

# Evaluation of search results

Chapter 8 - IIR



SAPIENZA  
UNIVERSITÀ DI ROMA

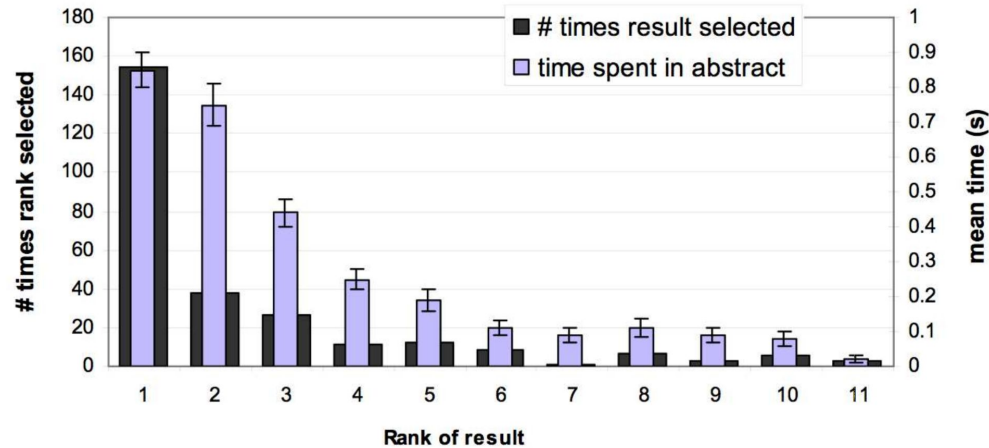
Fabrizio Silvestri

# Quick Recap



# Users' Behavior

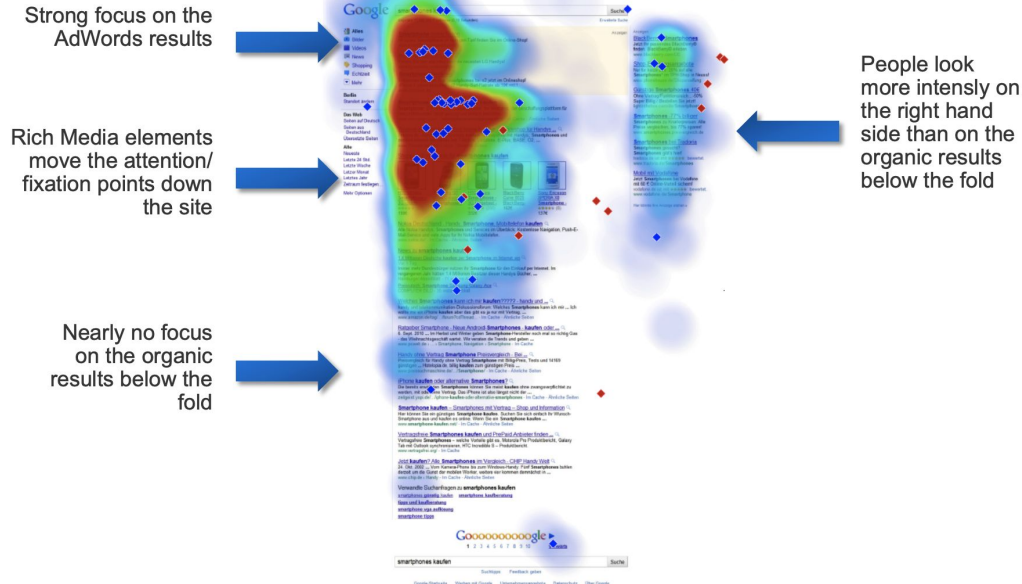
## Looking vs. Clicking



- Users view results one and two more often / thoroughly
- Users click most frequently on result one

# Attention of Users

## Desktop search engine result page



# Attention of Users

## Mobile search engine results page

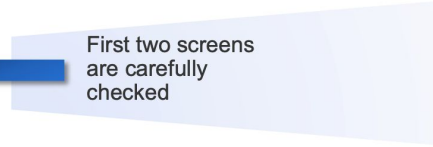
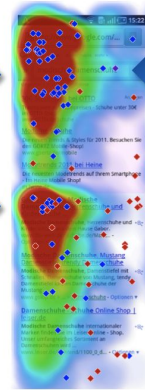
First focus on the search bar and the first AdWords ad

Second focus on the first organic result

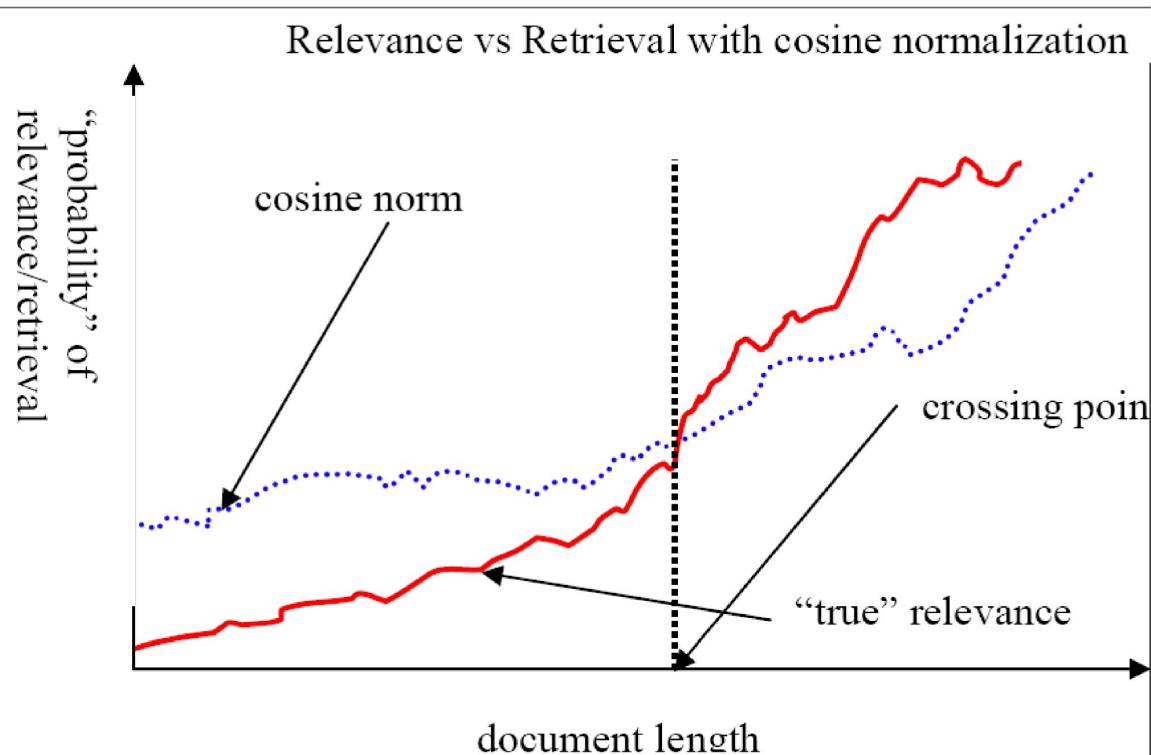
First two screens are carefully checked

A traditional above and below the fold is missing on the mobile page

Ads at the end of the page are noticed by the respondents



# Predicted and true probability of relevance



## Effect on Effectiveness: Amit Singhal's experiments

Cosine	Pivoted Cosine Normalization				
	Slope				
	0.60	0.65	0.70	<b>0.75</b>	0.80
6,526	6,342	6,458	6,574	<b>6,629</b>	6,671
0.2840	0.3024	0.3097	0.3144	<b>0.3171</b>	0.3162
Improvement	+ 6.5%	+ 9.0%	+10.7%	<b>+11.7%</b>	+11.3%

- (relevant documents retrieved and (change in) average precision)



# Evaluation of Search Results





# Outline

- Introduction to evaluation: Measures of an IR system
- Evaluation of unranked and ranked retrieval
- Evaluation benchmarks
- Result summaries



# What to measure

- How fast does it index?
  - e.g., number of bytes per hour
- How fast does it search?
  - e.g., latency as a function of queries per second
- What is the cost per query?
  - in dollars



# What to measure

- All of the preceding criteria are measurable: we can quantify speed / size / money
- However, the key measure for a search engine is **user happiness**.
- What is user happiness?
- Factors include:
  - Speed of response
  - Size of index
  - Uncluttered UI
  - Most important: **relevance**
    - (actually, maybe even more important: it's free)
- Note that none of these is sufficient: blindingly fast, but useless answers won't make a user happy.
- **How can we quantify user happiness?**



# Who is the user?

- Who is the user we are trying to make happy?
- Web search engine: searcher. Success: Searcher finds what she was looking for. Measure: rate of return to this search engine.
- Web search engine: advertiser. Success: Searcher clicks on ad. Measure: clickthrough rate.
- Ecommerce: buyer. Success: Buyer buys something. Measures: time to purchase, fraction of “conversions” of searchers to buyers.
- Ecommerce: seller. Success: Seller sells something. Measure: profit per item sold.
- Enterprise: CEO. Success: Employees are more productive (because of effective search). Measure: profit of the company.



# Most common definition of user happiness: Relevance

- User happiness is equated with the relevance of search results to the query.
- But how do you measure relevance?
- Standard methodology in information retrieval consists of three elements.
  - A benchmark document collection
  - A benchmark suite of queries
  - An assessment of the relevance of each query-document pair



# Relevance: query vs. information need

- Relevance to what?
- First take: relevance to the query
- “Relevance to the query” is very problematic.
- Information need i: “I am looking for information on whether drinking red wine is more effective at reducing your risk of heart attacks than white wine.”
- This is an information need, not a query.
- Query q: [red wine white wine heart attack]
- Consider document d': At the heart of his speech was an attack on the wine industry lobby for downplaying the role of red and white wine in drunk driving.
- d' is an excellent match for query q . . .
- d' is not relevant to the information need i.



# Relevance: query vs. information need

- User happiness can only be measured by relevance to an information need, not by relevance to queries.
- Our terminology is sloppy in these slides and in IIR: we talk about query-document relevance judgments even though we mean information-need-document relevance judgments.



# Relevance: query vs. information need

- User happiness can only be measured by relevance to an information need, not by relevance to queries.
- Our terminology is sloppy in these slides and in IIR: we talk about query-document relevance judgments even though we mean information-need-document relevance judgments.





# Unranked Evaluation



# Precision and Recall

- Precision (P) is the fraction of retrieved documents that are relevant

$$\text{Precision} = \frac{\#(\text{relevant items retrieved})}{\#(\text{retrieved items})} = P(\text{relevant}|\text{retrieved})$$

- Recall (R) is the fraction of relevant documents that are retrieved

$$\text{Recall} = \frac{\#(\text{relevant items retrieved})}{\#(\text{relevant items})} = P(\text{retrieved}|\text{relevant})$$



# Precision and Recall

	Relevant	Nonrelevant
Retrieved	true positives (TP)	false positives (FP)
Not retrieved	false negatives (FN)	true negatives (TN)

$$P = TP / (TP + FP)$$

$$R = TP / (TP + FN)$$



# Precision and Recall Tradeoff

- You can increase recall by returning more docs.
- Recall is a non-decreasing function of the number of docs retrieved.
  - A system that returns all docs has 100% recall!
- The converse is also true (usually): It's easy to get high precision for very low recall.
- Suppose the document with the largest score is relevant. How can we maximize precision?



# F-Measure

- F allows us to trade off precision against recall.

$$F = \frac{1}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R} \quad \text{where} \quad \beta^2 = \frac{1 - \alpha}{\alpha}$$

- $\alpha \in [0, 1]$  and thus  $\beta^2 \in [0, \infty]$
- Most frequently used: balanced F with  $\beta = 1$  or  $\alpha = 0.5$
- This is the harmonic mean of P and R:  $\frac{1}{F} = \frac{1}{2} \left( \frac{1}{P} + \frac{1}{R} \right)$
- What value range of  $\beta$  weights recall higher than precision?



## Example of P, R, F1-Measure

	relevant	not relevant	
retrieved	20	40	60
not retrieved	60	1,000,000	1,000,060
	80	1,000,040	1,000,120

- $P = 20 / (20 + 40) = \frac{1}{3}$
- $R = 20 / (20 + 60) = \frac{1}{4}$
- $F1 = 2 / ((1 / \frac{1}{3}) + (1 / \frac{1}{4}))$



# Accuracy

- Why do we use complex measures like precision, recall, and F?
- Why not something simple like accuracy?
- Accuracy is the fraction of decisions (relevant/nonrelevant) that are correct.
- In terms of the contingency table above,  
$$\text{accuracy} = (TP + TN)/(TP + FP + FN + TN).$$



# Example

- Compute precision, recall and F1 for this result set:

	relevant	not relevant
retrieved	18	2
not retrieved	82	1,000,000,000

- The snoogle search engine below always returns 0 results (“0 matching results found”), regardless of the query. Why does snoogle demonstrate that accuracy is not a useful measure in IR?





# Why accuracy is a useless measure in IR

- Simple trick to maximize accuracy in IR: always say no and return nothing
- You then get 99.99% accuracy on most queries.
- Searchers on the web (and in IR in general) want to find something and have a certain tolerance for junk.
- It's better to return some bad hits as long as you return something.
  - → We use precision, recall, and F for evaluation, not accuracy.



## F: Why harmonic mean?

- Why don't we use a different mean of P and R as a measure?
  - e.g., the arithmetic mean
- The simple (arithmetic) mean is close to 50% for snooglevsearch engine – which is too high.
- Desideratum: Punish really bad performance on either precision or recall.
- Taking the minimum achieves this.
- But minimum is not smooth and hard to weight.
- F (harmonic mean) is a kind of smooth minimum.



# Difficulties in using precision, recall and F

- We need relevance judgments for information-need-document pairs – but they are expensive to produce.
- For alternatives to using precision/recall and having to produce relevance judgments – see end of this lecture.



# Ranked Evaluation

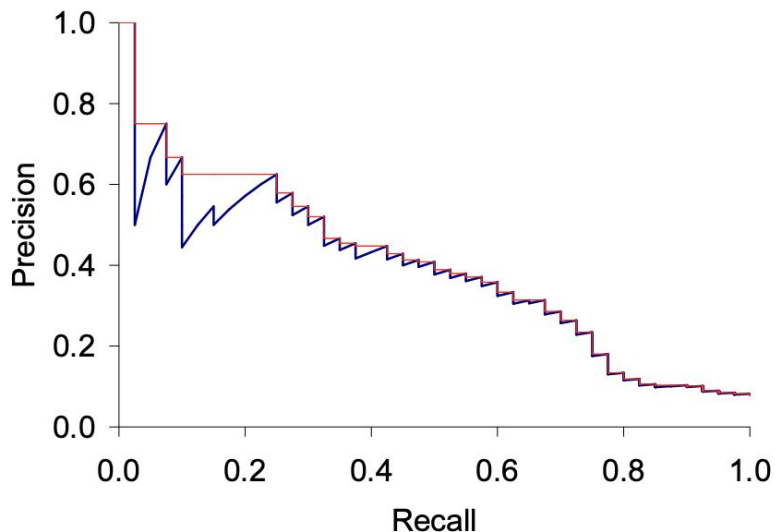


# Precision Recall (PR) Curve

- Precision/recall/F are measures for **unranked sets**.
- We can easily turn set measures into measures of **ranked lists**.
- Just compute the set measure for each “prefix”: the top 1, top 2, top 3, top 4 etc results
- Doing this for precision and recall gives you a **precision-recall curve**.



# Precision Recall (PR) Curve



- Each point corresponds to a result for the top  $k$  ranked hits ( $k = 1, 2, 3, \dots$ ).
- Interpolation (in red): Take maximum of all future points
- Rationale for interpolation: The user is willing to look at more stuff if both precision and recall get better.



# 11-point interpolated average precision

Recall	Interpolated Precision
--------	---------------------------

0.0	1.00
-----	------

0.1	0.67
-----	------

0.2	0.63
-----	------

0.3	0.55
-----	------

0.4	0.45
-----	------

0.5	0.41
-----	------

0.6	0.36
-----	------

0.7	0.29
-----	------

0.8	0.13
-----	------

0.9	0.10
-----	------

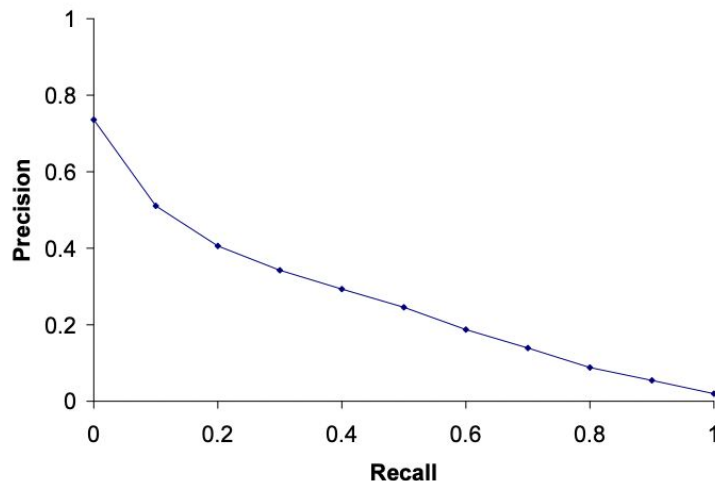
1.0	0.08
-----	------

11-point average:  $\approx$   
0.425

How can precision  
at 0.0 be  $> 0$ ?



# 11-point interpolated average precision

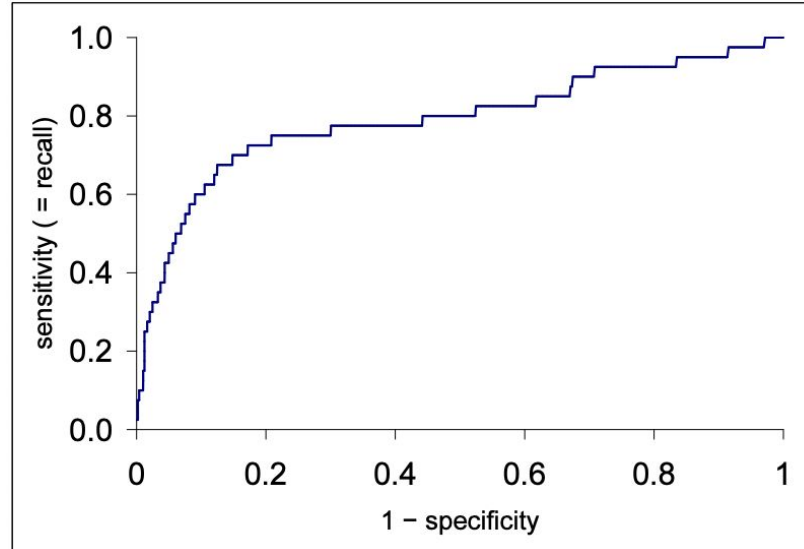


- Compute interpolated precision at recall levels 0.0, 0.1, 0.2, . . .
- Do this for each of the queries in the evaluation benchmark
- Average over queries
- This measure measures performance at all recall levels.
- The curve is typical of performance levels at TREC.
- Note that performance is not very good!





# ROC Curve



- Similar to precision-recall graph
- But we are only interested in the small area in the lower left corner.
- Precision-recall graph “blows up” this area.



# Variance of measures like precision/recall

- For a test collection, it is usual that a system does badly on some information needs (e.g.,  $P = 0.2$  at  $R = 0.1$ ) and really well on others (e.g.,  $P = 0.95$  at  $R = 0.1$ ).
- Indeed, it is usually the case that the **variance of the same system across queries** is much **greater than the variance of different systems on the same query**.
- That is, there are easy information needs and hard ones.




# Variance of measures like precision/recall

- For a test collection, it is usual that a system does badly on some information needs (e.g.,  $P = 0.2$  at  $R = 0.1$ ) and really well on others (e.g.,  $P = 0.95$  at  $R = 0.1$ ).
- Indeed, it is usually the case that the **variance of the same system across queries** is much **greater than the variance of different systems on the same query**.
- That is, there are easy information needs and hard ones.



# Precision@K

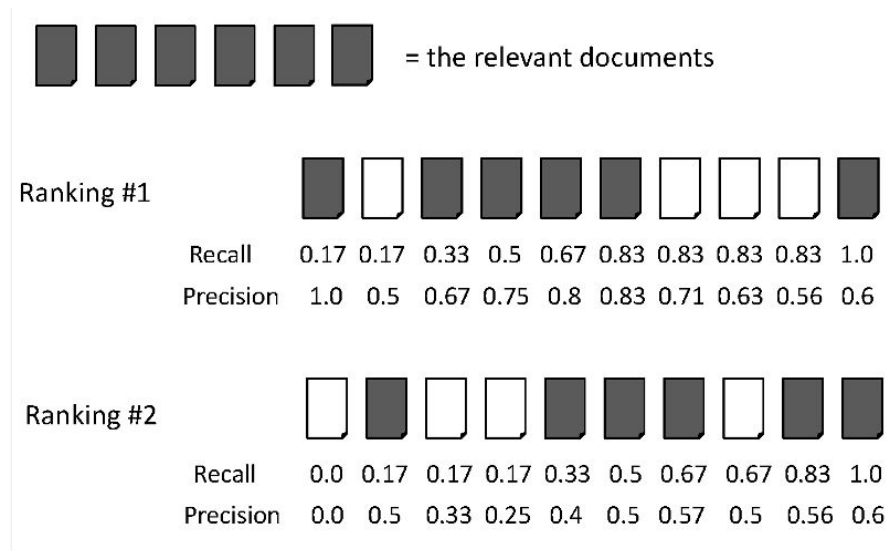
- Set a rank threshold K
- Compute % relevant in top K
- Ignores documents ranked lower than K

• Ex: 

- $\text{Prec}@3$  of  $2/3$
- $\text{Prec}@4$  of  $2/4$
- $\text{Prec}@5$  of  $3/5$



# Average Precision




$$\text{Ranking \#1: } (1.0 + 0.67 + 0.75 + 0.8 + 0.83 + 0.6)/6 = 0.78$$

$$\text{Ranking \#2: } (0.5 + 0.4 + 0.5 + 0.57 + 0.56 + 0.6)/6 = 0.52$$

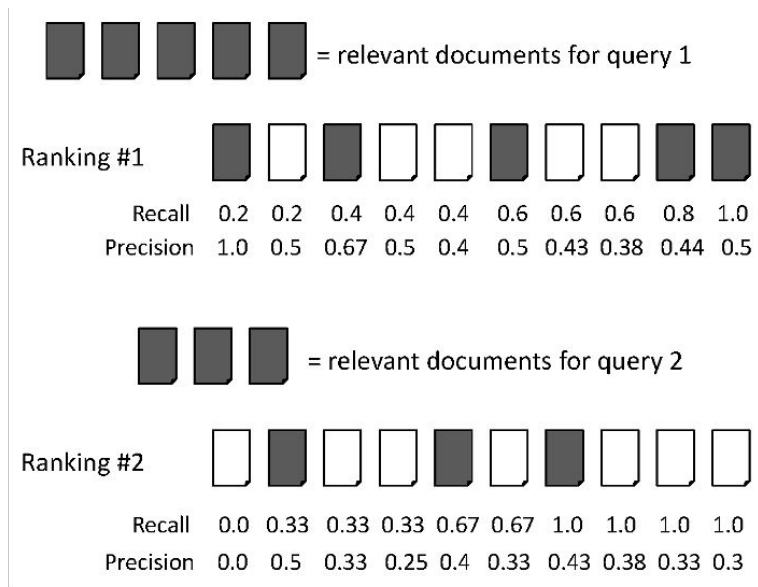


# Mean Average Precision (MAP)

- Consider rank position of each relevant doc
  - K1, K2, ... KR
- Compute Precision@K for each K1, K2, ... KR
- 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



# Mean Average Precision (MAP)



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

$$\text{average precision query 2} = (0.5 + 0.4 + 0.43)/3 = 0.44$$

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



# Mean Average Precision (MAP)

- If a relevant document never gets retrieved, we assume the precision corresponding to that relevant doc to be zero
- MAP is macro-averaging: each query counts equally
- Now perhaps most commonly used measure in research papers
- Good for web search?
- MAP assumes user is interested in finding many relevant documents for each query
- MAP requires many relevance judgments in text collection





# The case of a single relevant doc

- Scenarios:
  - known-item search
  - navigational queries
  - looking for a fact
- Search Length = Rank of the answer
  - measures a user's effort



# Mean Reciprocal Rank (MRR)

- Consider rank position,  $K$ , of first relevant doc
- Reciprocal Rank score =  $1 / K$
- MRR is the mean RR across multiple queries



# Critique of pure relevance

- Relevance vs Marginal Relevance
  - A document can be redundant even if it is highly relevant
    - Duplicates
    - The same information from different sources
  - Marginal relevance is a better measure of utility for the user
    - But harder to create evaluation set
- Using facts/entities as evaluation unit can more directly measure true recall
- Also related is seeking diversity in first page results

- See also recent paper on evaluation by N. Fuhr:  
[https://www.is.inf.uni-due.de/bib/pdf/ir/Fuhr\\_17b.pdf](https://www.is.inf.uni-due.de/bib/pdf/ir/Fuhr_17b.pdf)





artificial intelligence news



[All](#) [News](#) [Videos](#) [Images](#) [Maps](#) [More](#) [Settings](#) [Tools](#)

About 601,000,000 results (0.64 seconds)

[https://www.sciencedaily.com/news/computers\\_math/](https://www.sciencedaily.com/news/computers_math/)

### Artificial Intelligence News -- ScienceDaily

Artificial Intelligence News. Everything on AI including futuristic robots with artificial intelligence, computer models of human intelligence and more.

[Artificial emotional intelligence](#) · [Photonics for artificial...](#) · ["Audeo" teaches artificial...](#)

Fair

<https://artificialintelligence-news.com/>

### AI News - Artificial Intelligence News

Artificial Intelligence News provides the latest AI news and trends. Explore industry research and reports from the frontline of AI technology news.

[News](#) · [Featured: AI News' list of...](#) · [British intelligence agency...](#) · [Engagement](#)

Good

<https://news.mit.edu/topic/artificial-intelligence2/>

### Artificial intelligence | MIT News | Massachusetts Institute of ...

Artificial Intelligence. Download RSS feed: News Articles / In the Media. Displaying 1 - 15 of 706 news articles related to this topic. Show: News Articles.

Fair

<https://www.businessinsider.com/artificial-intelligence/>

### Artificial Intelligence News: Latest Advancements in AI ...

Artificial Intelligence (AI), or machine intelligence, is the field developing computers and robots capable of parsing data contextually to provide requested ...

What is Artificial Intelligence (AI)?	▼
How does Artificial Intelligence work?	▼
What are the different types of Artificial Intelligence?	▼
▼ Show more	

#### People also ask

What is the current status of artificial intelligence in the world?	▼
What is the most advanced AI right now?	▼
What is AI being used for today?	▼

[Feedback](#)

<https://www.bbc.co.uk/news/topics/artificial-intellige...>

### Artificial intelligence - BBC News

All the latest news about Artificial Intelligence from the BBC. ... An outright ban on some AI systems, such as "social scoring" by governments, is proposed for the ...



SAPIENZA  
UNIVERSITÀ DI ROMA

# Discounted Cumulative Gain

- Popular measure for evaluating web search and related tasks
- Two assumptions:
  - Highly relevant documents are more useful than marginally relevant document
  - the lower the ranked position of a relevant document, the less useful it is for the user, since it is less likely to be examined



# Discounted Cumulative Gain

- Uses graded relevance as a measure of usefulness, or gain, from examining a document
- Gain is accumulated starting at the top of the ranking and may be reduced, or discounted, at lower ranks
- Typical discount is  $1/\log(\text{rank})$ 
  - With base 2, the discount at rank 4 is  $1/2$ , and at rank 8 it is  $1/3$



# Summarize a Ranking: DCG

- What if relevance judgments are in a scale of  $[0, r]$ ?  $r > 2$
- Cumulative Gain (CG) at rank  $n$ 
  - Let the ratings of the  $n$  documents be  $r_1, r_2, \dots, r_n$  (in ranked order)
  - $CG = r_1 + r_2 + \dots + r_n$
- Discounted Cumulative Gain (DCG) at rank  $n$ 
  - $DCG = r_1 + r_2 / \log_2 2 + r_3 / \log_2 3 + \dots + r_n / \log_2 n$ 
    - We may use any base for the logarithm, e.g., base= $b$



# Discounted Cumulative Gain

- DCG is the total gain accumulated at a particular rank  $p$ :

$$DCG_p = rel_1 + \sum_{i=2}^p \frac{rel_i}{\log_2 i}$$

- Alternative formulation:

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

- used by some web search companies
- emphasis on retrieving highly relevant documents





# DCG Example

- 10 ranked documents judged on 0-3 relevance scale:

3, 2, 3, 0, 0, 1, 2, 2, 3, 0

- discounted gain:

3,  $2/1$ ,  $3/1.59$ , 0, 0,  $1/2.59$ ,  $2/2.81$ ,  $2/3$ ,  $3/3.17$ , 0

= 3, 2, 1.89, 0, 0, 0.39, 0.71, 0.67, 0.95, 0

- DCG:

3, 5, 6.89, 6.89, 6.89, 7.28, 7.99, 8.66, 9.61, 9.61



# Summarize a Ranking: NDCG

- Normalized Cumulative Gain (NDCG) at rank  $n$ 
  - Normalize DCG at rank  $n$  by the DCG value at rank  $n$  of the ideal ranking
  - The ideal ranking would first return the documents with the highest relevance level, then the next highest relevance level, etc
  - Compute the precision (at rank) where each (new) relevant document is retrieved  $\Rightarrow p(1), \dots, p(k)$ , if we have  $k$  rel. Docs
- NDCG is now quite popular in evaluating Web search



# NDCG Example

4 documents:  $d_1, d_2, d_3, d_4$

i	Ground Truth		Ranking Function <sub>1</sub>		Ranking Function <sub>2</sub>	
	Document Order	$r_i$	Document Order	$r_i$	Document Order	$r_i$
1	d4	2	d3	2	d3	2
2	d3	2	d4	2	d2	1
3	d2	1	d2	1	d4	2
4	d1	0	d1	0	d1	0
	NDCG <sub>GT</sub> =1.00		NDCG <sub>RF1</sub> =1.00		NDCG <sub>RF2</sub> =0.9203	

$$DCG_{GT} = 2 + \left( \frac{2}{\log_2 2} + \frac{1}{\log_2 3} + \frac{0}{\log_2 4} \right) = 4.6309$$

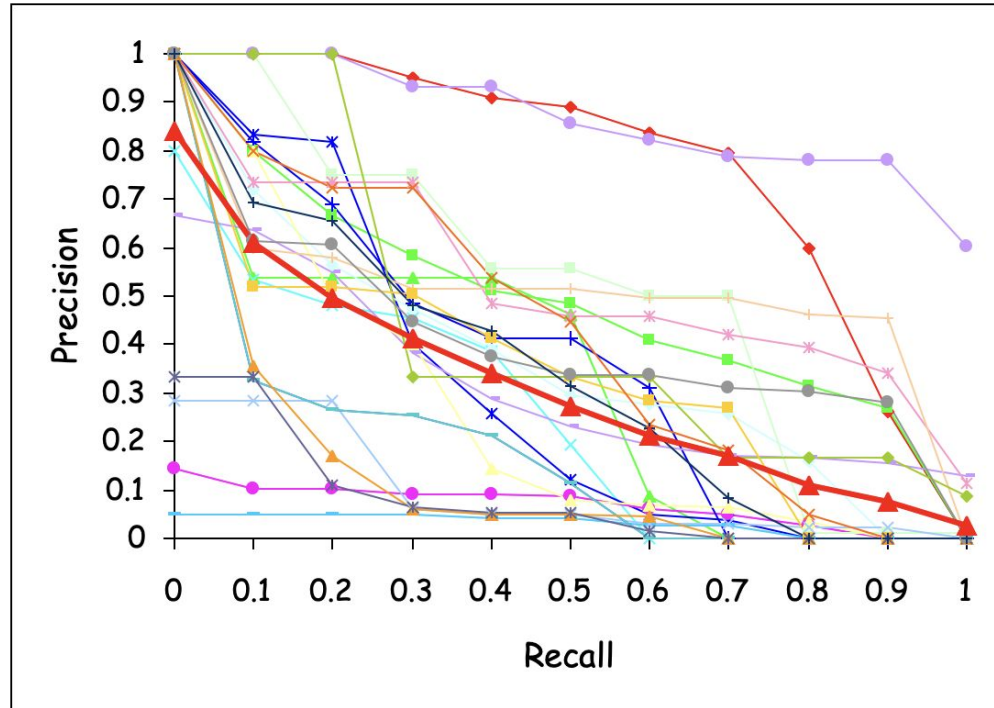
$$DCG_{RF1} = 2 + \left( \frac{2}{\log_2 2} + \frac{1}{\log_2 3} + \frac{0}{\log_2 4} \right) = 4.6309$$

$$DCG_{RF2} = 2 + \left( \frac{1}{\log_2 2} + \frac{2}{\log_2 3} + \frac{0}{\log_2 4} \right) = 4.2619$$

$$MaxDCG = DCG_{GT} = 4.6309$$



# What Query Averaging Hides



Slide from Doug Oard's presentation, originally from Ellen Voorhees' presentation



# Benchmarks



# What we need for a benchmarks

- A collection of documents
  - Documents should be representative of the documents we expect to see in reality.
- A collection of information needs (often incorrectly called queries)
  - Information needs should be representative of the information needs we expect to see in reality.
- Human relevance assessments
  - We need to hire/pay “judges” or assessors to do this.
  - Expensive, time-consuming
  - Judges should be representative of the users we expect to see in reality.



# First standard relevance benchmark: Cranfield

- Pioneering: first testbed allowing precise quantitative measures of information retrieval effectiveness
- Late 1950s, UK
- 1398 abstracts of aerodynamics journal articles, a set of 225 queries, exhaustive relevance judgments of all query-document-pairs
- Too small, too untypical for serious IR evaluation today



# Second-generation relevance benchmark: TREC

- TREC = Text Retrieval Conference (TREC)
- Organized by the U.S. National Institute of Standards and Technology (NIST)
- TREC is actually a set of several different relevance benchmarks.
- Best known: TREC Ad Hoc, used for first 8 TREC evaluations between 1992 and 1999
- 1.89 million documents, mainly newswire articles, 450 information needs
- No exhaustive relevance judgments – too expensive
- Rather, NIST assessors' relevance judgments are available only for the documents that were among the top k returned for some system which was entered in the TREC evaluation for which the information need was developed.





# Example of more recent benchmark: ClueWeb09

- 1 billion web pages
- 25 terabytes (compressed: 5 terabyte)
- Collected January/February 2009
- 10 languages
- Unique URLs: 4,780,950,903 (325 GB uncompressed, 105 GB compressed)
- Total Outlinks: 7,944,351,835 (71 GB uncompressed, 24 GB compressed)



# Validity of relevance assessments

- Relevance assessments are only usable if they are consistent.
- If they are not consistent, then there is no “truth” and experiments are not repeatable.
- How can we measure this consistency or agreement among judges?
- → Kappa measure



# Kappa measure

- Kappa is measure of how much judges agree or disagree.
- Designed for categorical judgments
- Corrects for chance agreement
- $P(A)$  = proportion of time judges agree
- $P(E)$  = what agreement would we get by chance

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)}$$

- $\kappa$  =? for (i) chance agreement (ii) total agreement



# Kappa measure

- Values of  $\kappa$  in the interval  $[2/3, 1.0]$  are seen as acceptable.
- With smaller values: need to redesign relevance assessment methodology used etc.



# Calculating the Kappa statistics

		Judge 2 Relevance			
		Yes	No	Total	
Judge 1 Relevance	Yes	300	20	320	Observed proportion of
	No	10	70	80	
	Total	310	90	400	

the times the judges agreed

$$P(A) = (300 + 70)/400 = 370/400 = 0.925$$

Pooled marginals

$$P(\text{nonrelevant}) = (80 + 90)/(400 + 400) = 170/800 = 0.2125$$

$$P(\text{relevant}) = (320 + 310)/(400 + 400) = 630/800 = 0.7875$$

Probability that the two judges agreed by chance  $P(E) =$

$$P(\text{nonrelevant})^2 + P(\text{relevant})^2 = 0.2125^2 + 0.7875^2 = 0.665$$

Kappa statistic  $\kappa = (P(A) - P(E))/(1 - P(E)) =$

$$(0.925 - 0.665)/(1 - 0.665) = 0.776 \text{ (still in acceptable range)}$$



## Interjudge agreement at TREC

information need	number of docs judged	disagreements
51	211	6
62	400	157
67	400	68
95	400	110
127	400	106



# Impact of interjudge disagreement

- Judges disagree a lot. Does that mean that the results of information retrieval experiments are meaningless?
  - No.
- Large impact on absolute performance numbers
- Virtually no impact on ranking of systems
- Supposes we want to know if algorithm A is better than algorithm B
- An information retrieval experiment will give us a reliable answer to this question . . .
  - . . . even if there is a lot of disagreement between judges.



# Evaluation at Large Search Engines

- Recall is difficult to measure on the web
- Search engines often use precision at top  $k$ , e.g.,  $k = 10$  . . .
  - . . . or use measures that reward you more for getting rank 1 right than for getting rank 10 right.
- Search engines also use non-relevance-based measures.
  - Example 1: clickthrough on first result
  - Not very reliable if you look at a single clickthrough (you may realize after clicking that the summary was misleading and the document is nonrelevant) . . .
    - . . . but pretty reliable in the aggregate.
  - Example 2: Ongoing studies of user behavior in the lab – recall last lecture
  - Example 3: A/B testing





# A/B testing

- Purpose: Test a single innovation
- Prerequisite: You have a large search engine up and running.
- Have most users use old system
- Divert a small proportion of traffic (e.g., 1%) to the new system that includes the innovation
- Evaluate with an “automatic” measure like clickthrough on first result
- Now we can directly see if the innovation does improve user happiness.
- Probably the evaluation methodology that large search engines trust most



# Results Summary



# How do we present results to users?

- Originally, as a list – aka “10 blue links”
- How should each document in the list be described?
- This description is crucial.
- The user often can identify good hits (= relevant hits) based on the description.
- No need to actually view any document



# Each result is

- Most commonly: doc title, url, some metadata . . .
- . . . and a summary
- How do we “compute” the summary?



# Summaries / Snippets

- Two basic kinds: (i) static (ii) dynamic
- A **static summary** of a document is always the same, regardless of the query that was issued by the user.
- **Dynamic summaries** are **query-dependent**. They attempt to explain why the document was retrieved for the query at hand

[https://en.wikipedia.org › wiki › Tower\\_of\\_fields](https://en.wikipedia.org/wiki/Tower_of_fields) ▼

## Tower of fields - Wikipedia

In mathematics, a **tower of fields** is a sequence of **field** extensions  $F_0 \subseteq F_1 \subseteq \dots \subseteq F_n \subseteq \dots$ . The name comes from such sequences often being written in the form. A **tower of fields** may be finite or infinite.



# Static Summaries

- In typical systems, the static summary is a subset of the document.
- Simplest heuristic: the first 50 or so words of the document
- More sophisticated: extract from each document a set of “key” sentences
  - Simple NLP heuristics to score each sentence
  - Summary is made up of top-scoring sentences.
  - Machine learning approach
- Most sophisticated: complex NLP to synthesize/generate a summary
  - For most IR applications: not quite ready for prime time yet

[https://en.wikipedia.org › wiki › Volkswagen](https://en.wikipedia.org/wiki/Volkswagen) ▼

## Volkswagen - Wikipedia

listen)), is a German motor vehicle manufacturer founded in 1937 by the German Labour Front, known for the iconic Beetle and headquartered in Wolfsburg. It is ...

**Founder:** [German Labour Front](#)

**Founded:** 1937; 84 years ago

**Key people:** Ralf Brandstaetter (brand CEO)

**Industry:** [Automotive](#)



**SAPIENZA**  
UNIVERSITÀ DI ROMA

# Dynamic Summaries

- Present one or more “windows” or snippets within the document that contain several of the query terms.
- Prefer snippets in which query terms occurred as a phrase
- Prefer snippets in which query terms occurred jointly in a small window
- The summary that is computed this way gives the entire content of the window – all terms, not just the query terms.



# Generating Dynamic Summaries

- Where do we get these other terms in the snippet from?
- We cannot construct a dynamic summary from the positional inverted index – at least not efficiently.
- We need to cache documents.
- The positional index tells us: query term occurs at position 4378 in the document.
- Byte offset or word offset?
- Note that the cached copy can be outdated
- Don't cache very long documents – just cache a short prefix





# Dynamic Summaries

- Real estate on the search result page is limited → snippets must be short . . .
  - . . . but snippets must be long enough to be meaningful.
- Snippets should communicate whether and how the document answers the query.
- Ideally: linguistically well-formed snippets
- Ideally: the snippet should answer the query, so we don't have to look at the document.
- Dynamic summaries are a big part of user happiness because . . .
  - . . . we can quickly scan them to find the relevant document we then click on.
  - . . . in many cases, we don't have to click at all and save time.



# Main Takeaways

- Introduction to evaluation: Measures of an IR system
- Evaluation of unranked and ranked retrieval
- Evaluation benchmarks
- Result summaries

