# CS 589 Fall 2020

# Information Retrieval Evaluation

# Retrieval Feedback

**Instructor: Susan Liu**
**TA: Huihui Liu**

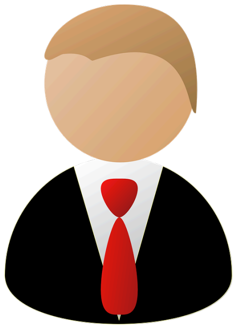
**Stevens Institute of Technology**

# Information retrieval evaluation

- Last lecture: basic ingredients for building a document search engine

- You graduate and join Bing

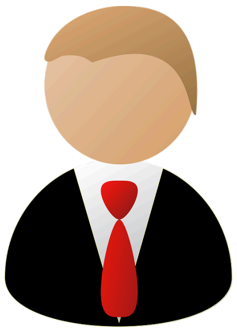
Beat Google!



# Information retrieval evaluation

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

Beat Amazon!



# 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

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

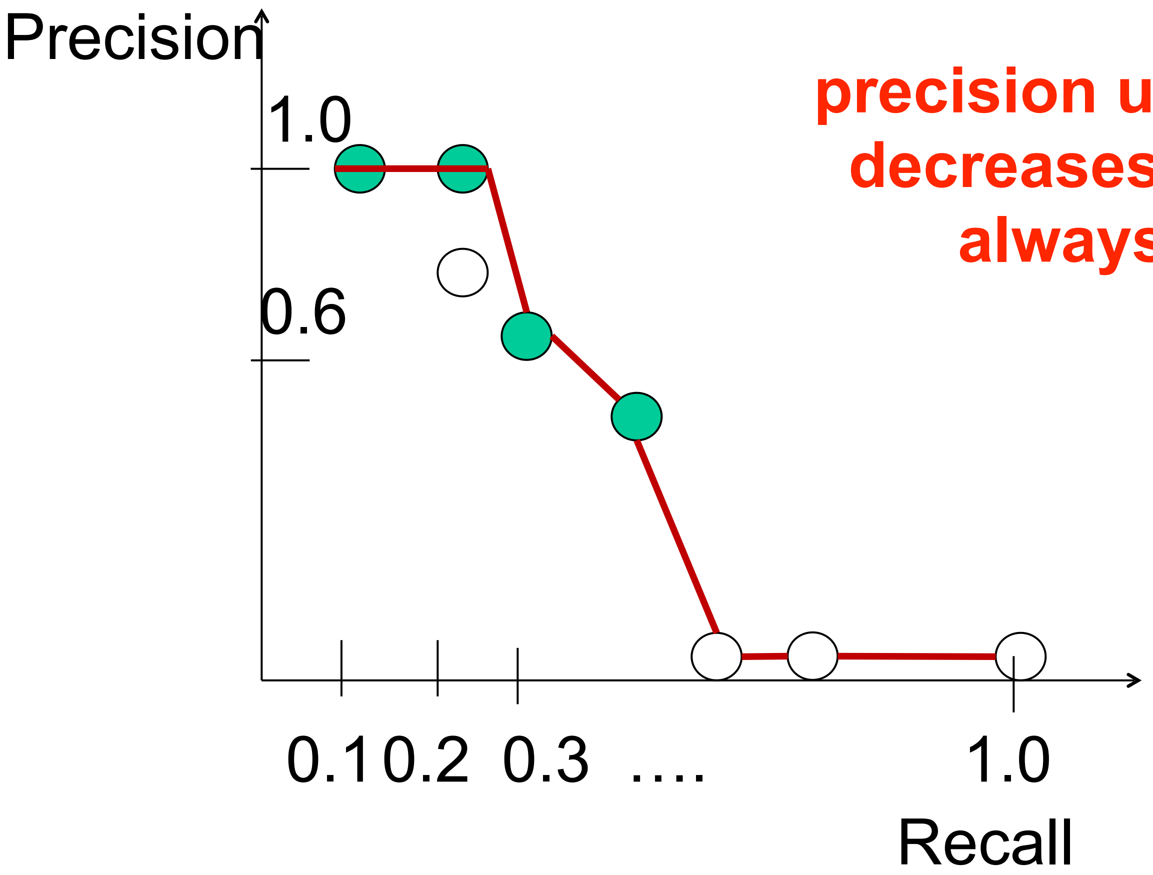
- Fraction of relevant documents that are retrieved

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

# Precision-recall curve

| | Precision | Recall |
|---|---|---|
| + | 1/1 | 1/4 |
| + | 2/2 | 2/4 |
| – | | |
| – | | |
| + | 3/5 | 3/4 |
| – | | |
| – | | |
| + | 4/8 | 4/4 |
| – | | |
| – | | |

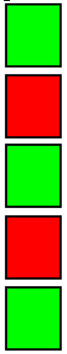
$(1/1 + 2/2 + 3/5 + 4/8) / 4$



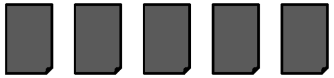

**precision usually decreases (not always)**

***Slides from UIUC CS598***

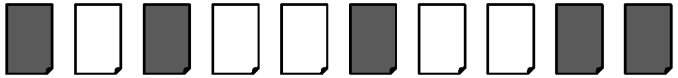
# Mean average precision

- Consider rank position of each ***relevant*** doc
  - $K_1, K_2, \ldots K_R$
- Compute Precision@K for $K = K_1, K_2, \ldots K_R$
- Average precision = average of P@K
- Ex:  has AvgPrec of $\frac{1}{3} \cdot \left( \frac{1}{1} + \frac{2}{3} + \frac{3}{5} \right) \approx 0.76$
- MAP is Average Precision across multiple queries/ rankings

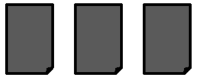
# MAP


= relevant documents for query 1

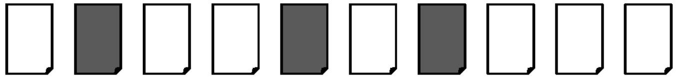
Ranking #1



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Recall | 0.2 | 0.2 | 0.4 | 0.4 | 0.4 | 0.6 | 0.6 | 0.6 | 0.8 | 1.0 |
| Precision | 1.0 | 0.5 | 0.67 | 0.5 | 0.4 | 0.5 | 0.43 | 0.38 | 0.44 | 0.5 |


= relevant documents for query 2

Ranking #2



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Recall | 0.0 | 0.33 | 0.33 | 0.33 | 0.67 | 0.67 | 1.0 | 1.0 | 1.0 | 1.0 |
| Precision | 0.0 | 0.5 | 0.33 | 0.25 | 0.4 | 0.33 | 0.43 | 0.38 | 0.33 | 0.3 |

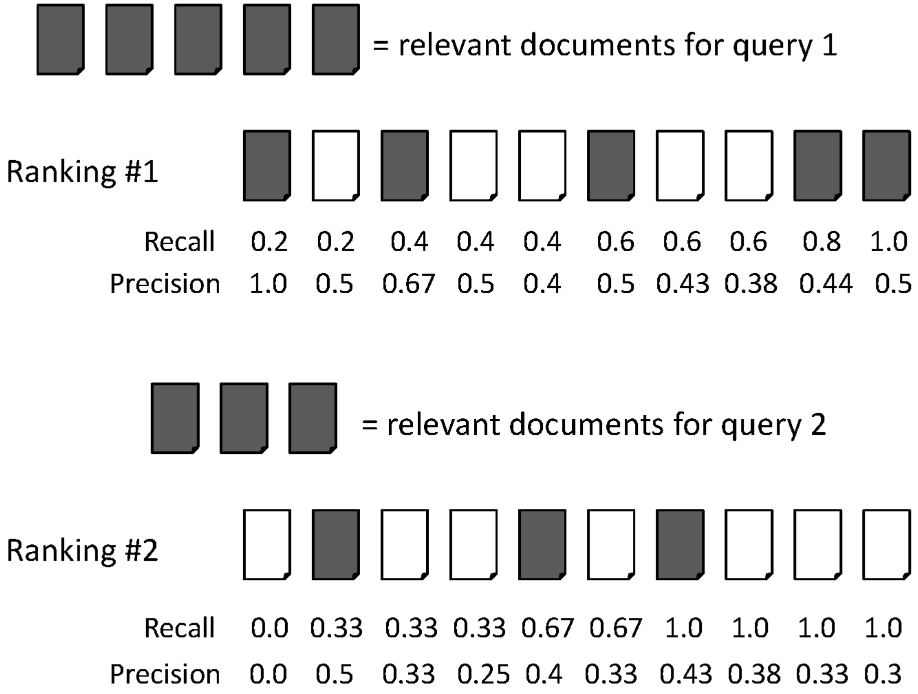
$$\textit{average precision query 1} = (1.0 + 0.67 + 0.5 + 0.44 + 0.5)/5 = 0.62$$
$$\textit{average precision query 2} = (0.5 + 0.4 + 0.43)/3 = 0.44$$

$$\textit{mean average precision} = (0.62 + 0.44)/2 = 0.53$$

***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***

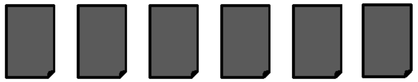
# Beyond binary relevance

- Discounted cumulative gain (DCG)

- Popular measure for evaluating web search and related tasks

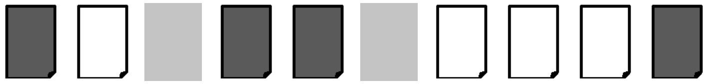
- Information gain-based evaluation (economics)
  - For each relevant document, the user has gained some information
  - The higher the relevance, the higher gain
  - The gain is discounted when the relevant document appears in a lower position

# Discounted cumulative gain (DCG)



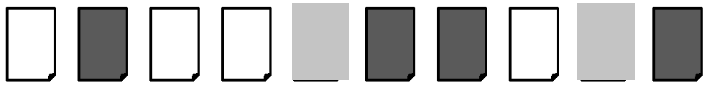
= the relevant documents

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?

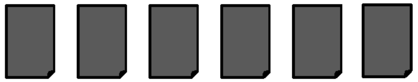
- If we do not normalize DCG, the performance will be biased towards systems that perform well on queries with larger DCG scales

= 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 |

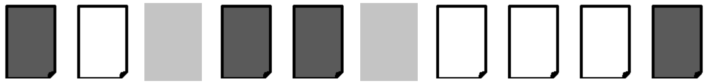
**bias towards B**

# Normalized Discounted cumulative gain (nDCG)



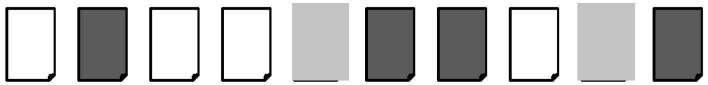
= the relevant documents

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

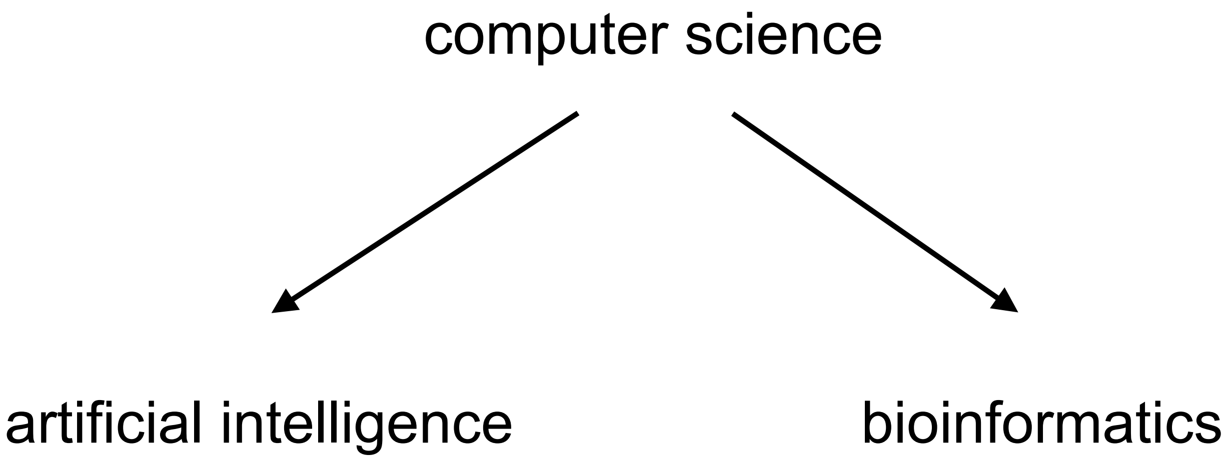
<desc> Description:
How are pets or animals used in therapy for humans and what are the benefits?

<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

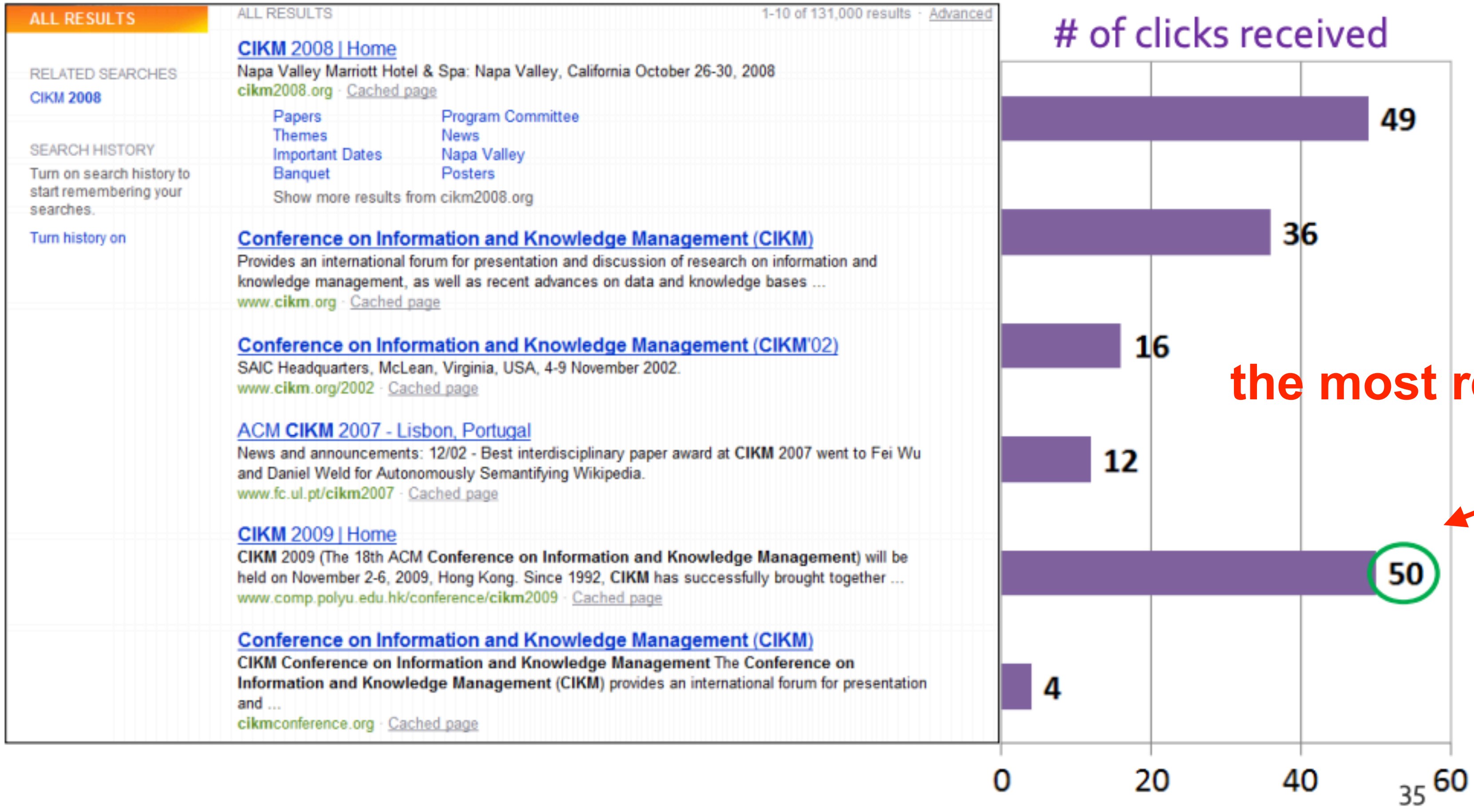
- TREC style relevance judgment
  - Explicit relevance judgment
  - Difficult to achieve large scalability
  - Relevance is **fixed**

- 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**

35

# 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" |

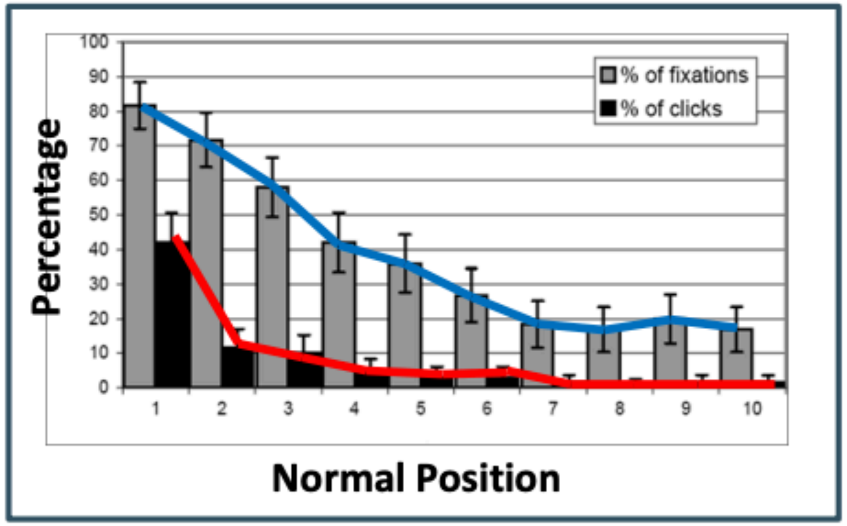
# Online evaluation methodology

- 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

# Position bias [Craswell 08]

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




# Decoy effects

vs

$400, 20G

~~click probability = 0.3~~

**click probability = 0.5**

$500, 30G

~~click probability = 0.4~~

**click probability = 0.5**

$550, 20G

# 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

| |
|---|
| Kernel machines |
| Kernel machines |
| SVMs |
| SVM-light |
| Intro to SVMs |
| Lucent SVM demo |
| Archives of SVM |
| Royal Holl. SVM |
| SVM-light |

remove dup

| |
|---|
| Kernel machines |
| Kernel machines |
| SVMs |
| SVM-light |
| Intro to SVMs |
| Lucent SVM demo |
| Archives of SVM |
| Royal Holl. SVM |
| SVM-light |

**A clicks = 3, B clicks = 1**

# Online evaluation methodology

- Bing has an existing ranking algorithm A
  - Testing algorithm B is better than A
    - Strategy 1: Running A of 1 month, running B for the next month
    - Strategy 2: Running A 50% of the time, B 50% of the time

- Disadvantage with Strategy 1 and 2:
  - If B fails, it will hurts user experience from the B group

- Running B 5% of the time, running A 95% of the time

# 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

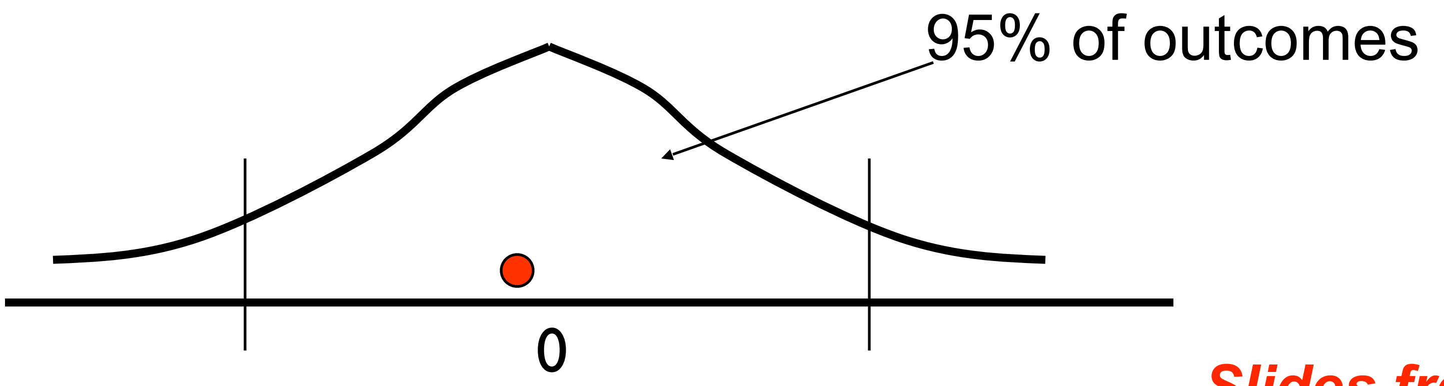
| Query | System A | System B |
|---|---|---|
| 1 | 0.02 | 0.76 |
| 2 | 0.39 | 0.07 |
| 3 | 0.16 | 0.37 |
| 4 | 0.58 | 0.21 |
| 5 | 0.04 | 0.02 |
| 6 | 0.09 | 0.91 |
| 7 | 0.12 | 0.46 |
| Average | 0.20 | 0.40 |



*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 |

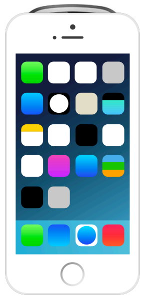




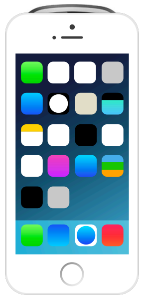

*Slides from UIUC CS598*

# Retrieval feedback in session search

query = “best phone”



$400, 20G,
Nokia



$500, 30G,
Nokia

$600, 40G,
iphone

observed click

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

**session 2**

# Rocchio feedback

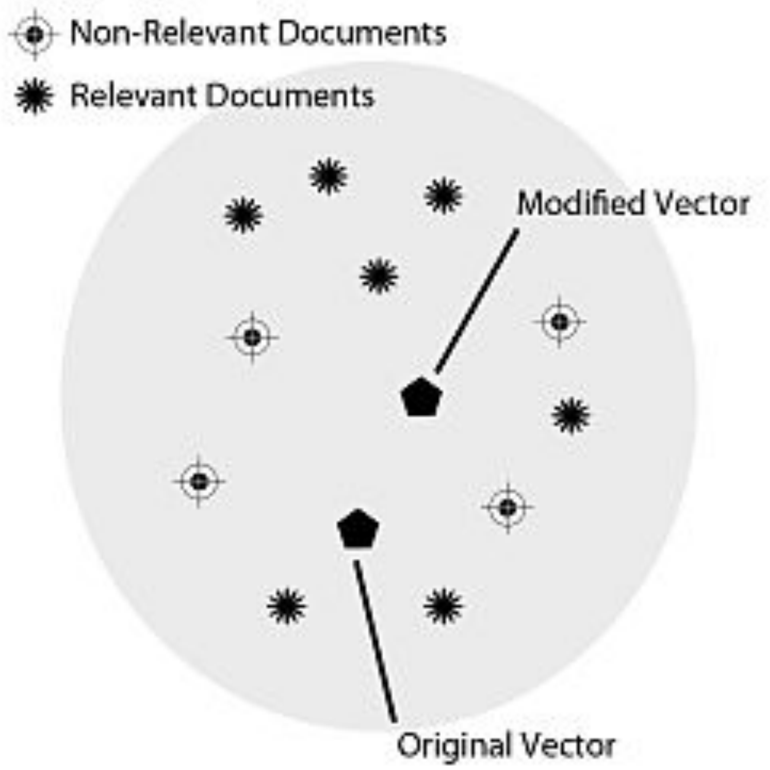
- Feedback for vector-space model

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

rel docs

non-rel docs

$\alpha \gg \beta$




- 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

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

**(Robertson & Sparck Jones 76)**

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

Probability for a word to appear in a relevant doc

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

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

# (Pseudo)relevance feedback language model

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

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

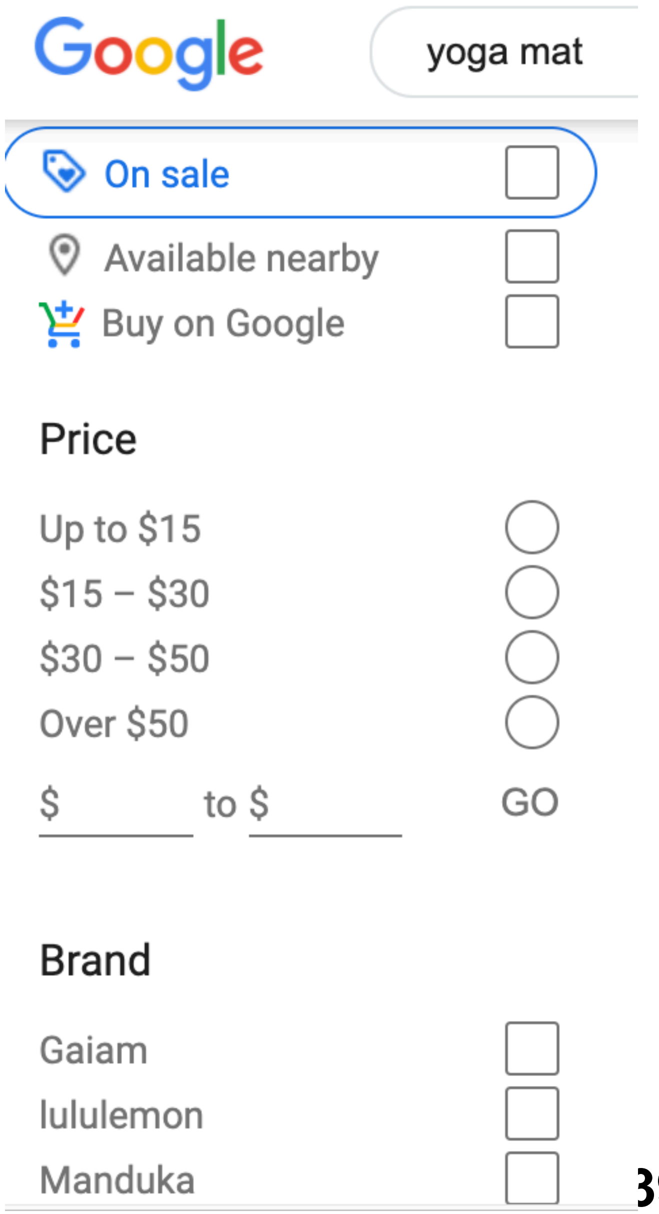
***sparsity***

$$\begin{tikzcd}
d \arrow[r] & \theta_d \arrow[dr] & \\
q \arrow[r] & \theta_q \arrow[r] & -D(\theta_q|\theta_d) \arrow[d] \\
 & \theta_q^F \arrow[u, "(1-\alpha)\theta_q + \alpha\theta_q^F"] & d_1, d_2, \cdots, d_n \arrow[l]
\end{tikzcd}$$

# Performance of relevance feedback models

| S.w. | Metric | MLE | RM3 | RM4 | DMM | SMM | RMM |
|---|---|---|---|---|---|---|---|
| Trained on AP1 and Tested on AP2 | | | | | | | |
| w/ | AvgPr | 0.220 | 0.295 | 0.301 | 0.290 | **0.304** | 0.299 |
| | Pr@10 | 0.386 | 0.408 | 0.418 | **0.422** | 0.400 | 0.398 |
| | Recall | 3074 | 3810 | 3892 | 3681 | **3933** | 3859 |
| w/o | AvgPr | 0.231 | 0.312 | 0.321 | 0.289 | **0.324** | 0.323 |
| | Pr@10 | 0.398 | 0.436 | **0.448** | 0.424 | 0.432 | 0.446 |
| | Recall | 3154 | 3913 | 3908 | 3674 | 3921 | **3927** |
| Trained on TREC6 and Tested on TREC78 | | | | | | | |
| w/ | AvgPr | 0.217 | 0.249 | 0.242 | 0.235 | **0.251** | 0.243 |
| | Pr@10 | 0.437 | 0.438 | 0.426 | 0.443 | 0.443 | **0.451** |
| | Recall | 5114 | 5805 | 5739 | 5476 | **5821** | 5625 |
| w/o | AvgPr | 0.217 | 0.251 | 0.243 | 0.235 | **0.252** | 0.249 |
| | Pr@10 | 0.434 | **0.454** | 0.446 | 0.433 | 0.441 | 0.443 |
| | Recall | 5107 | 5799 | 5776 | 5500 | **5896** | 5833 |
| Well-Tuned on WT2G | | | | | | | |
| w/ | AvgPr | 0.293 | **0.338** | 0.319 | 0.327 | 0.330 | 0.309 |
| | Pr@10 | 0.450 | **0.500** | 0.470 | 0.494 | 0.496 | 0.458 |
| | Recall | 1830 | 1822 | 1806 | 1843 | **1856** | 1811 |
| w/o | AvgPr | 0.306 | **0.344** | 0.328 | 0.326 | 0.331 | 0.319 |
| | Pr@10 | 0.456 | **0.490** | **0.490** | 0.476 | 0.476 | 0.482 |
| | Recall | 1870 | 1862 | 1879 | 1873 | **1889** | 1863 |

# Query expansion

# Query expansion

- Query expansion/reformulation techniques
  - Using manually created synonyms
  - Using automatic 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 |