, X. 2007. Using spam farm to boost PageRank. In *Proceedings of the 3rd international Workshop on Adversarial information Retrieval on the Web* (Banff, Alberta, Canada, May 08 - 08, 2007). AIRWeb '07, vol. 215. 29-36.
  - Ntoulas, A., Najork, M., Manasse, M., and Fetterly, D. 2006. Detecting spam web pages through content analysis. In *Proceedings of the 15th international Conference on World Wide Web* (Edinburgh, Scotland, May 23 - 26, 2006). WWW '06. ACM, New York, NY, 83-92.
  - Nadav Eiron , Kevin S. McCurley , John A. Tomlin, Ranking the web frontier, Proceedings of the 13th international conference on World Wide Web, May 17-20, 2004, New York, NY, USA.
  - Predicting Bounce Rates in Sponsored Search Advertisements, D. Sculley, Robert Malkin, Sugato Basu, Roberto J. Bayardo, *Proc. of KDD* 2009.
- Temporal IR
  - D. Lewandowski. A three-year study on the freshness of Web search engine databases. Journal of Information Science Vol. 34, No. 6, 817-831 (2008).
  - E. Adar, J. Teevan, S. Dumais and J. Elsas (2009). The Web changes everything: Understanding the dynamics of Web content. In *Proceedings of WSDM 2009*.
  - Eytan Adar, Jaime Teevan, and Susan Dumais, "Large Scale Analysis of Web Revisitation Patterns," CHI'08, Florence, Italy, April 5-10, 2008.

Acknowledgments

- We gratefully acknowledge contributions by the following:
  - Filip Radlinski, MSR Cambridge