Machine Learning and IR:
Recent Successes and New Opportunities

ECIR 2010 Tutorial

Paul Bennett
Misha Bilenko
Kevyn Collins-Thompson

http://research.microsoft.com/ecir-2010-mlir-tutorial

Microsoft Research

Tutorial Audience & Goals

- Audience
  - Half have applied machine learning (perhaps as black box)
  - Half are IR researchers with casual machine learning exposure
  - Some (10-15%) have developed new ML techniques and are looking for other target applications.
- Goals
  - Provide overview of core Machine Learning (ML) methods
  - Show how ML methods apply to important IR problems
  - Survey recent high-impact ML contributions to IR
  - Highlight IR areas with promising opportunities for ML

Tutorial Overview

1. Introduction
   - Machine learning background via text classification
   - IR challenges for machine learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance and learning to rank
   - Learning from user behavior
   - Query-related prediction tasks
   - Social media
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs and other optimization frameworks for retrieval
   - Learning complex structured outputs
4. Summary and Bibliography

Basic Machine Learning Approach

- Formulate prediction problem
- State as optimization
- Optimize

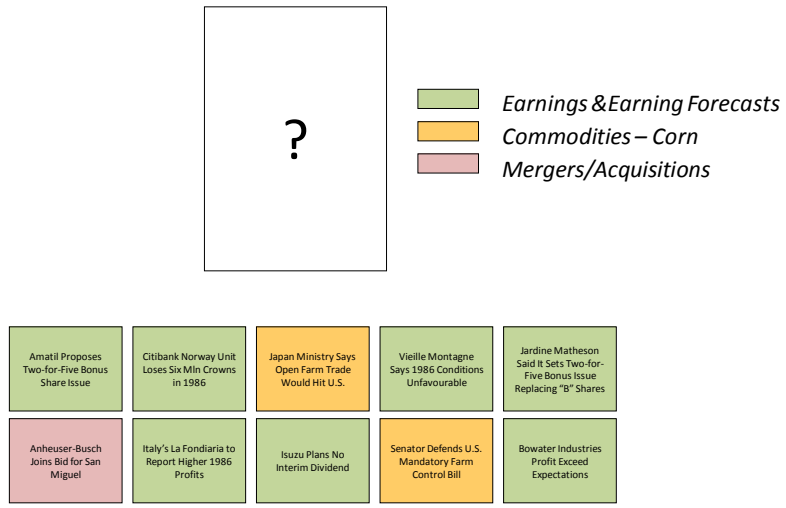
Introduction to ML: Text Classification

- Given: Collection of example news stories already labeled with a category (topic).
- Task: Predict category for news stories not yet labeled.
- For our example, we'll only get to see the headline of the news story.
- We'll represent categories using colors. (All examples with the same color belong to the same category.)
  - *Earnings and Earning Forecasts*
  - *Commodities – Corn*
  - *Mergers/Acquisitions*

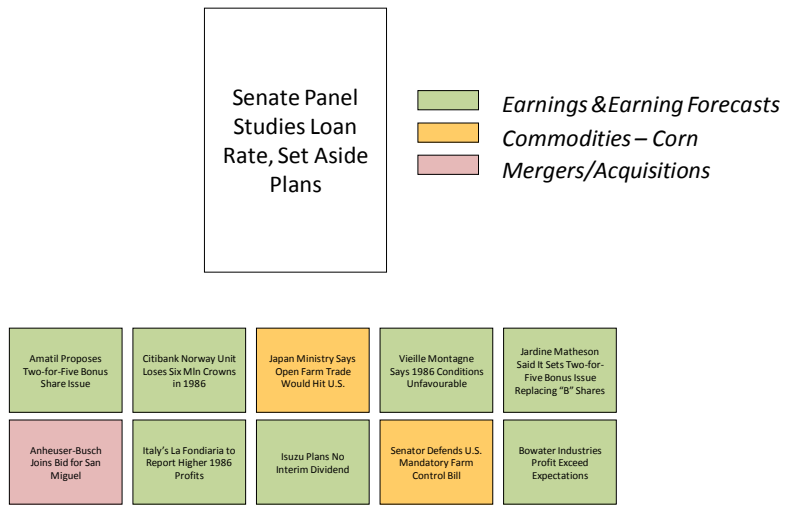
Labeled Examples

Amatil Proposes Two-for-Five Bonus Share Issue	Citibank Norway Unit Loses Six Mln Crowns in 1986	Japan Ministry Says Open Farm Trade Would Hit U.S.	Vieille Montagne Says 1986 Conditions Unfavourable	Jardine Matheson Said It Sets Two-for-Five Bonus Issue Replacing "B" Shares
Anheuser-Busch Joins Bid for San Miguel	Italy's La Fondiaria to Report Higher 1986 Profits	Isuzu Plans No Interim Dividend	Senator Defends U.S. Mandatory Farm Control Bill	Bowater Industries Profit Exceed Expectations

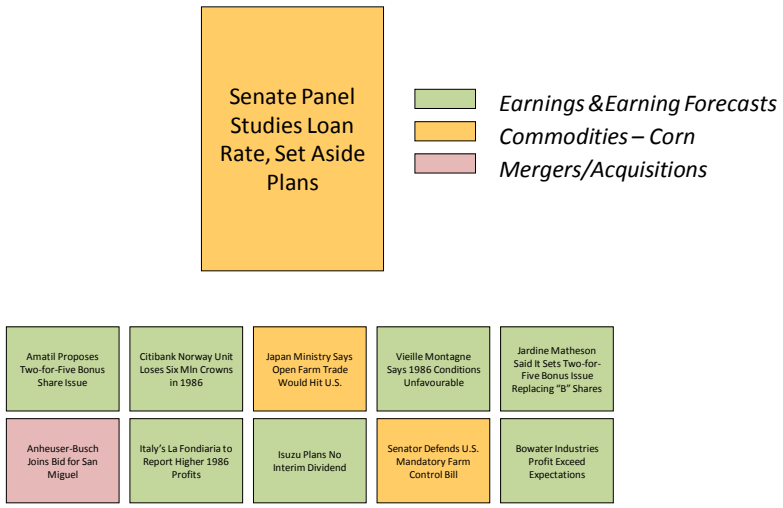
What topic (color) to predict before seeing the document contents?



Predict with Evidence



The Actual Topic



Defining Rules By Hand

Experience has shown that hand-coding rules is:

- too time consuming
- too difficult
- increasingly inconsistent as the rule set gets large

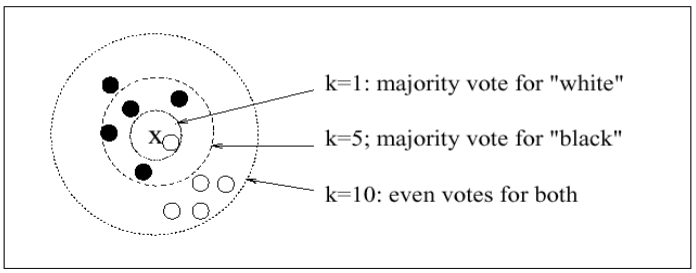
Supervised Statistical Learning

- Humans can encode knowledge of what constitutes membership in a category by providing labeled examples (supervision).
- This encoding can then be automatically applied by a machine to categorize new examples.
- Classic problem: given a set of labeled examples $\langle x_1, y_1 \rangle \ldots \langle x_n, y_n \rangle$ drawn from a distribution $\mathcal{D}$, produce a function f such that $E_{\mathcal{D}}[f(x) \neq y]$ is minimized.

K-Nearest Neighbor (kNN)

(Fix & Hodges, 1951; Duda & Hart, 1957)

K-Nearest Neighbor using a *majority* voting scheme

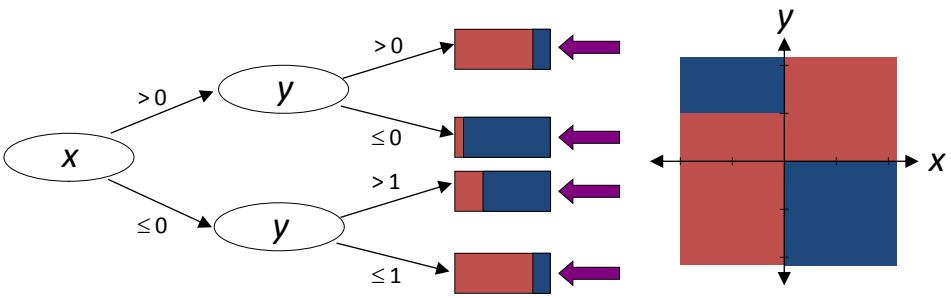


- Error minimized when we predict class, *y,* with maximal posterior $P(y \mid x)$. Best such possible called Bayes error.
- If $k \to \infty$ and $k/n \to 0$ as $n \to \infty$, then kNN approaches Bayes error with probability 1 (Devroye, Györfi, Lugosi, 1996). Called *Bayes consistent* or *strong consistency*.

[Diagram courtesy of Yiming Yang, CMU]

Decision Trees

(Hunt *et al.,* 1966; Friedman, 1977; Breiman *et al.,* 1984)



- Partitions the input space (into hyper-rectangles) and makes a single prediction for each region. Also Bayes consistent.
- Optimization goal: smallest tree with lowest entropy at leaves (various formulations with complexity penalty + prediction).
- Typically greedy optimization step.

13

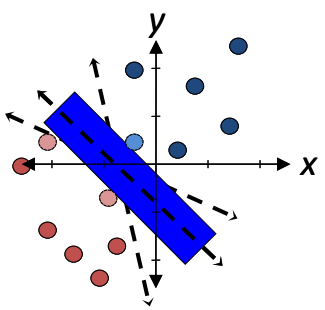
Support Vector Machines (SVMs)

(Vapnik, 1995; Cortes & Vapnik, 1995)

- To classify, x, use a weight on each x_i in the training set and a kernel (similarity meeting inner product conditions), $K(\cdot, \cdot)$. Let $y_i \in \{-1,1\}$, and $y_i = 1$ *iff* x_i is in the positive class. Classify x as positive when:

$$\sum_i \alpha_i y_i K(x, x_i) + b \geq 0$$

- To find α_i, find maximally separating hyperplane.
- Weakens error to "margin" per example (distance from separator) and minimizes sum which upper-bounds error.
- Kernels enable modeling higher-order effects without enumerating cross-product evaluation, e.g. $K(x_i,x_j)=(x_i \cdot x_j+1)^d$.



14

Naïve Bayes
(Maron & Kuhns, 1960; Abramson, 1963)

- Propose model of how data are *generated* and predict most likely class.

$$\arg\max_{c \in Classes} P(c \mid d) = \arg\max_{c \in Classes} P(c)P(d \mid c)$$

- Unigram (multinomial naïve Bayes)

$$P(c \mid d) \propto P(c) \prod_{w \in d} P(w \mid c)^{c(w,d)}$$

- Assumes *class-conditional* independence
  - NOT the same as independence.
- Parameters set to maximize likelihood of data under assumptions.
- Still used because of simplicity, but see *latent dirichlet allocation (LDA)* and derivatives (e.g. labeled LDA) for modeling higher order effects.

15

Representation choice can be critical

- As in IR, representation plays a key role in TC and ML more generally.
  - *tfidf,* stemming, stop word removal, feature selection …
- Some representation choices are important nearly all the time (*e.g.* document length normalization).
- Most representation issues are impacted by:
  - Underlying learning task.
  - Algorithm used to learn (*e.g.* feature selection less important when using a *regularizer* – common in SVM & logistic regression).

Feature Representation

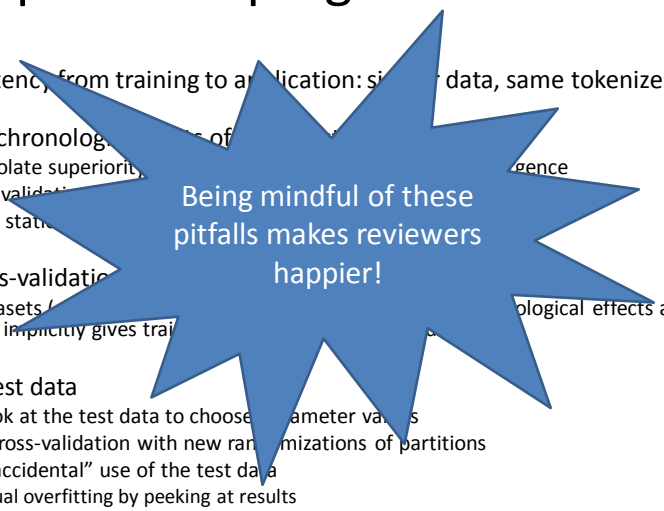
- Features can be more than the words that occurred.
  - Metadata: origin IP, source (English, French, …) language of web page, type of document (pdf, html, mp4, pptx), …
  - Predicting Commerciality:
    - Are ads on the web page?
  - Predicting Spam:
    - Percent capitalized letters
    - # of exclamation points
  - Score for topical cohesion of search results returned from a query.
  - Learning to rank features
    - Document: Page Rank, …
    - Query: length of query, …
    - Query-Document: BM25, …
- Exploring the feature space is often a quick way to get performance improvements, but beware the black box …
  - Heterogeneous features on very different scales.
  - Relationships outside of model's hypothesis space.

Typical pitfalls in pragmatics & evaluation

- Keep consistency from training to application: similar data, same tokenizer, ….
- Beware the chronological splits of train/test
  - Hard to isolate superiority/inferiority of model from possible divergence
  - Use cross-validation as a sanity check
  - Very large static test sets can also help
- Beware cross-validation (IR perspective)
  - Many datasets (e.g. queries related to the same event) have chronological effects and cross-validation implicitly gives training data linked to the test data
- Honor the test data
  - NEVER look at the test data to choose parameter values
  - Perform cross-validation with new randomizations of partitions
  - Beware "accidental" use of the test data
    - Gradual overfitting by peeking at results
    - Computing idfs or feature selection over all the data instead of just training

Typical pitfalls in pragmatics & evaluation

- Keep consistency from training to application: same data, same tokenizer, ….
- Beware the chronological aspects of …
  - Hard to isolate superiority … gence
  - Use cross-validation …
  - Very large statistical …
- Beware cross-validation …
  - Many datasets … biological effects and cross-validation implicitly gives train…
- Honor the test data
  - NEVER look at the test data to choose parameter values
  - Perform cross-validation with new randomizations of partitions
  - Beware "accidental" use of the test data
    - Gradual overfitting by peeking at results
    - Computing idfs or feature selection over all the data instead of just training



Basic Machine Learning Approach

- Basics
  - Formulate prediction problem
  - State as optimization
  - Optimize
- Optimization too hard:
  - Solve something that asymptotically is good (then get more data)
  - Weaken the optimization to a related (pref upper-bound) problem and solve that
- Need better performance
  - Get more data (labeled or unlabeled)
  - Explore feature space
  - Explore higher order effects
    - Different similarity functions (kernels)
    - Different modeling assumptions (graphical models or structured prediction)

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

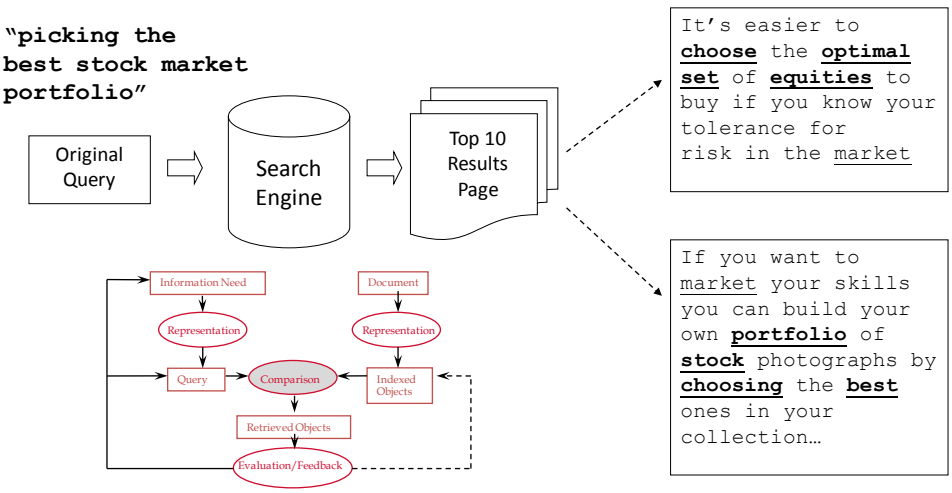
IR: Cornucopia of ML Problems

- *Classification*: content/query categorization, spam detection, entity recognition, …
  - Traditional supervised (labeled-data) classification as covered in introduction.
- *Ranking*: result selection and ordering
  - Rank items with supervision, e.g. order e-mails in Inbox in priority order.
- *Clustering*: retrieval result organization, user need segmentation
  - Organize data into "natural classes" with no supervision.
- *Semi-supervised learning*: unlabeled data is omnipresent
  - Use unlabeled data to improve solution from supervised learning.
- *Active learning*: ranking, recommenders
  - Max performance with min labels by choosing examples to label intelligently.
- *Multi-instance learning*: image retrieval
  - Presence of a concept in a larger bag, e.g. this e-mail contains a "to-do" item.
- *Reinforcement learning*: online advertising
  - Find policy to maximize reward – algorithm takes "action" and receives reward as feedback.

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: What if query and document terms don't match? Or match incorrectly?



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

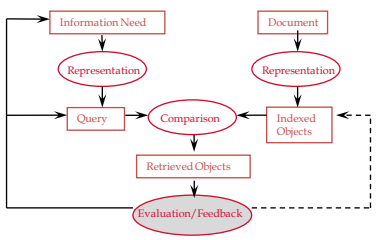
- `System-oriented' relevance:
  - Overlap in representations of Q and D
- But simple overlap ignores many important factors, such as:
  - Preferences and prior knowledge of user who issued request
  - Task that prompted the request
  - Other documents in collection
  - Previous queries of this or other users
  - External information on the (non) relevance of D
- Mizzarro [1997] surveyed 160 different formulations of relevance for IR tasks
- Learning challenge: Objective is not well-defined.

A machine learning view of relevance

- Observations:
  - Feedback, documents, query, user model, etc.
- Training data:
  - Explicit or implicit signals about relevance or non-relevance
- Objective:
  - Learn a relevance prediction function that optimizes performance measure (e.g. MAP) over query training set
  - Must effectively generalize to unseen queries

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
- Learning challenges: noisy data and converting IR performance measures like MAP to tractable ML objectives



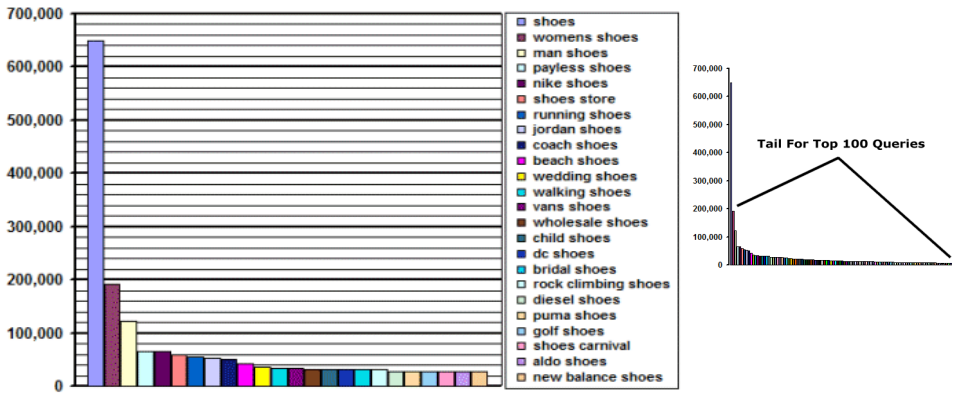
IR challenge: Integrating multiple resources

Machine learning challenges:

- 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]
  - Vertical search
- 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]

- 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)

IR challenge: the 'long tail' of search logs



- 25% of queries are new each day, and 50% of words are seen once
- Skewed distribution, high-dimensional feature space
- Learning challenges: sparse evidence, generalizing to unseen queries, evaluation (obtaining a good sample of judgments) etc.

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

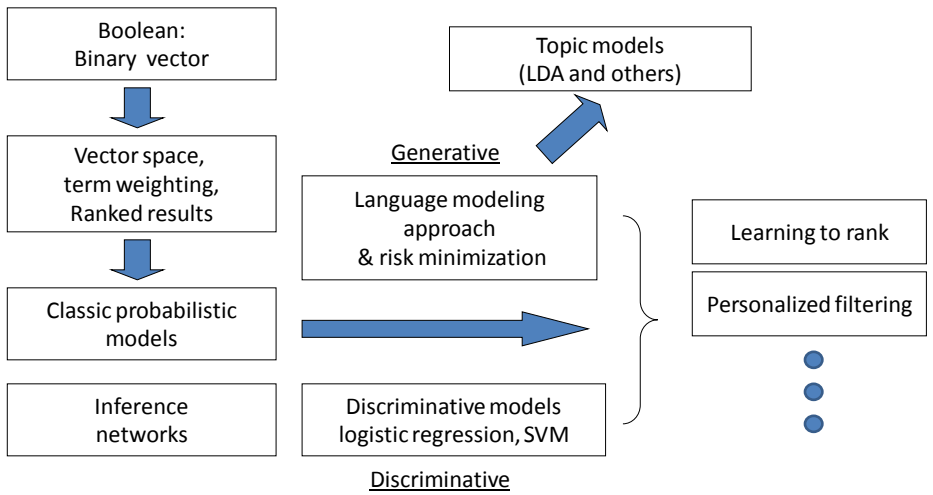
IR challenge: Adversarial issues

- Continuous, evolving `war' between providers and spammers
- Spamming: Artificial rank 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
- Learning challenge: modeling outliers, spam detection

Tutorial Overview

1. Introduction
   - Machine learning background via text classification
   - IR challenges for machine learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance and learning to rank
   - Learning from user behavior
   - Query-related prediction tasks
   - Social media
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs and other optimization frameworks for retrieval
   - Learning complex structured outputs
4. Summary and Bibliography

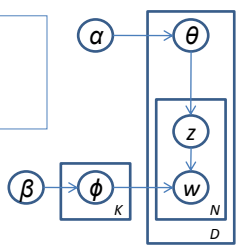
Highly simplified summary of IR retrieval model development

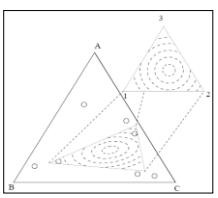


Generative topic models: Latent Dirichlet Allocation [Blei, Ng, Jordan. 2001]

- 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
- Advantages:
  - Allows topics and words to vary in generality
  - Simple assumptions, interpretable parameters
- Disadvantages:
  - Not good at handling outliers
  - LDA: Bursty in topics, but not in words
  - DCM-LDA: Captures topic + word burstiness
    [Doyle & Elkan, ICML 2009]

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

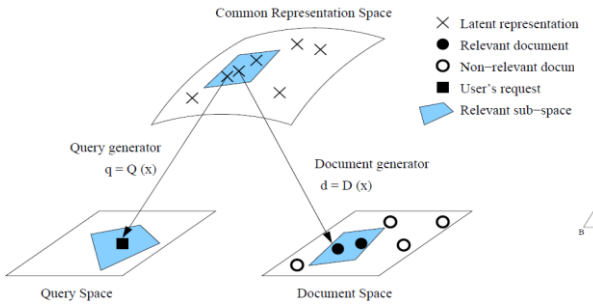


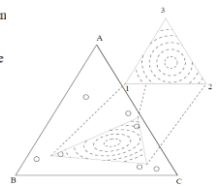


LDA with vocabulary A, B, C

Generative Relevance Model

[Lavrenko 2004]





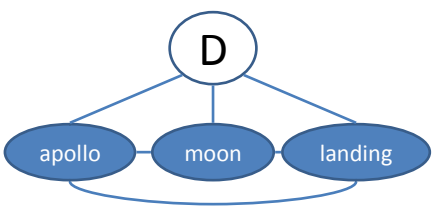
LDA

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

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

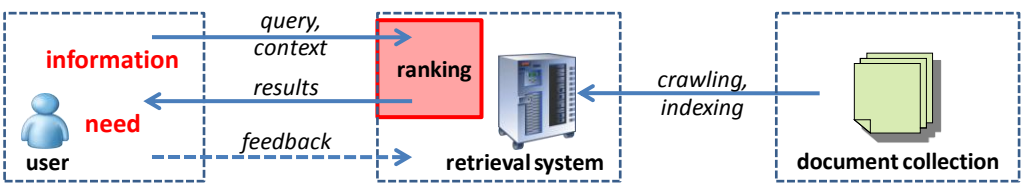
$$P_{G,\Lambda}(Q,D) \overset{rank}{=} \sum_{c \in T_{QD}} \lambda_c f_c(c) + \sum_{c \in O_{QD}} \lambda_c f_c(c) + \sum_{c \in U_{QD}} \lambda_c f_c(c) + \sum_{c \in D} \lambda_c f_c(c)$$

Individual terms LM or BM25 weights	Ordered query terms	Unordered query terms	Document

Discriminative and hybrid models

- Linear and log-linear feature-based models
  - Logistic regression [Gey 1994]
  - Linear discriminant model [Gao et al. '05]
  - Maximum Entropy [Cooper 1993, Nallapati 2004]
  - Markov Random Field model [Metzler and Croft '05]
- Learning methods
  - Direct maximization of MAP using parameter sweep [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]
- Learning Challenge: many negative, few positive examples

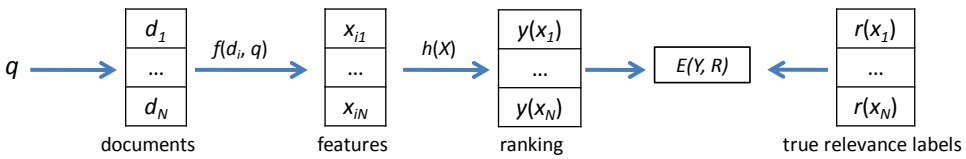
Ranking: Core Prediction Problem in IR



- Context-dependent vs. context-independent ranking
  - *Context-dependent*: relevance w.r.t. information need (query, page, user)
    - Search result ranking, advertisement selection, news recommending
  - *Context-independent*: absolute relevance (PageRank, HITS, SALSA, etc.)
    - Static webpage ranking, crawl scheduling, index tiering

Ranking as a Learning Problem

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



- Can also be trained on relative feedback : $(q, d_1 \prec d_2)$
- Features capture diverse evidence sources
  - Query-document, document, query, behavioral data, …
  - E.g.: BM25, PageRank, query frequency, historical CTR, …

Practical Considerations

- 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
- Exhaustive labeling is impossible: distribution is *always skewed*
  - Judges label all top-rated documents, plus heuristically selected lower-rated
- Labeling issues: intent ambiguity, interjudge disagreement, implicit feedback, ….

Ranking Evaluation

Binary

$Query_1$	P@k	$Query_2$	P@k
	1.0		0
	1.0		0
	0.667		0.333
	0.5		0.25
	0.6		

Ordinal

$Query_1$	Gain	$Query_2$	Gain
	2^3-1=7		2^1-1=1
	2^2-1=3		2^3-1=7
	2^1-1=1		2^1-1=1
	2^0-1=0		2^2-1=3
	2^3-1=7		

- MAP: Mean Average Precision

$$MAP = \frac{1}{2}\left(\frac{1}{5}\left(1 + 1 + \frac{2}{3} + \frac{2}{4} + \frac{3}{5}\right) + \frac{1}{4}\left(0 + 0 + \frac{1}{3} + \frac{1}{4}\right)\right) \approx 0.45$$

- MRR: Mean Reciprocal Rank

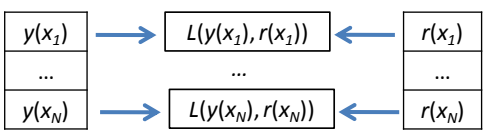
$$MRR = \frac{1}{2}\left(\frac{1}{1} + 1/3\right) \approx 0.67$$

- 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: $Z(i) = \max DCG\ (i)$

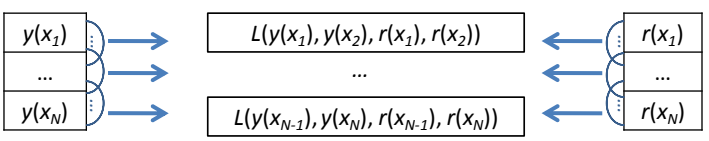
$$NDCG(k) = \frac{1}{Z(k)} \sum_{i=1..k} \frac{2^{r(d_i)} - 1}{\log(i+1)}$$

Learning To Rank: Approach Families

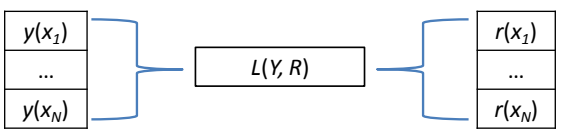
- *Pointwise*: loss is computed for each document independently



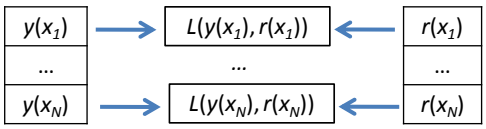
- *Pairwise*: loss is computed on pairs of documents



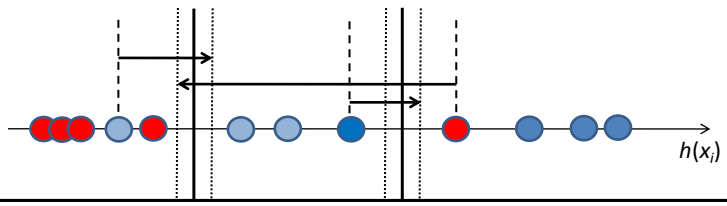
- *Structural*: optimize loss directly



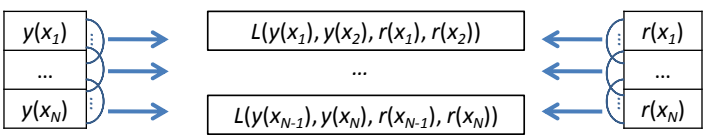
Pointwise Approaches



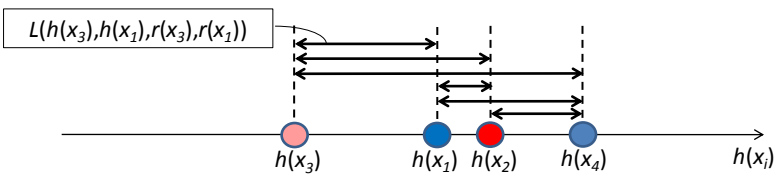
- Learn from each document *in isolation*
- Standard reductions
  - Regression: relevance/loss is a real-valued variable
  - Classification: relevance is categorical
  - Ordinal regression
- Example: Ordinal regression [Krammer-Singer '01, Chu-Keerthi '05]
  - Loss is based on *thresholds* separating the classes, minimization based on *margin/regret*



Pairwise Approaches

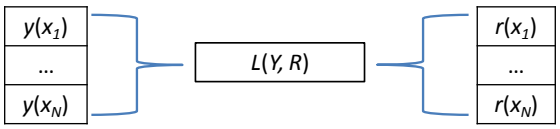


- Pointwise approaches ignore relative positions of documents
- Alternative: view ranking as *pairwise classification*
- Overall loss is aggregated over pairwise predictions
- Pairwise agreements = AUC (for binary labels)
- Natural reduction for incorporating preference training data

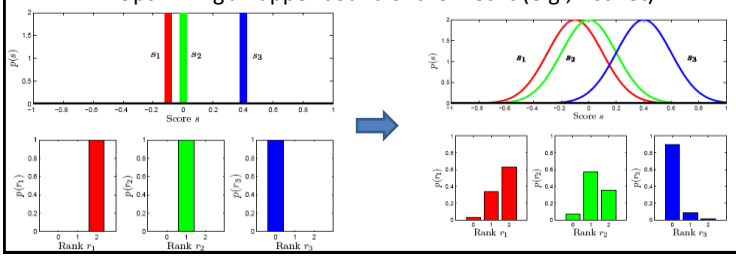


[Cohen *et al.* '98]
[Herbrich *et al.* '00]
[Freund *et al.* '03]
[Joachims '05]
[Burges *et al.* '05]
[Burges *et al.* '06]
[Cao *et al.* '07]

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, BoltzRank)
  - Optimizing an upper bound of the metric (e.g., ListNet)



[Cao *et al.* '07]
[Xia *et al.* '08]
[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: what is the right objective?

Tutorial Overview

1. Introduction
   - Machine learning background via text classification
   - IR challenges for machine learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance and learning to rank
   - Learning from user behavior
   - Query-related prediction tasks
   - Social media
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs and other optimization frameworks for retrieval
   - Learning complex structured outputs
4. Summary and Bibliography

Common IR+ML Paradigm

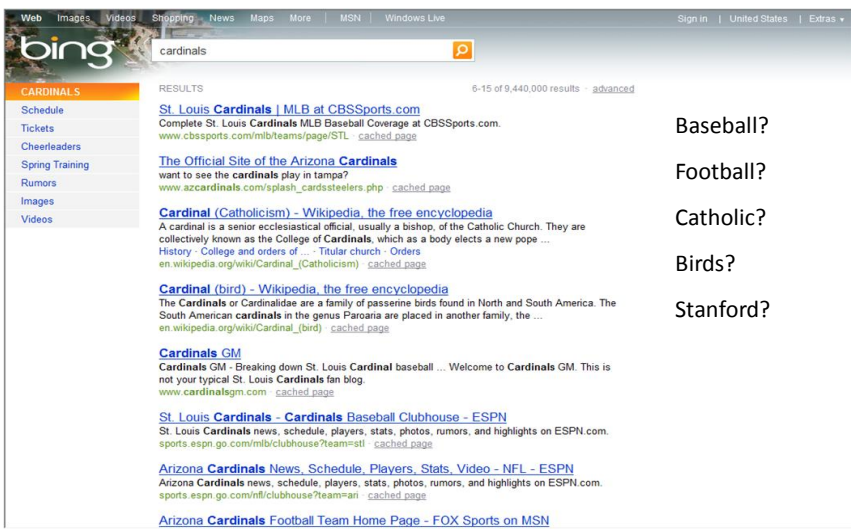
- Construct hypothesis about what matters to the user.
- Translate performance measure to optimization criterion.
- Formulate a hypothesis regarding connection between data and measure to optimize.
- Mine for patterns that match hypothesis → add as feature for ranker, convert to ground truth
- Mine for patterns that violate hypothesis → revise optimization criterion, treat as weighted optimization.

- This section will present a series of examples focused on web search that fall into these paradigms. General lessons apply to any IR task.

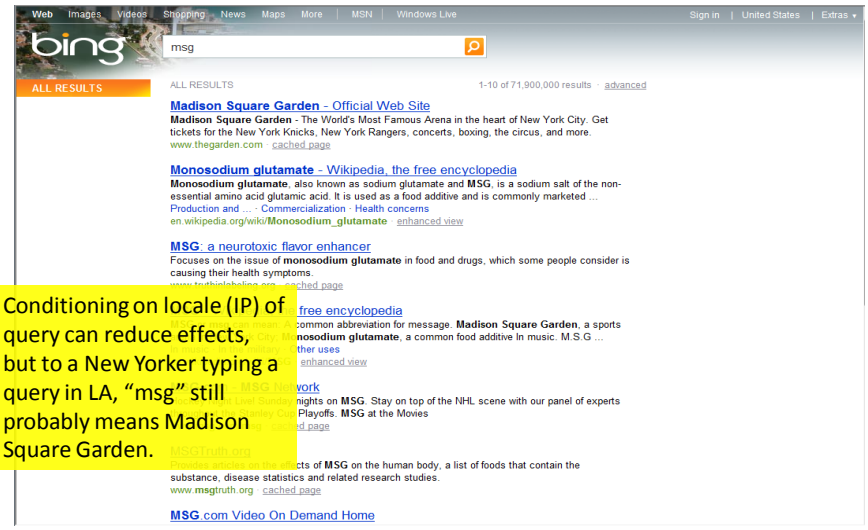
Hypothesis: Search is simply many classification tasks.

- 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

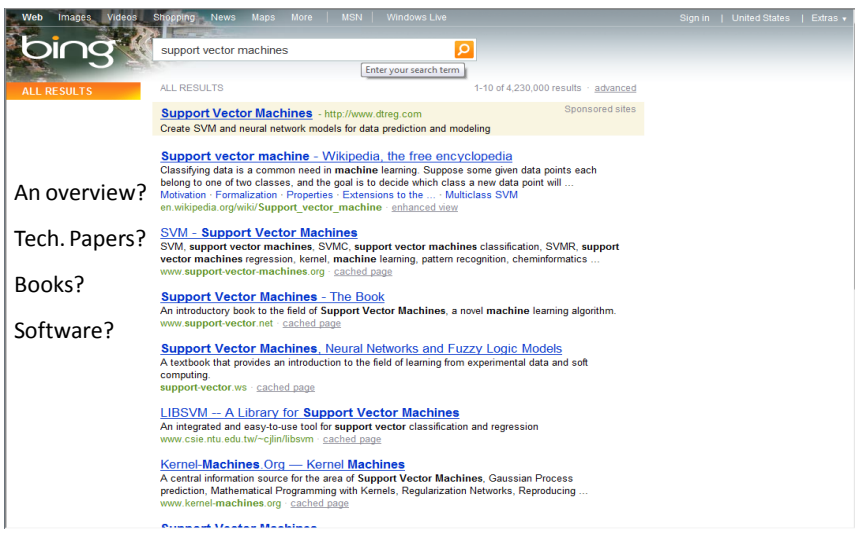


Locale & Ambiguity



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



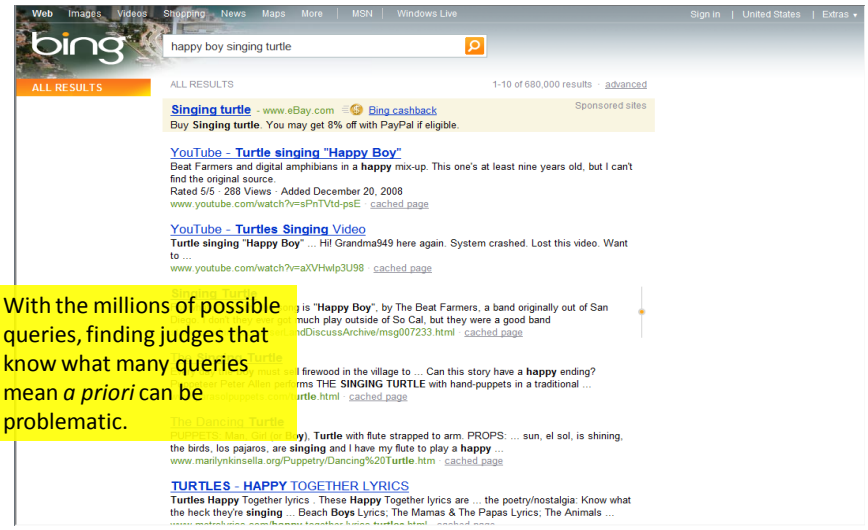
An overview?

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)?
- Need an approach that can generalize to breadth of users, queries, and needs.

Learning From User Behavior

- Implicit measures carry a relevance signal (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
- Two common ways to integrate
  - Use as a feature in combination with expert judgments.
  - Directly optimize for measure based on user behavior.

Interpreting a Click

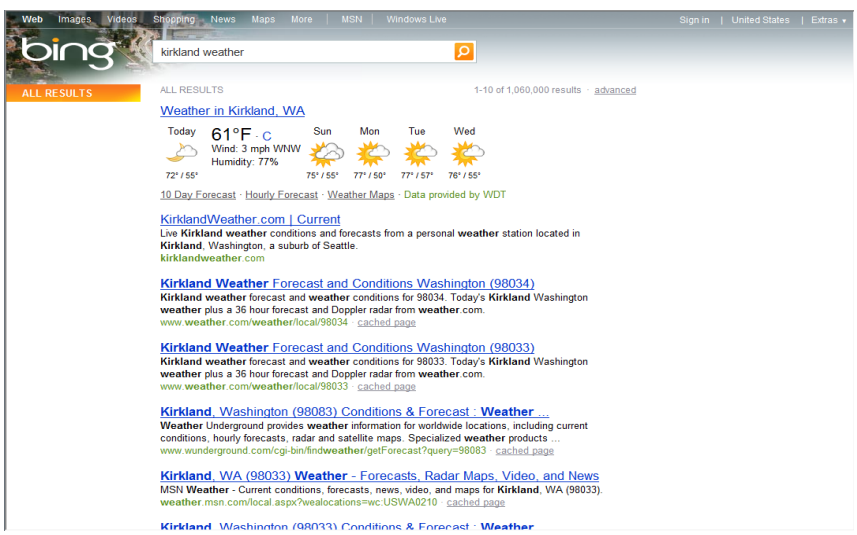
- Hypothesis: A click is a judgment that the clicked item is relevant.
- Rank Bias – the more highly ranked an item, the more likely it is to get a click regardless of relevance.
  - When order is reversed, higher ranked items still typically get more clicks. (Joachims *et al*, SIGIR '05).
- Clicks are 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?



Other Common Kinds of User Behavior

- Abandonment – user does not click on a search result.
  - Usually implies irrelevant?
  - Radlinski *et al*. (ICML '08) minimize abandonments using multi-armed bandits for queries that are repeated.
- 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?

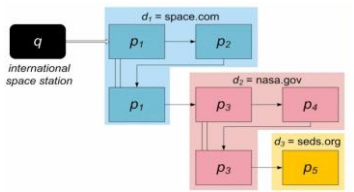
Last Click as Relevance Vote

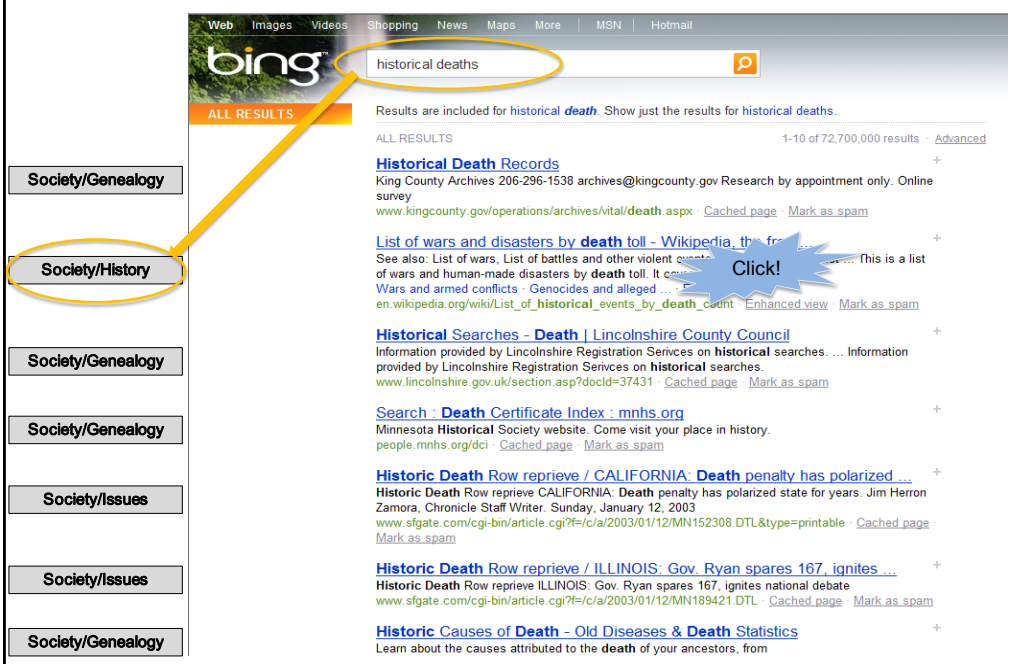
- Hypothesis: user continues until they find a document that satisfies their needs.
- Goal: separate click as relevance from presentation order effects.
- Predict clicks on urls presented in order B,A when trained from A,B order (Craswell *et al.*, WSDM '08)
- Possible explanatory models
  - Baseline – 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.

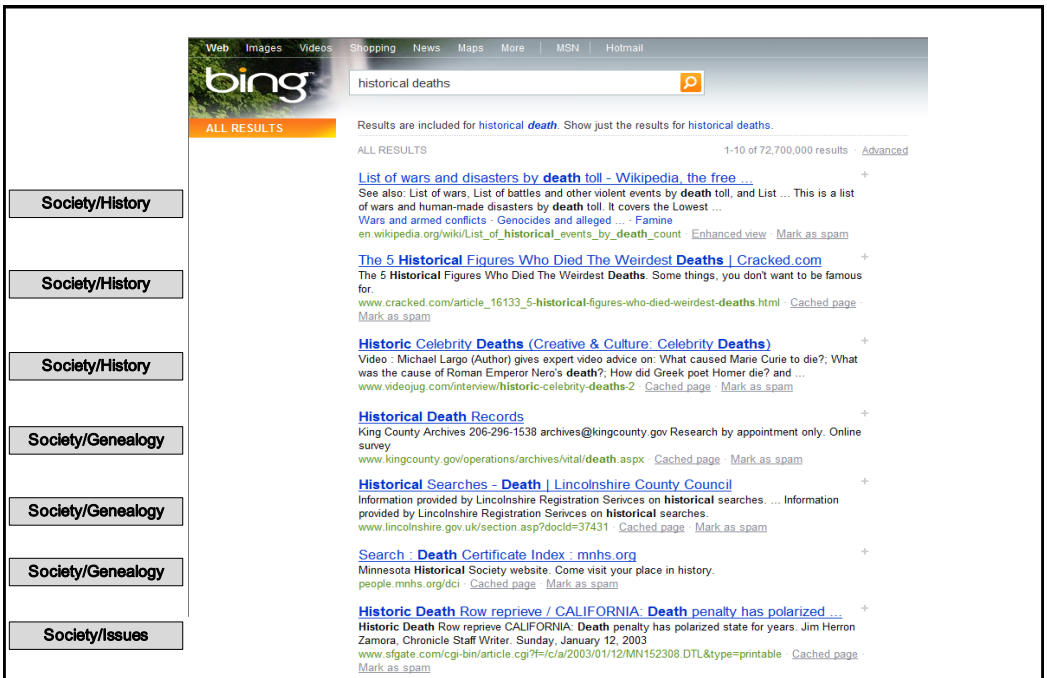
[Craswell *et al.*, WSDM '08]

Sessions and Browsing

- Clearly, a click for a single query is not only the only indication of relevance.
- 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.







Click as Desire for Class of Information

[Bennett *et al.*, 2010]

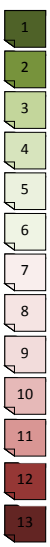
- Assign URL classes.
- Estimate query class distribution using result URLs.
  - Weight class by function of click evidence [Bennett *et al*., 2010]
  - Alternatively weight class by ranking or ranker score [Broder *et al.,* 2007; Gabrilovich *et al*., 2009]
- Introduce ranker features derived from query, url distributions and their matching.
- Ranker learns weights to integrate this signal with others.
- Improved rankings over head, tail, short, and long queries.

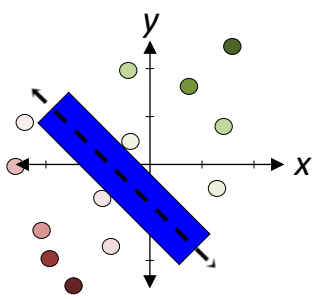
Is Collection-Building for Evaluation and Training the Same?

- State of IR for Evaluation
  - 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
  - Carterette *et al.* (ECIR 2009) present overview and study of current methods of min labeling effort for *evaluation*.
- Do these lessons apply for gathering training data?
  - New developing area (Aslam *et al.,* SIGIR 2009).

Applying IR Evaluation Techniques to Training Data Collection

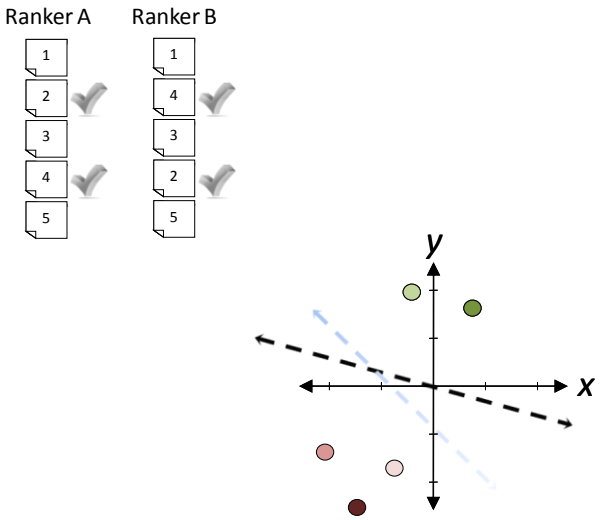
Optimal Ranking





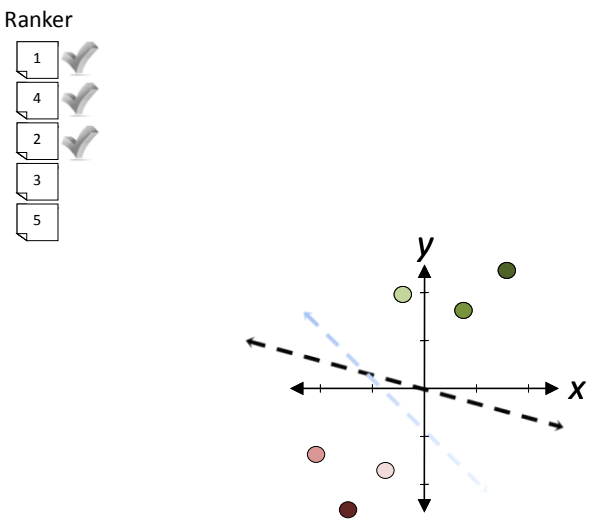
67

Sample High Ranking Points that Differ



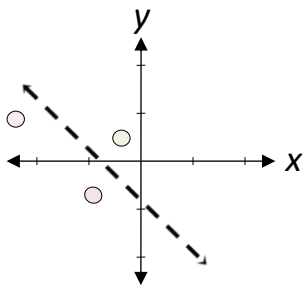
68

Sample High Ranking Points



69

Learning Fully Constrained By Points near Decision Boundary



70

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 imply different choices of machine learning tools as well as where research might contribute 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.
    - Can extend training data with "gold-standard" consensus rankings.
  - Mechanical Turk
    - See Omar Alonso's tutorial on *Crowdsourcing for Relevance Evaluation* this afternoon (also paper here, Alonso *et al.,* 2008).
- Other Human Computation Questions
  - How many times to relabel in context of Mechanical Turk (Sheng & Provost, KDD '08).
  - Selecting the most appropriate expert (Donmez *et al.,* KDD '09).
- Implication for learning: model as weighted optimization of "expert" labeled, crowdsourced, implicit user-feedback, and unlabeled data.

Learning from User Behavior Summary

- Reality — use both implicit and explicit judgments as a source of information. A common approach is to use:
  - Explicit judgments as ground truth
  - Click-derived signals as features.
  - Other approaches where the objective is cast in terms of clicks, reformulations, abandonments, *etc*. [cf. Das Sarma *et al.,* KDD '08].
- Emerging models optimize multi-criterion objectives

  [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.

Break: 15 min

Tea and biscuits!

Tutorial Overview

1. Introduction
   - Machine learning background via text classification
   - IR challenges for machine learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance and learning to rank
   - Learning from user behavior
   - Query-related prediction tasks
   - Social media
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs and other optimization frameworks for retrieval
   - Learning complex structured outputs
4. Summary and Bibliography

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: Which corpus best answers this query?
   - Vertical search
   - Federated search: save access costs, reduce noise.
3. Reformulation risk: Will query reformulation be effective?
   - Big win if we could accurately predict when and how to rewrite/expand any given query.

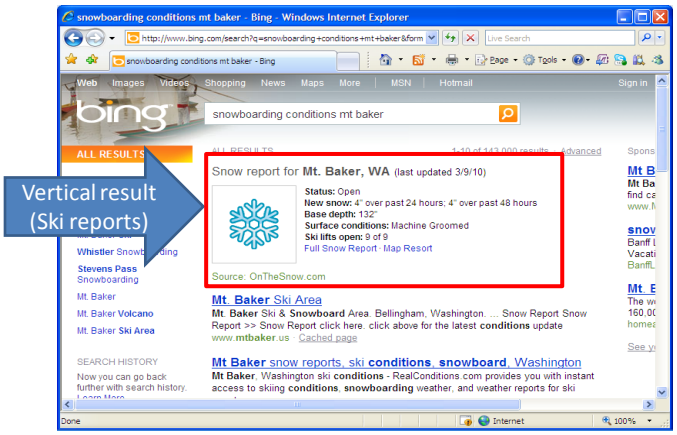
Learning to predict query difficulty

- 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

Vertical search: Exploiting domain-specific sub-collections



Increasing number of specialized resources:

- News
- Images
- Videos
- Sporting events
- Job listings
- Current ski reports
- Movies
- Company summaries
- Artist profiles…

Machine learning problem:
Predict relevant verticals (*if any*) for a given query

Predicting relevant verticals (if any) for a given query

Input features

Enhanced query representation

Key phrases
Correlation between named entity, vertical type
`"abba lyrics" → music`

Rule-based triggers
Map intent class to vertical (regex, dictionary)
`{product review, product} → shopping`

Geographical entities
Location-related verticals: local, travel, maps.
p(*Entity* | *Query*).
`"Eiffel Tower" → point_of_interest`

Direct vertical query logs
Estimate language model θ_V: vertical *V* or Web (no-relevant)
Feature: *p*(Query | θ_V)

Vertical corpora
Issue query to a vertical collection (smaller representative corpora), look at doc rankings

- KL divergence between query and collection model
- Soft Modified ReDDE: expected number of relevent docs to query. (account for diff between original and sampled sizes)
- Compared to class distribution of general web docs

Learning model

Logistic regression: multiclass classifier (one vs all).

- No-relevant-vertical uses confidence of 18 binary vertical classifiers. Combined with argmax plus threshold.
- Goal: maximize single vertical precision

Evaluation data

- 25195 queries
- Labeled with 0 – 6 relevant verticals.
- 70% of queries had 0 (26%) or 1 (44%) vertical.
- Some ambiguous queries.

[Sources of evidence for vertical selection J. Arguello, F. Diaz, J. Callan, J.-F. Crespo SIGIR 2009]

Tutorial Overview

1. Introduction
   - Machine learning background via text classification
   - IR challenges for machine learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance and learning to rank
   - Learning from user behavior
   - Query-related prediction tasks
   - Social media
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs and other optimization frameworks for retrieval
   - Learning complex structured outputs
4. Summary and Bibliography

Social media and machine learning

- Explicit sources of social structure and behavior
  - Twitter, Facebook
  - Blogs, wikis
- Implicit socially-driven applications
  - Anchor text, social tagging systems
  - Collaborative filtering and recommendation systems
  - User session information
- Modeling and inference problems:
  - Propagation and influence [e.g. Twitter, blogs]
  - Expertise and trust [e.g. Wikis, blogs, newsgroups, QA systems]
  - Social phenomena and emergent social behavior
  - User needs, motivations and goals

Collaborative filtering (CF) and recommendation systems

- Task: Predict new ratings on items from older ones.
  - Learning to rank may be an attractive alternative framework
  - Equally address all users, not just the heavy raters we have data for.
  - CF models try to capture interactions between users and items that produce the different ratings.
- ML objective: Minimize Root Mean-Squared Error (RMSE) of predicted ratings against actual ratings.
- ML challenges:
  - Not enough ratings per user (in total, or after binning in various ways)
    - 99% of possible ratings are missing: user only rates small portion of movies
    - Users that don't rate much are harder to predict
  - Must also account for rating biases for users or items:
    - Some users tend to give higher/lower ratings than others (Joe the critic)
    - Some movies receive higher ratings than others (Titanic)
  - High-dimensional feature space: many thousands of users, movies

NetFlix winner: BellKor's Pragmatic Chaos

[Koren KDD 2009]

$$B_{ui}(t) = \mu + b_u(t) + b_i(t)$$

- μ is average rating over all users and movies, B_{ui} is observed deviation of user u and movie i from average.
- Adding each time-changing baseline improves accuracy:
  - + Item's popularity b_i can change over time (e.g. actor in new film)
    - Not daily, but over extended periods: stationary part + time changing part
  - + Users change baseline b_u over time
    - e.g. context of other ratings given recently, household rater can change over time. Can change daily.
  - + Selection bias during `bulk' rating: Frequency of ratings on specific day explains significant portion of score variability.
    - Favored movies are marked as favorites, disliked ones are not rated

Key ML conclusions from NetFlix effort

- Accurate treatment of main effects is likely to be at least as significant as fancy new models.
  - Focus on better baseline predictors.
- Temporal dynamics can have a big effect on accuracy compared to designing more complex algorithms.
  - Modeling changes in ratings over time is critical.
- Use sophisticated blending schemes for multiple individual predictors.
  - New individual features unlikely to make the difference.
  - Simultaneous adjustment to multiple predictors.
  - Gradient Boosted Decision Trees were highly effective

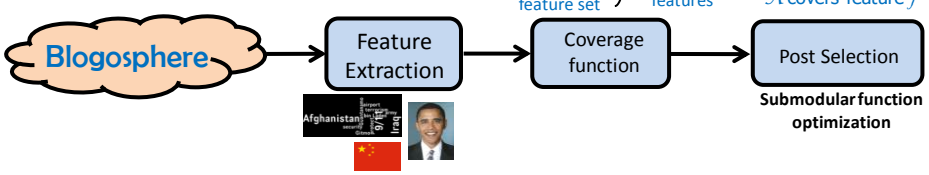
[Koren KDD 2009]

Filtering Example: Blog Post Recommendation

- Objectives: coverage + personalization
- Solution: submodular optimization, learning [El-Arini *et al.*]

$$F(\mathcal{A}) = \sum_{f \in \mathcal{U}} w_f \, cover_{\mathcal{A}}(f)$$

feature set; weights on features; probability that set $\mathcal{A}$ covers feature f



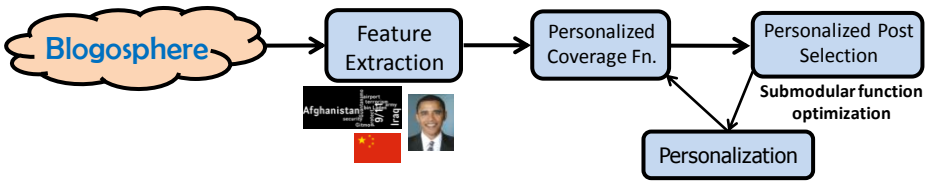
[El-Arini *et al.* '09]

Filtering Example: Blog Post Recommendation

- Objectives: coverage + personalization
- Solution: submodular optimization, learning [El-Arini *et al.*]

$$F_\pi(\mathcal{A}) = \sum_{f \in \mathcal{U}} \pi_f w_f cover_{\mathcal{A}}(f)$$

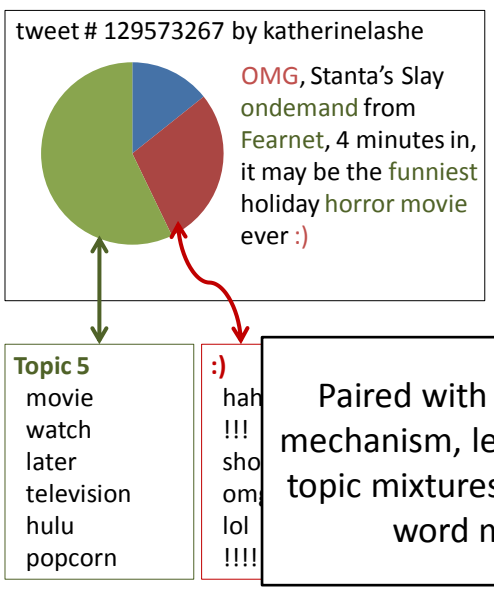
User preference



- Learning personalized preferences π_f from feedback (•,•)

[El-Arini *et al.* '09]

Modeling Twitter feeds with LDA-family topic models



Labeled LDA assumes each tweet's words generated by some topic's word distribution.

Tweets can use all latent topics and some labeled ones.

Paired with an inference mechanism, learns per-tweet topic mixtures and per-topic word mixtures

rom features
n tweet or
s, questions,
gs, wefollow tags)

Twitter topic visualization example

Analysis of two users: @w3c (left) and @oprah (right). Recent posts from each user are mapped into Labeled LDA dimensions. The usage of dimensions from substance (top row), status (second row), social (third row), or style (bottom row) categories is shown in the vertical bars, with Twitter's average usage shown in the center. Common words in selected dimensions from each category are shown as word clouds. Word size is proportional to the word's frequency in that dimension globally, and word shade is proportional to the word's frequency in the user's recent tweets. Light gray words are unused in recent tweets.

[Courtesy of Ramage, Dumais, Liebling ICWSM 2010]

Tutorial Overview

1. Introduction
   - Machine learning background via text classification
   - IR challenges for machine learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance and learning to rank
   - Learning from user behavior
   - Query-related prediction tasks
   - Social media
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs and other optimization frameworks for retrieval
   - 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:

$$E[R(ad_i)] = p(click \mid ad_i) \cdot CPC(ad_i)$$

- CPC depends on auction type; in 2nd price auctions $CPC(ad_i) \leq bid(ad_i)$
- *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*]

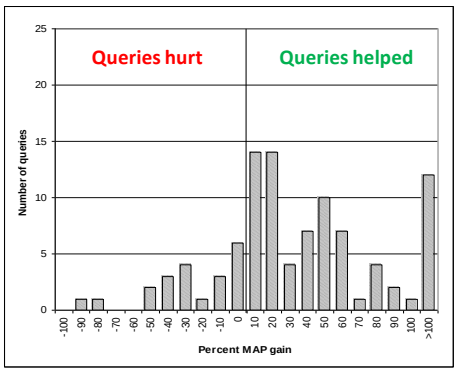
Learning for Personalized Advertising

- 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
  - Identifying "urban car shoppers", "expecting parents who refinance", …
- Biggest challenges: *privacy* and *scale*
  - Scale: distributed learning via MapReduce

Tutorial Overview

1. Introduction
   - Machine learning background via text classification
   - IR challenges for machine learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance and learning to rank
   - Learning from user behavior
   - Query-related prediction tasks
   - Social media
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs and other optimization frameworks for retrieval
   - Learning complex structured outputs
4. Summary and Bibliography

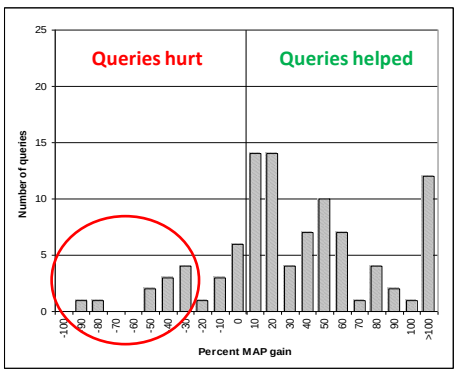
Current query expansion methods work well <u>on average</u>…



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
- It is hard to integrate multiple task constraints for increasingly complex estimation problems:
  - Personalization, computation constraints, implicit/explicit 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

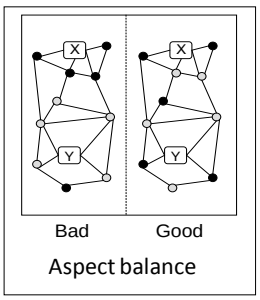
Select terms as a set, not individually, for a more balanced query model

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 robust optimization 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)

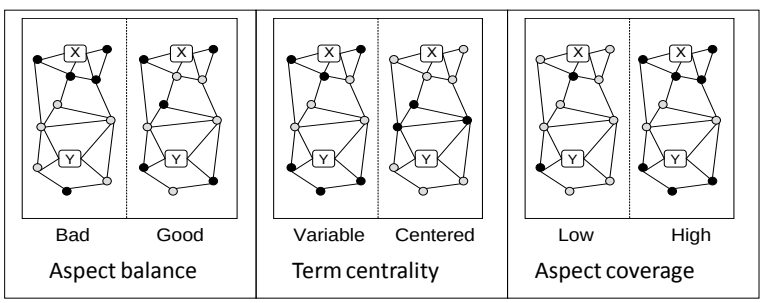


Two-term query: "X Y"

[Collins-Thompson, NIPS 2008]

Query expansion as an optimization problem

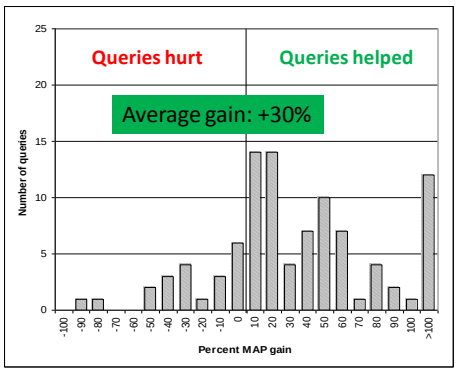
[Collins-Thompson NIPS 2008, CIKM 2009]



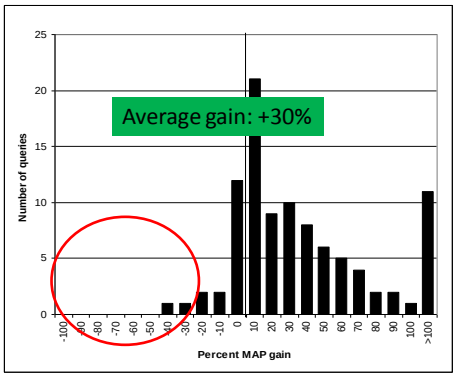
QMOD algorithm

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

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



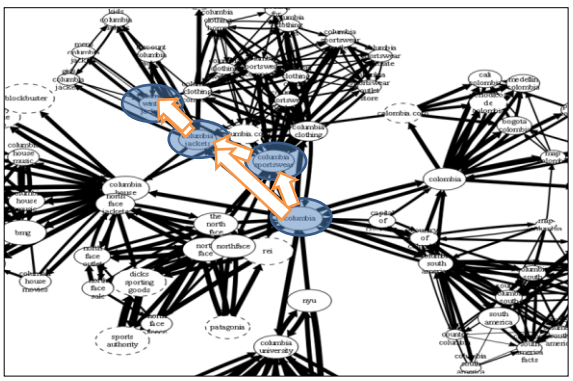
Query expansion:
Current state-of-the-art method



Robust version

Optimization frameworks for query recommendation

- Model query behavior of users with a query flow graph
- A sequence of queries is a path on this graph
- Assign scores to queries
- Derive the *expected utility* achieved by a given path
- Add shortcuts to nudge the reformulation paths of users to get paths of higher expected utility



[columbia] → [columbia sportswear] → [columbia jackets]

[columbia] → [columbia winter jackets]

[Anagnostopoulos et al., WSDM 2010 paper]

[Graphic courtesy of F. Radlinski, MSRC]

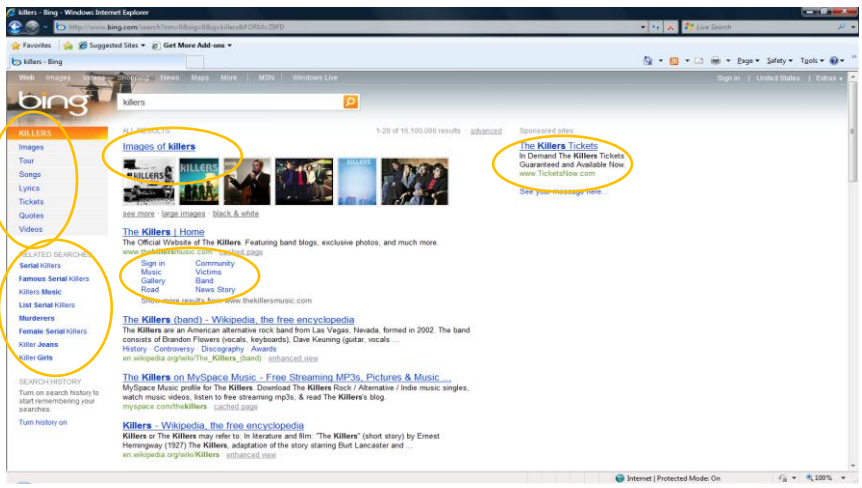
Future directions

- Optimization frameworks have broad applicability in information retrieval scenarios
  - Query expansion, query reformulation, when to personalize, resource selection, resource-constrained IR algorithms, …
- 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. Introduction
   - Machine learning background via text classification
   - IR challenges for machine learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance and learning to rank
   - Learning from user behavior
   - Query-related prediction tasks
   - Social media
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs and other optimization frameworks for retrieval
   - Learning complex structured outputs
4. Summary and Bibliography

IR: Beyond the Ranked List



Structure Is 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 needed 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 queries
  - Types of sources (e.g. small and large news outlets)

Maximal Marginal Relevance

(Goldstein & Carbonell, SIGIR '98)

- For similarity function $sim(d, d')$ and 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* $\mid d_1$ 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. Introduction
   - Machine learning background via text classification
   - IR challenges for machine learning
2. Recent Advances at IR-ML Crossroads
   - Modeling relevance and learning to rank
   - Learning from user behavior
   - Query-related prediction tasks
   - Social media
3. Emerging Opportunities for Learning in IR
   - Online advertising
   - Risk-reward tradeoffs and other optimization frameworks for retrieval
   - Learning complex structured outputs
4. Summary and Bibliography

IR / ML Summary

- Basic IR paradigm: satisfying users' information needs

- At its core, IR studies retrieval-related ***prediction tasks***
  - ML studies prediction – provides algorithms/frameworks for IR
- Much of IR is driven by focus on measurement and improvement against user satisfaction.
  - ML helps formalize the objectives and develop principled solutions

IR Increasingly Relies on ML

- General shift from heuristics to formal statistical models.
- More recent shift to discriminative algorithms, for which "traditional" models provide input features.
- Salient properties of IR tasks push state-of-the-art in ML:
  - Massive amounts of documents
    - Need for scalable ML algorithms
  - Nearly infinite variety in expressing an information need.
    - ML for new, interesting objectives
  - Huge amount of user-generated data
    - ML methods that incorporate different types of supervision

Summary of ML paradigm for IR tasks

- Features: Choose representation of relevance signals/features
  - Implicit: f (behavior), e.g. clicks, dwells, session query trails
  - Explicit: from a source tasked with evaluation, e.g. assessors, crowdsourcing
  - Techniques for dealing with weak, noisy, or sparse features
- Objective: Map IR performance goal to appropriate ML objective/loss function
  - Incurred loss over training set
  - Maximum (expected) likelihood of the observed data
  - (Smoothed) IR performance metric
  - Regularization to encode prior knowledge of what `good models' look like:
    - Model complexity
    - Feature sparsity
  - Risk/variance: robustness to uncertainty in data via minimizing worst-case loss or regret
- Training: Tailor the learning models to the types of labeled data available
  - Pairwise preferences, explicit labels, similarity sets
  - Supervised, semi-supervised, unsupervised

Selected future directions

- Methods that use multiple sources of evidence about relevance:
  - Clicks, expert judgments, human computation labels, …
  - How to combine into a single optimization criterion?
  - Handling non-uniform noise in ground truth labels
- Prediction tasks
  - Predicting relevance for real-time ranking or assistance tasks
    - e.g. "instant answers" or query suggestion
  - Predicting clicks
    - e.g. On a result, an ad, a query suggestion, …
  - Predicting when to personalize or contextualize
    - e.g. based on session context, location
  - Predicting query performance
    - Reduce the risk of reformulation operations, increase quality of filter set

Selected future directions

- New optimization problems/objectives
  - Risk-aware IR algorithms: e.g. preventing worst-case behavior
  - Ranking effectively and efficiently
  - Identifying value of components in structured retrieval.
    - Diversification, novelty, over results, suggested queries, …
- Building evaluation collections
  - Does using an evaluation corpus for training introduce systematic bias?
  - Can the evaluation corpus be de-biased for use in learning using principled methods?

Pointers to Machine Learning and Optimization Tools

- MLOSS is a great overall index of all packages:
  http://mloss.org/software/
- Weka is probably the best all-purpose package:
  http://www.cs.waikato.ac.nz/ml/weka/
- Support Vector Machines (classification & regression, structured output, large-scale data, ranking)
  svmLight http://svmlight.joachims.org/
  libSVM http://www.csie.ntu.edu.tw/~cjlin/libsvm/
- CVX: High-level convex programming (Matlab)
  http://www.stanford.edu/~boyd/cvx/

Pointers to Data Resources

- Yahoo! Learning to Rank Challenge
  - http://learningtorankchallenge.yahoo.com/
- LETOR
  - Learning to rank data: http://research.microsoft.com/en-us/um/beijing/projects/letor/index.html
- CLUEWEB
  - Large-scale Web crawl with relevance judgments http://boston.lti.cs.cmu.edu/Data/clueweb09/
- 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 (Bennett *et al.* WWW09)
  - 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)

Tutorial slides are available online:

http://research.microsoft.com/ecir-2010-mlir-tutorial

Current version: 3/30/2010 5:07 AM

Bibliography

- 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'd)

- 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. "Reducing the risk of query expansion via robust constrained optimization". *Proceedings of the Eighteenth International Conference on Information and Knowledge Management (CIKM 2009)*. ACM. Hong Kong. pg. 837-846.
  - K. Collins-Thompson. "Robust model estimation methods for information retrieval". Ph.D. thesis, Carnegie Mellon University, 2008.

Bibliography (cont'd)

- Query difficulty /query performance prediction/vertical search
  - 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
  - J. Arguello, F. Diaz, J. Callan, J.-F. Crespo, Sources of evidence for vertical selection. SIGIR 2009.

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 and collaborative filtering
  - Y. Koren, Collaborative Filtering with Temporal Dynamics, KDD 2009.
  - 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.
- Social media
  - D. Ramage, S. Dumais, D. Liebling. Characterizing Microblogs with Topic Models. ICWSM 2010.

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:

  Susan Dumais, Dan Liebling, Filip Radlinski, Dan Ramage