CS 589 Fall 2021 Lecture 3

IR Evaluation
(Pseudo)-relevance feedback

Monday 6:30-9:00
Babbio 122

All zoom links in Canvas

photo: https://www.scubedstudios.com/information-retrieval/

Review of Lecture 2: Probabilistic Ranking Principle

- Given the query q, all documents should be ranked by their probability of relevance $p(rel = 1|q, d)$
  - RSJ model:
    - Doesn't leverage the TF information
    - Rely on relevance judgment
  - BM25 model
    - Estimate the probability using the 2-Poisson model conditioned on the ***eliteness of a word to a document***
    - Obtain a parameter-free model by approximating the 2-Poisson model

Review of Lecture 2: LM-based retrieval model

- Given the query q, rank all documents by the probability $p(q|d)$ of generating q from d
  - Estimate p(q|d) based on the i.i.d. assumption
  - Estimate p(w|d) based on the (unigram) statistical language model
    - MLE: $$p(w|d) = \frac{count(w,d)}{|d|}$$
    - Jelinek-Mercer smoothing: $$p_s(w_i|d) = \lambda p(w_i|d) + (1-\lambda)p(w_i|C)$$
    - Dirichlet smoothing:

$$score^{Dir}(q,d) = \sum_{w_i, w_i \in d, p(w_i|\hat{\theta}_q)} p(w_i|\hat{\theta}_q) \log\left(1 + \frac{count(w_i,d)}{\mu p(w_i|C)}\right) + \log\frac{\mu}{\mu + |dl|}$$

Pop quiz (LM-based retrieval model)

- Suppose d1={“the”, “more”, “the”, “better”}, d2={“the”, “pizza”}, d3={“just”, “do”, “it”}, what is the probability of p(“the”|d1) and p(“the”|d3)? Suppose alpha_d=0.1

$$p(w_i|d) = \begin{cases} p_{seen}(w_i|d) & \text{if } w_i \text{ is seen in d} \\ \alpha_d p(w_i|C) & o.w. \end{cases}$$

- A: 0.5, 0.3333
- B: 0.5, 0.03333
- C: 0.4831, 0.3125
- D: 0.4831, 0.03125

Pop quiz (LM-based retrieval model)

- In the feedback-based retrieval model, if the 2 feedback documents are d1={“the”, “more”, “the”, “better”}, d2={“the”, “pizza”}, q={“pizza”, “hut”}, what is the feedback retrieval model? The feedback retrieval model is computed below, and lambda = 0.9

$$\theta_q \leftarrow \lambda\theta_q + (1-\lambda)\theta_q^F$$

- A: pizza: 0.5, hut: 0.5
- B: pizza: 0.46, hut: 0.45
- C: pizza: 0.46, hut: 0.45, the: 0.05, more: 0.016, better: 0.016

Lecture 3

- Basic evaluation metrics for an IR system
  - Precision/recall
  - MAP, MRR, NDCG
- The Cranfield evaluation methodology
  - Pooling strategy
- Online evaluation, A/B test
- Relevance feedback

Information retrieval evaluation

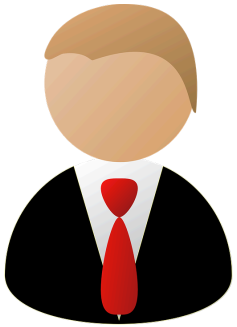
- After learning CS589, you graduate and join Bing

Information retrieval evaluation

- After learning CS589, you graduate and join Bing

Information retrieval evaluation

- How to know
    - If your search engine has outperformed another search engine
    - If your search engine performance has improved compared to the last quarter?



Metrics for a good search engine

- Return what the users are looking for
- Return results fast
- Users likes to come back

- Relevance, CTR = click thru rate
- Latency
- Retention rate

Rank-based measurements

- Binary relevance
  - Precision@K
  - Mean average precision (MAP)
  - Mean reciprocal rank (MRR)

- Multiple levels of relevance
  - Normalized discounted cumulative gain (NDCG)

Precision of retrieved documents

- Fraction of retrieved docs that are relevant

$$precision = \frac{\#relevant\&retrieved}{\#retrieved}$$

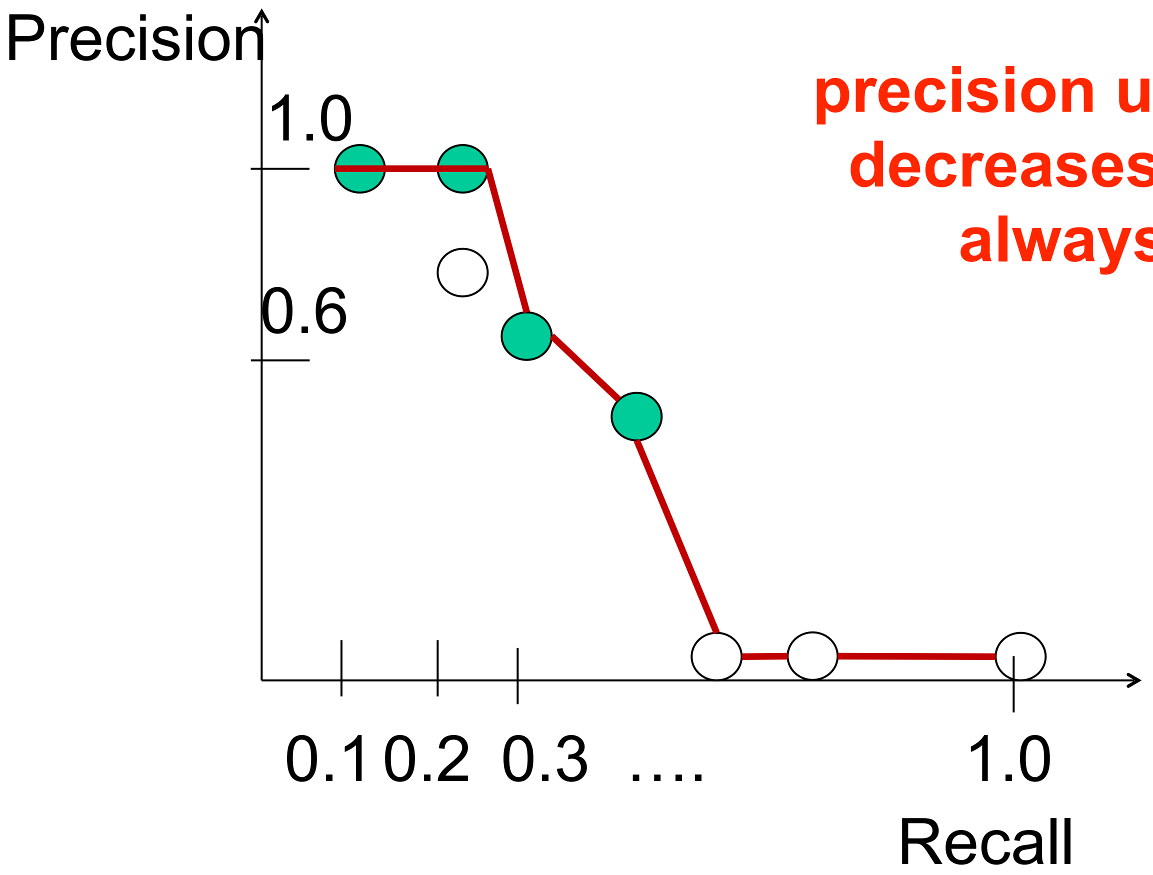
- Fraction of relevant documents that are retrieved

$$recall = \frac{\#relevant\&retrieved}{\#relevant}$$

Precision-recall curve

	Precision	Recall
+	1/1	1/4
+	2/2	2/4
−		
−		
+	3/5	3/4
−		
−		
+	4/8	4/4
−		
−		

(1/1 + 2/2 + 3/5 + 4/8) / 4



precision usually decreases (not always)

Slides from UIUC CS598

Average precision

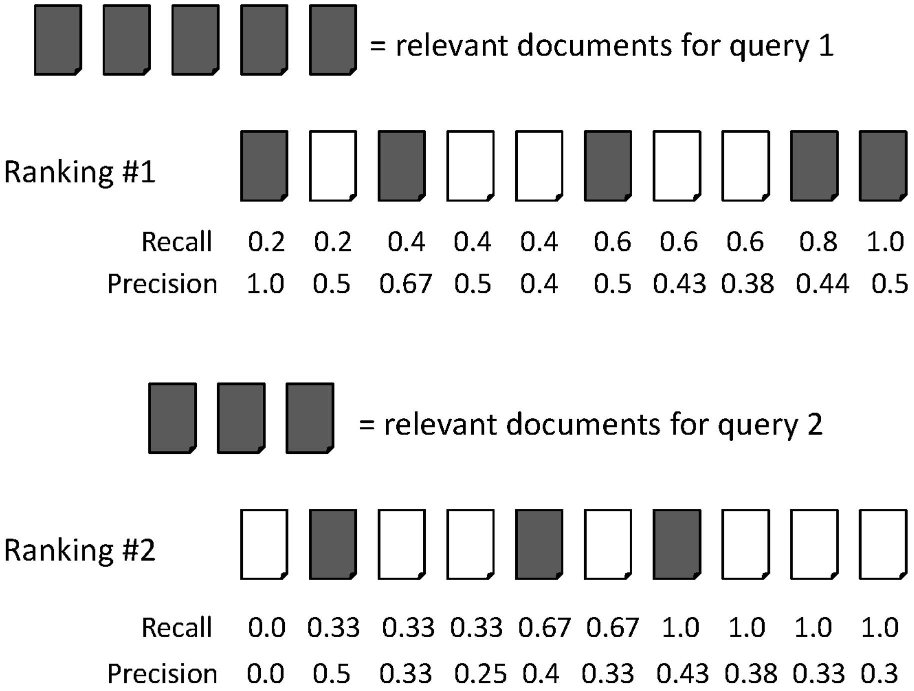
- Consider the rank position of each ***relevant and retrieved*** doc
  - $K_1, K_2, \ldots K_R$
- Compute Precision@K for $K = K_1, K_2, \ldots K_R$
- Average precision:

 # retrieved documents

$$\text{AveP} = \frac{\sum_{k=1}^{n} (P(k) \times \text{rel}(k))}{\text{number of relevant documents}}$$

 # relevant documents, not # retrieved documents

MAP



Suppose there are 5 relevant documents for both query 1 and 2

This value = #relevant documents, not # retrieved relevant documents (why?)

$$\mathit{average\ precision\ query\ 1} = (1.0 + 0.67 + 0.5 + 0.44 + 0.5)\boxed{/5} = 0.62$$

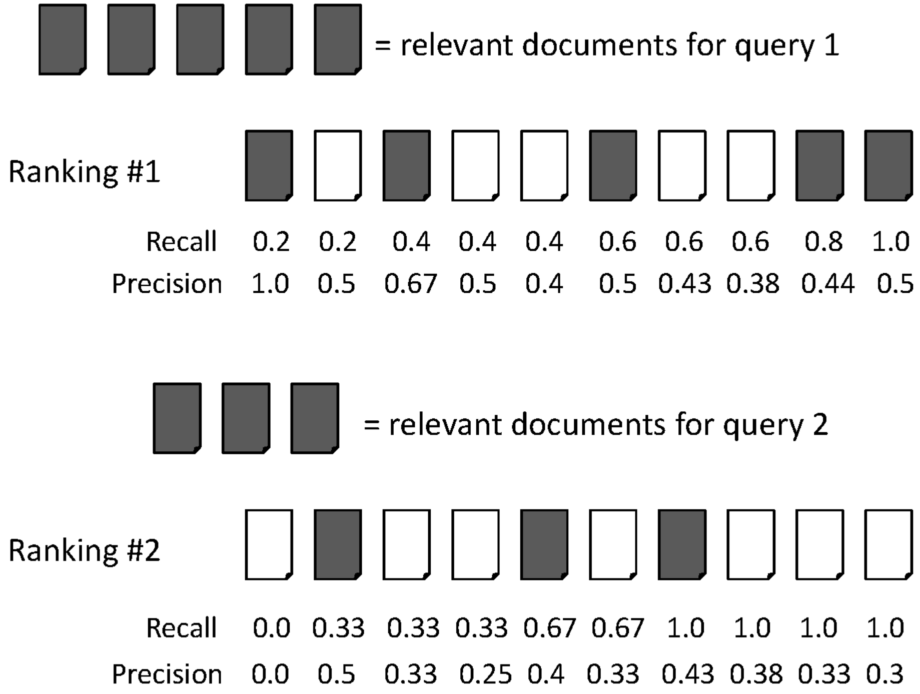
$$\mathit{average\ precision\ query\ 2} = (0.5 + 0.4 + 0.43)/5 = 0.266$$

$$\mathit{mean\ average\ precision} = (0.62 + 0.266)/2 = 0.443$$

Slides from Stanford CS276

Mean reciprocal rank

- Measure the effectiveness of the ranked results
  - Assume users are only looking for one relevant document
-



RR = 1.0 / (1.0 + rank_1)

p starts from 0

$$MRR = 1/2 \times (1 + 1/2) = 0.75$$

Slides from UVA CS4780

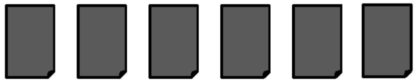
Beyond binary relevance

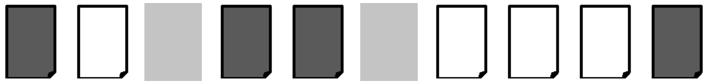
- Discounted cumulative gain (DCG)

- Popular measure for evaluating web search and related tasks

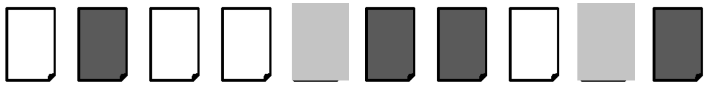
- Information gain-based evaluation (economics)
  - For each relevant document, the user has gained some information
  - The higher the relevance, the higher gain
  - The gain is discounted when the relevant document appears in a lower position

Discounted cumulative gain (DCG)

 = the relevant documents

Ranking #1 

2 0 1 2 2 1 0 0 0 2

Ranking #2 

0 2 0 0 1 2 2 0 1 2

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

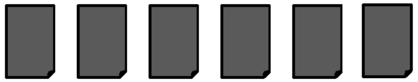
p starts from 1

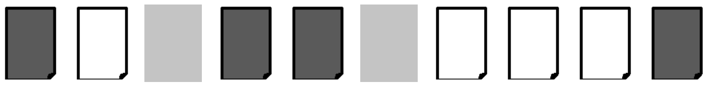
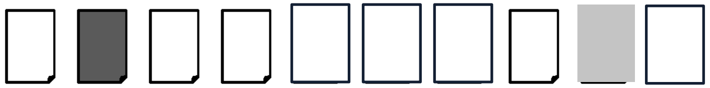
$$DCG@4\,query\ 1 = \frac{2^2 - 1}{\log_2 2} + \frac{2^1 - 1}{\log_2 4} + \frac{2^2 - 1}{\log_2 5} = 4.79$$

$$DCG@4\,query\ 2 = \frac{2^2 - 1}{\log_2 3} = 1.89$$

Why normalizing DCG?

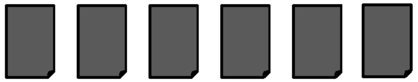
- If we do not normalize DCG, the performance will be biased towards systems that perform well on queries with larger DCG scales

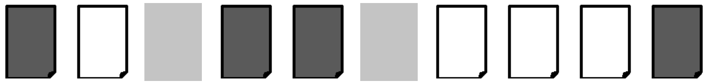
 = the relevant documents

		system A	system B
"TV"	 2 0 1 2 2 1 0 0 0 2	DCG=4.79	DCG=5.79
"clothing"	 0 2 0 0 1 2 2 0 1 2	DCG=1.89	DCG=1.39
		avg=3.34	avg=3.59

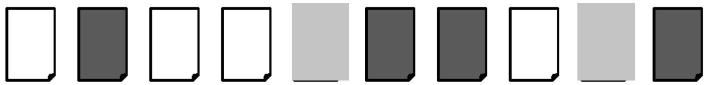
bias towards B

Normalized Discounted cumulative gain (nDCG)

 = the relevant documents

Ranking #1 

2 0 1 2 2 1 0 0 0 2

Ranking #2 

0 2 0 0 1 2 2 0 1 2

$$nDCG_4 = (4.79/7.68 + 1.89/7.68)/2 = 0.43$$

$$IDCG@4\ query\ 1 = \frac{2^2-1}{\log_2 2} + \frac{2^2-1}{\log_2 3} + \frac{2^2-1}{\log_2 4} + \frac{2^2-1}{\log_2 5} = 7.68$$

$$IDCG@4\ query\ 2 = \frac{2^2-1}{\log_2 2} + \frac{2^2-1}{\log_2 3} + \frac{2^2-1}{\log_2 4} + \frac{2^2-1}{\log_2 5} = 7.68$$

Relevance evaluation methodology

- Offline evaluation:
  - Evaluation based on annotators' annotation (explicit)
    - TREC conference
    - Cranfield experiments
    - Pooling
  - Evaluation based on user click through logs (implicit)

- Online evaluation
  - A/B testing

Text REtrieval Conference (TREC)

- Since 199
- Relevanc
  - The re g
- Different t
  - Web
  - Quest
  - Microb

<top>
<num> Number: 794

<title> pet therapy

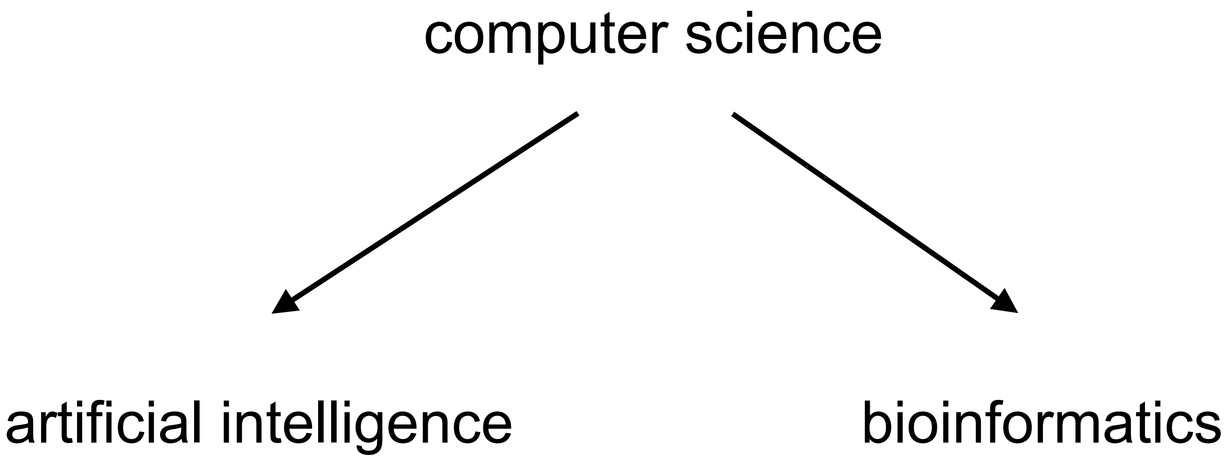
<desc> Description:
How are pets or animals used in therapy for humans and what are the benefits?

<narr> Narrative:
Relevant documents must include details of how pet- or animal-assisted therapy is or has been used. Relevant details include information about pet therapy programs, descriptions of the circumstances in which pet therapy is used, the benefits of this type of therapy, the degree of success of this therapy, and any laws or regulations governing it.

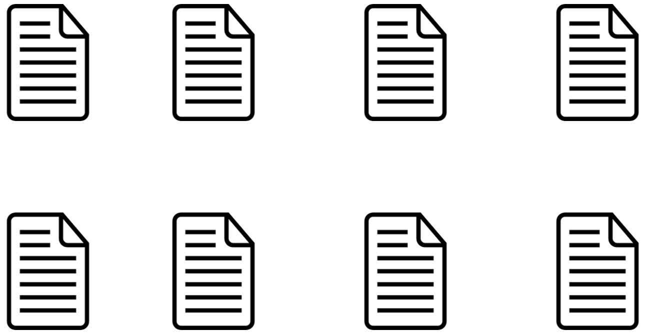
</top>

The Cranfield experiment (1958)

- Imagine you need to help users search for literatures in a digital library, how would you design such a system?



query **=** *“subject = AI & subject = bioinformatics”*



system 1: the Boolean retrieval system

The Cranfield experiment (1958)

- Imagine you need to help users search for literatures in a digital library, how would you design such a system?

Document-term matrix

	intelligence	book	the	cat	artificial	dog	business
Doc1	0	1	3	1	0	1	0
Doc2	1	0	0	0	0	0	1
query	1	0	1	0	1	0	0

query **=** *"artificial intelligence"*

bags of words representation

system 2: indexing documents by lists of words

The Cranfield experiment (1958)

- Basic ingredients
  - A corpus of documents (1.4k paper abstracts)
  - A set of 225 queries and their information needs
  - Binary relevance judgment for each (q, d) pair
  - Reuse the relevance judgments for each (q, d) pair

query = “best phone”, time = 2012,
relevance = 1

Nokia

query = “best phone”, time = 2022,
relevance = 0

Scalability problem in human annotation

- TREC contains 225 x 1.4k = 315k (query, documents) pairs
- How to annotate so many pairs?
- Pooling strategy
  - For each of K system, first run the system to get top 100 results
  - Annotate the union of all such documents

Evaluation based on user click through logs

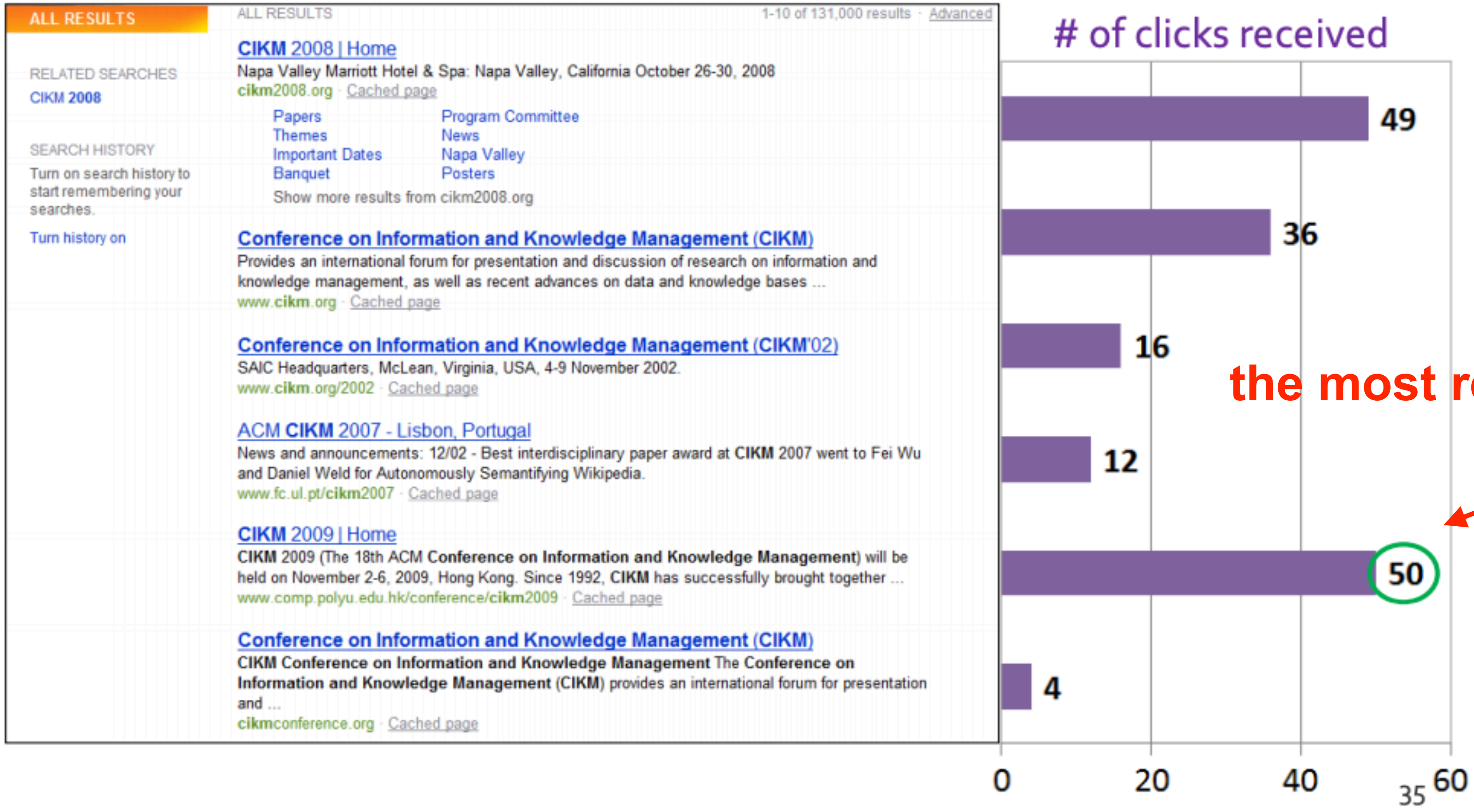
- TREC style relevance judgment
  - Explicit relevance judgment
  - Difficult to achieve large scalability
  - Relevance is **fixed**

- Relevance judgment using user clicks
  - Implicit relevance judgment
  - Effortless relevance judgment at a large scale
  - Relevance is **fixed, (assume relevance judgment stays the same upon reranking)**

Evaluation based on user click through logs

- Click logs for "CIKM"

slides from Stanford CS276



the most relevant document

Evaluation based on user click through logs

- System logs the users engagement behaviors:
  - Time stamp
  - Session id
  - Query id, query content
  - Items viewed by the user (in sequential order)
  - Whether each item has been clicked by the user
  - User’s demographic information, search/click history, location, device
  - Dwell time, browsing time for each document
  - Eye tracking information

Evaluation based on user click through logs

- Click logs are stored in large tables
- Using SQL to extract a subset of query logs

Session Id	*Timestamp*	*Action*	*Action details*
....			
123457	1388494920	search	Query ='flawless'
123457	1388494980	click	Page Id = '755'
123457	1388495060	reformulation	Query ='flawless beyonce' => Reformulation = 'beyonce'
123457	1388495115	click	Page Id = '170'
123458	1388495415	search	Query ='cikm conference'
123456	1388361661	reformulation	Query ='cikm conference' => Reformulation = '2014'
123456	1388361720	click	Page Id = "45"

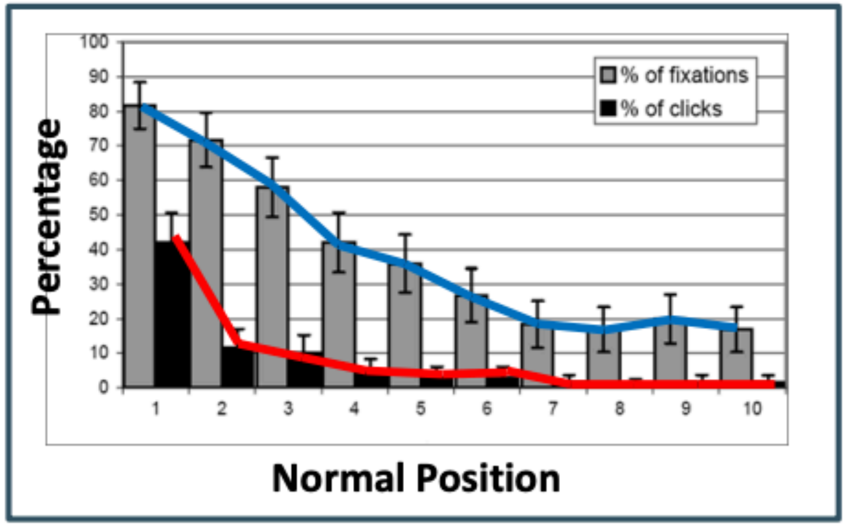
Online evaluation methodology

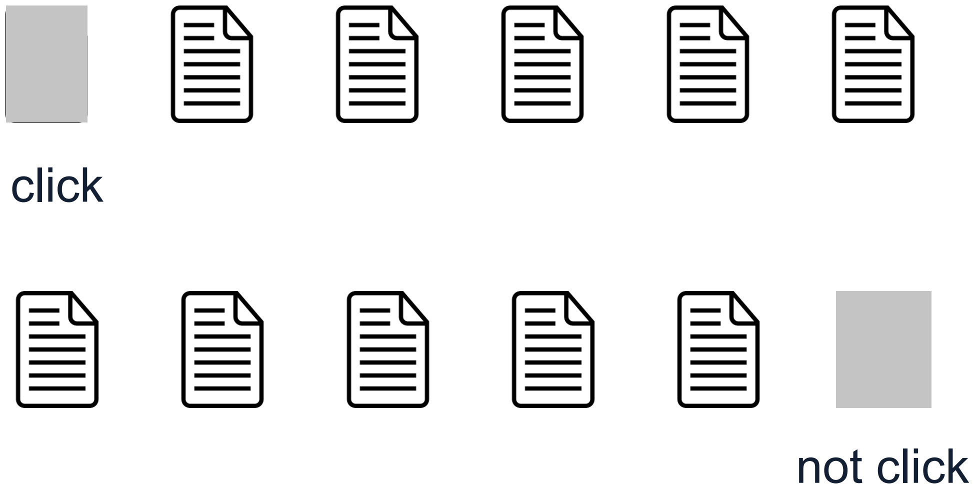
- Assumption made by offline evaluation
  - After reranking, relevance judgment stays the same
  - Which is not true…

- Relevance judgment is dynamic, subject to user bias
  - Bias based on positions
  - Preference shifting over time, location
  - Decoy effects
    - Change in preference between two options when also presented with a third option that is asymmetrically dominated

Position bias [Craswell 08]

- Position bias
  - Higher position receives more attention
  - The same item gets lower click in lower position





An experimental comparison of click position-bias models

Position bias [Craswell 08]

- Which model captures the position bias?
  - Baseline hypothesis: no position bias
  - Mixture hypothesis: click is due to a mixture of relevance and constant bias:

$$c_{di} = \lambda r_d + (1 - \lambda) b_i$$

  - Cascade model: a linear traversal through the ranking, and that documents below a clicked result are not examined

$$c_{di} = r_d \prod_{j=1}^{i-1} (1 - r_{docinrank:j})$$

Position bias [Craswell 08]

- Which model captures the position bias?
  - Baseline hypothesis: no position bias
  - Mixture hypothesis: click is due to a mixture of relevance and constant bias:

$$c_{di} = \lambda r_d + (1 - \lambda) b_i$$

  - Cascade model: a linear traversal through the ranking, and that documents below a clicked result are not examined

$$c_{di} = r_d \prod_{j=1}^{i-1} (1 - r_{docinrank:j})$$

Position bias [Craswell 08]

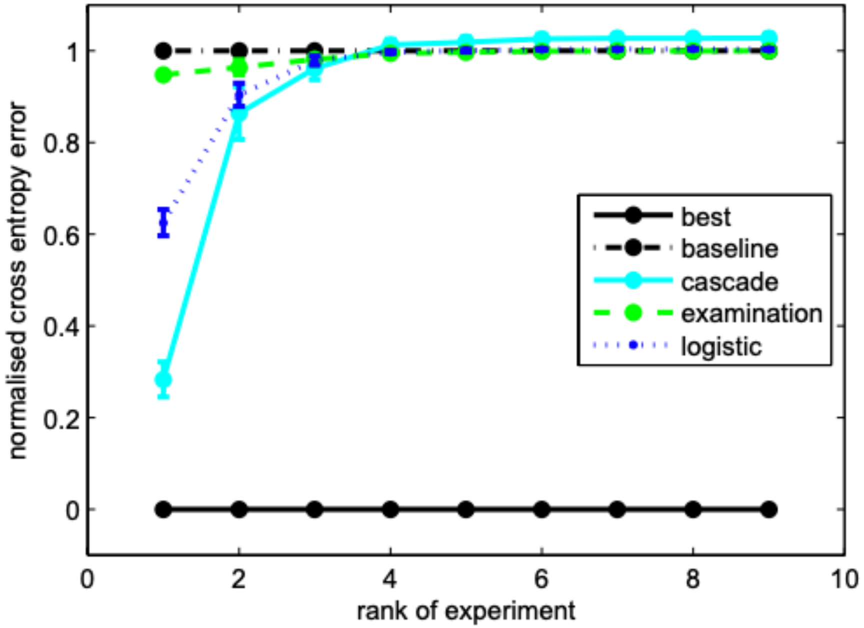
- Controlled experiment:
  - Show document A and B at position m and m+1
  - Flip the two documents
  - Four outcomes: A clicked or skipped, B clicked or skipped

- Test the three hypothesis by comparing their probability with the true click probability:

$$CE(hyp) = -\sum_{i=1}^{4} p_{hyp}(outcome_i) \log p_{true}(outcome_i)$$

Position bias [Craswell 08]

- Result of CE:
  - At upper rank, the baseline model works better
  - At lower rank, the cascade model works the best



An experimental comparison of click position-bias models

Decoy effects

vs

$400, 20G

$500, 30G

$550, 20G

~~click probability = 0.3~~

~~click probability = 0.4~~

click probability = 0.5

click probability = 0.5

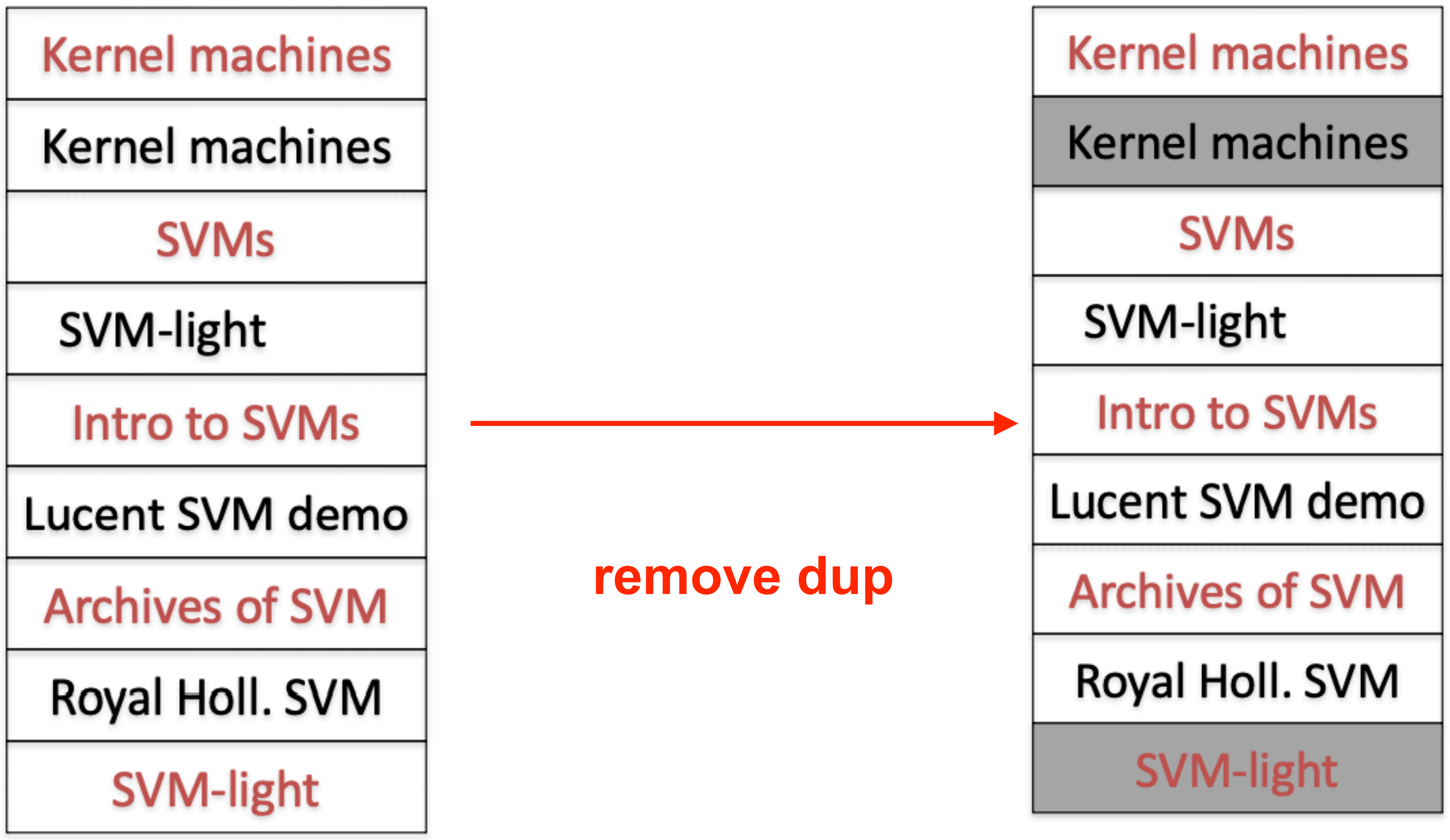
Online evaluation methodology

- Evaluation by actually having the system deployed and observe user response
  - Less scalable
  - A/B testing

Query: [support vector machines]

Ranking A	Ranking B
Kernel machines	Kernel machines
SVM-light	SVMs
Lucent SVM demo	Intro to SVMs
Royal Holl. SVM	Archives of SVM
SVM software	SVM-light
SVM tutorial	SVM software

Interleaving



A clicks = 3, B clicks = 1

Statistical significance testing

- How sure can you be that an observed difference doesn't simply result from the particular queries you chose?

Experiment 1

Query	System A	System B
1	0.20	0.40
2	0.21	0.41
3	0.22	0.42
4	0.19	0.39
5	0.17	0.37
6	0.20	0.40
7	0.21	0.41
Average	0.20	0.40

Experiment 2

Query	System A	System B
1	0.02	0.76
2	0.39	0.07
3	0.16	0.37
4	0.58	0.21
5	0.04	0.02
6	0.09	0.91
7	0.12	0.46
Average	0.20	0.40

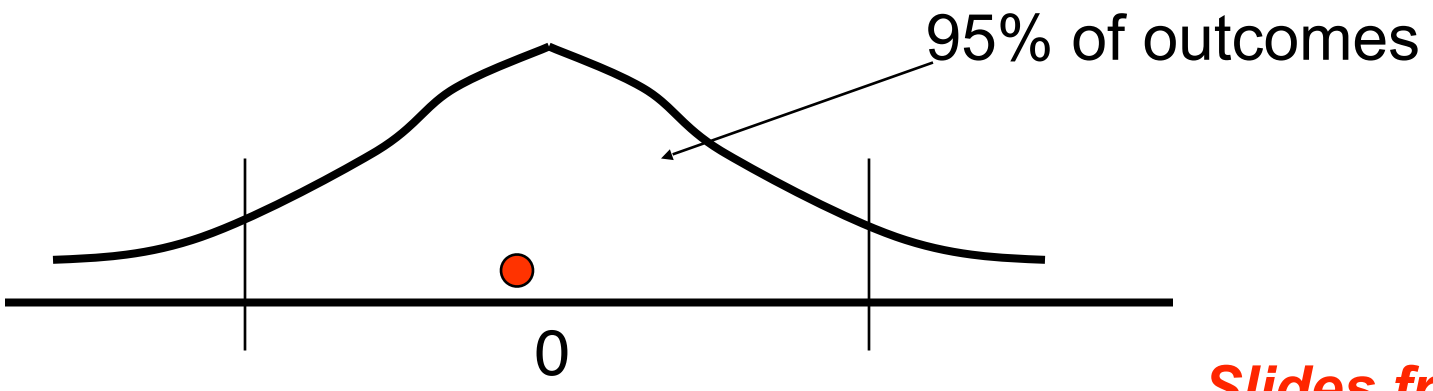
Slides from UIUC CS598

Statistical significance testing

Query	System A	System B	Sign Test	Wilcoxon
1	0.02	0.76	+	+0.74
2	0.39	0.07	-	- 0.32
3	0.16	0.37	+	+0.21
4	0.58	0.21	-	- 0.37
5	0.04	0.02	-	- 0.02
6	0.09	0.91	+	+0.82
7	0.12	0.46	-	- 0.38
Average	0.20	0.40	p=1.0	p=0.9375

Wilcoxon test:

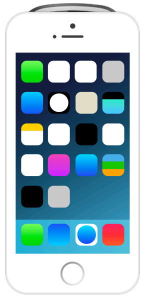
$$W = \sum_{i=1}^{N} [\operatorname{sgn}(x_{2,i} - x_{1,i}) \cdot R_i]$$



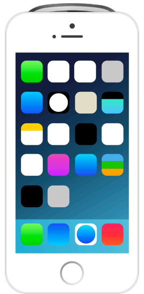
Slides from UIUC CS598

Retrieval feedback in session search

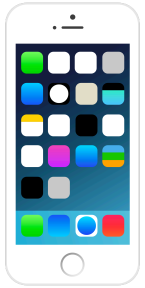
query = “best phone”



$400, 20G,
Nokia



$500, 30G,
Nokia



$600, 40G,
iphone

Does the user prefer lower priced phone, or high end phones? Larger storage, better camera?

session 2

observed click

Rocchio feedback

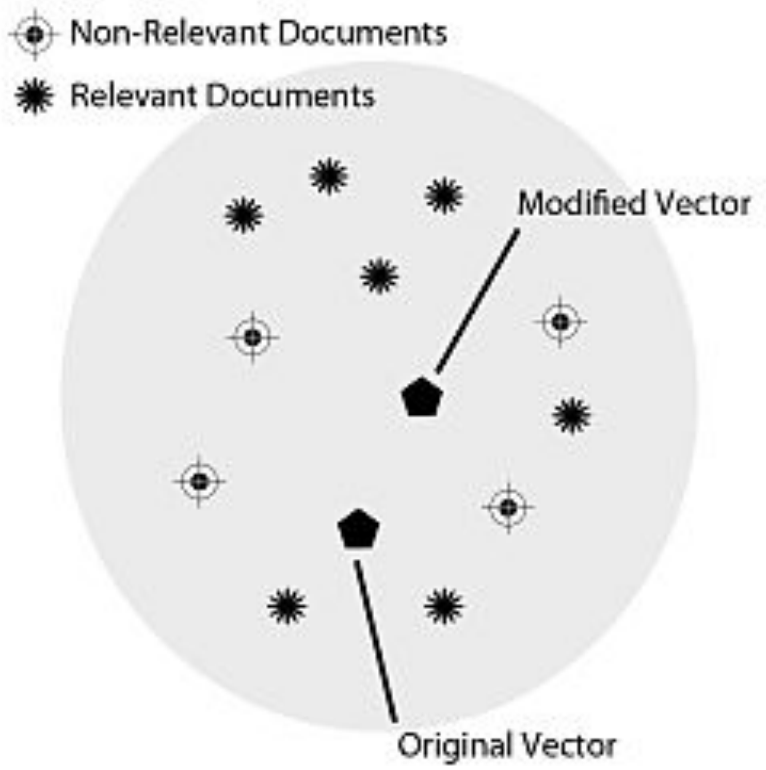
- Feedback for vector-space model

$$q_F = \alpha q + \frac{\beta}{|D_r|} \sum_{d_r \in D_r} d_r - \frac{\gamma}{|D_n|} \sum_{d_n \in D_n} d_n$$

rel docs

non-rel docs

beta >> gamma



- Rocchio's practical issues
  - Large vocabularies (only consider important words)
  - Robust and effective
  - Requires relevance feedback

Pseudo-relevance feedback

- What if we do not have relevance judgments?
  - Use the top retrieved documents as “pseudo relevance documents”

- Why does pseudo-relevance feedback work?

query = “fish tank”

www.petsmart.com › fish › aquariums

Fish Tanks & Aquariums | PetSmart

125 Items - Shop the latest **fish tanks** and aquariums at PetSmart to find interesting ways showcase your favorite fish. Browse large and small tanks, fresh and ...

Tanks, Aquariums & Nets · Fish Tanks for Sale: Discount · Fish Aquariums

Relevance feedback in RSJ model

$$O(rel=1|q,d) \stackrel{rank}{=} \sum_{w_i=1} \log \frac{\alpha_i(1-\beta_i)}{\beta_i(1-\alpha_i)}$$

(Robertson & Sparck Jones 76)

$$\alpha_i = p(w_i=1|q,rel=1) = \frac{count(w_i=1,rel=1)+0.5}{count(rel=1)+1}$$

Probability for a word to appear in a relevant doc

$$\beta_i = p(w_i=0|q,rel=0) = \frac{count(w_i=0,rel=0)+0.5}{count(rel=0)+1}$$

Probability for a word to appear in a non-relevant doc

(Pseudo)relevance feedback language model

$$score^{JM}(q,d) = \sum_{w_i, w_i \in d, p(w_i|\hat{\theta}_q)} \boxed{p(w_i|\hat{\theta}_q)} \log\left(1 + \frac{(1-\lambda)count(w_i,d)}{\lambda p(w_i|C)}\right)$$

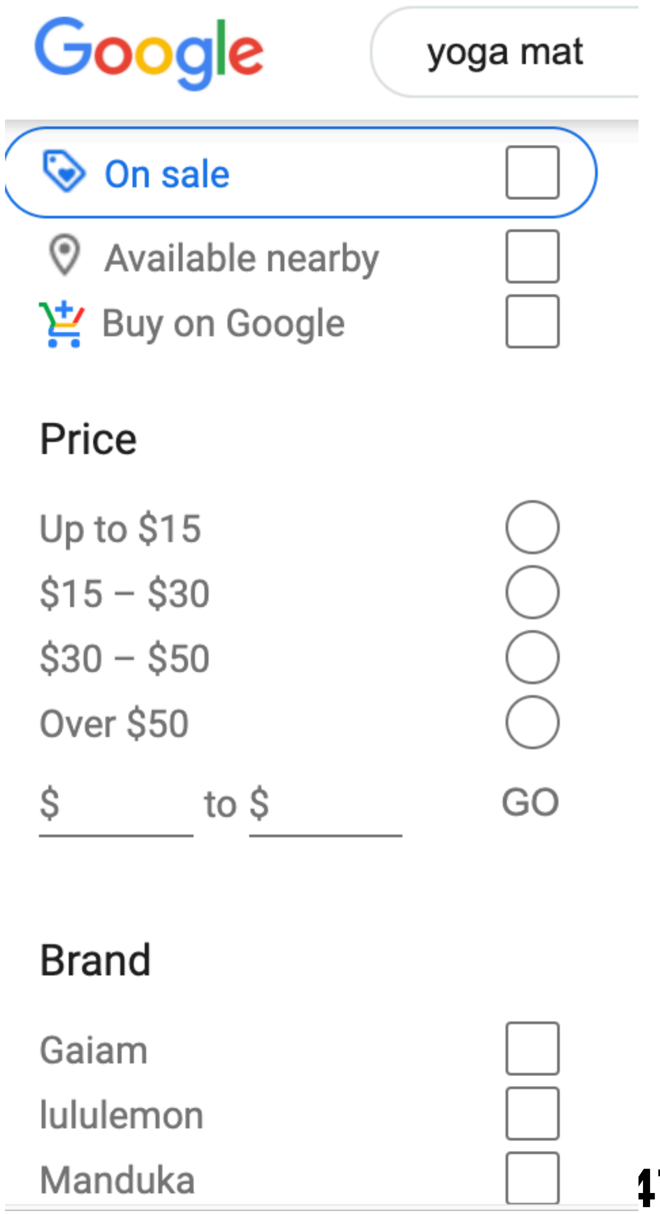
$$p(w_i|q) = \frac{count(w_i,q)}{|q|}$$

sparsity

$$\begin{tikzcd}
d \arrow[r, "\text{get document model}"] & \theta_d \arrow[dr] & \\
q \arrow[r] & \theta_q \arrow[r] & -D(\theta_q|\theta_d) \arrow[d, "\text{retrieve}"] \\
\theta_q \leftarrow \lambda\theta_q + (1-\lambda)\theta_q^F \arrow[ur] & \theta_q^F \arrow[l] & d_1, d_2, \cdots, d_n \arrow[l, "\text{infer } \theta_q^F \text{ w/ EM algo}"']
\end{tikzcd}$$

Model-based feedback in the language modeling approach to information retrieval

Query expansion



Query reformulation

- Query expansion/reformulation techniques
  - Using manually created synonyms
  - Using automatically derived thesaurus
  - Using query log mining

Word	Nearest neighbors
absolutely	absurd, whatsoever, totally, exactly, nothing
bottomed	dip, copper, drops, topped, slide, trimmed
captivating	shimmer, stunningly, superbly, plucky, witty
doghouse	dog, porch, crawling, beside, downstairs
makeup	repellent, lotion, glossy, sunscreen, skin, gel
mediating	reconciliation, negotiate, case, conciliation
keeping	hoping, bring, wiping, could, some, would
lithographs	drawings, Picasso, Dali, sculptures, Gauguin
pathogens	toxins, bacteria, organisms, bacterial, parasite
senses	grasp, psyche, truly, clumsy, naive, innate

Summary

- Know how to compute Prec/recall, MAP, NDCG, MRR
  - Try implementing them on your own for HW1 and reproduce the results

- Know how the Cranfield experimental methodology and pooling works

- Know how the feedback retrieval model works

not click