BITS Pilani
Pilani Campus

Information Retrieval

Abhishek
January 2020

innovate achieve lead

BITS Pilani
Pilani Campus

CS F469, Information Retrieval

Lecture topics: Evaluation in IR

innovate achieve lead

BITS Pilani
Pilani Campus

Most of these slides are based on:

https://web.stanford.edu/class/cs276/
https://www.inf.unibz.it/~ricci/ISR/
https://www.cis.uni-muenchen.de/~hs/teach/14s/ir/

This Lecture

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

Measures of IR Systems

Measures of IR Systems

- 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

Measures of IR Systems

- All of the preceding criteria are **measurable**:
  - we can quantify speed / size / money

Measures of IR Systems

- 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.

Measures of IR Systems

- 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?

- 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
- 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.
  - Document collection
  - A benchmark suite of queries
  - An assessment of the relevance of each query-document pair

Relevance: query vs. information need

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 the textbook: we talk about query-document relevance judgments even though we mean information-need-document relevance judgments.

Evaluation Benchmarks

What we need for a benchmark

- 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

Public Benchmarking Datasets

- Series of datasets/competitions organized by Text Retrieval Conference (TREC)
- 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.

Evaluations in Unranked Retrieval

Precision and Recall

- **Precision:** fraction of retrieved docs that are relevant =

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

- **Recall:** fraction of relevant docs that are retrieved =

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

	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/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?

F1 score

$$F_1 = \left(\frac{2}{\mathrm{recall}^{-1} + \mathrm{precision}^{-1}} \right) = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}.$$

- A balance measure
- If P=0.9, R=0.2, what will be F1?
- If P=0.2, R=0.99, what will be F1?
- If P=0.9, R=0.9, what will be F1?

F1 score

$$F_1 = \left(\frac{2}{\mathrm{recall}^{-1} + \mathrm{precision}^{-1}}\right) = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}.$$

- A balance measure
- If P=0.9, R=0.2, what will be F1? : **0.3**
- If P=0.2, R=0.99, what will be F1? : **0.33**
- If P=0.9, R=0.9, what will be F1? : **0.9**

Example for precision, recall and F1

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

- $P = 20/(20 + 40) = 1/3$
- $R = 20/(20 + 60) = 1/4$
- $F_1 = 2\frac{1}{\frac{1}{\frac{1}{3}} + \frac{1}{\frac{1}{4}}} = 2/7$

Accuracy

- Why do we use complex measures like **precision**, **recall**, and **F1**?
- Why not something simple like **accuracy**?

Accuracy

- Why do we use complex measures like **precision**, **recall**, and **F1**?
- Why not something simple like **accuracy**?
  - What is retrieved is categorized by the IR System as "relevant" and what is not retrieved is classified as "non relevant"
- Accuracy is the fraction of decisions (relevant/non relevant) that are correct.
- In terms of the contingency table:
  - accuracy = (TP + TN)/(TP + FP + FN + TN)
- Why is this not a very useful evaluation measure in IR?

Accuracy

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

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

Accuracy

- Compute precision, recall, F1 and Accuracy 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. What will its accuracy?

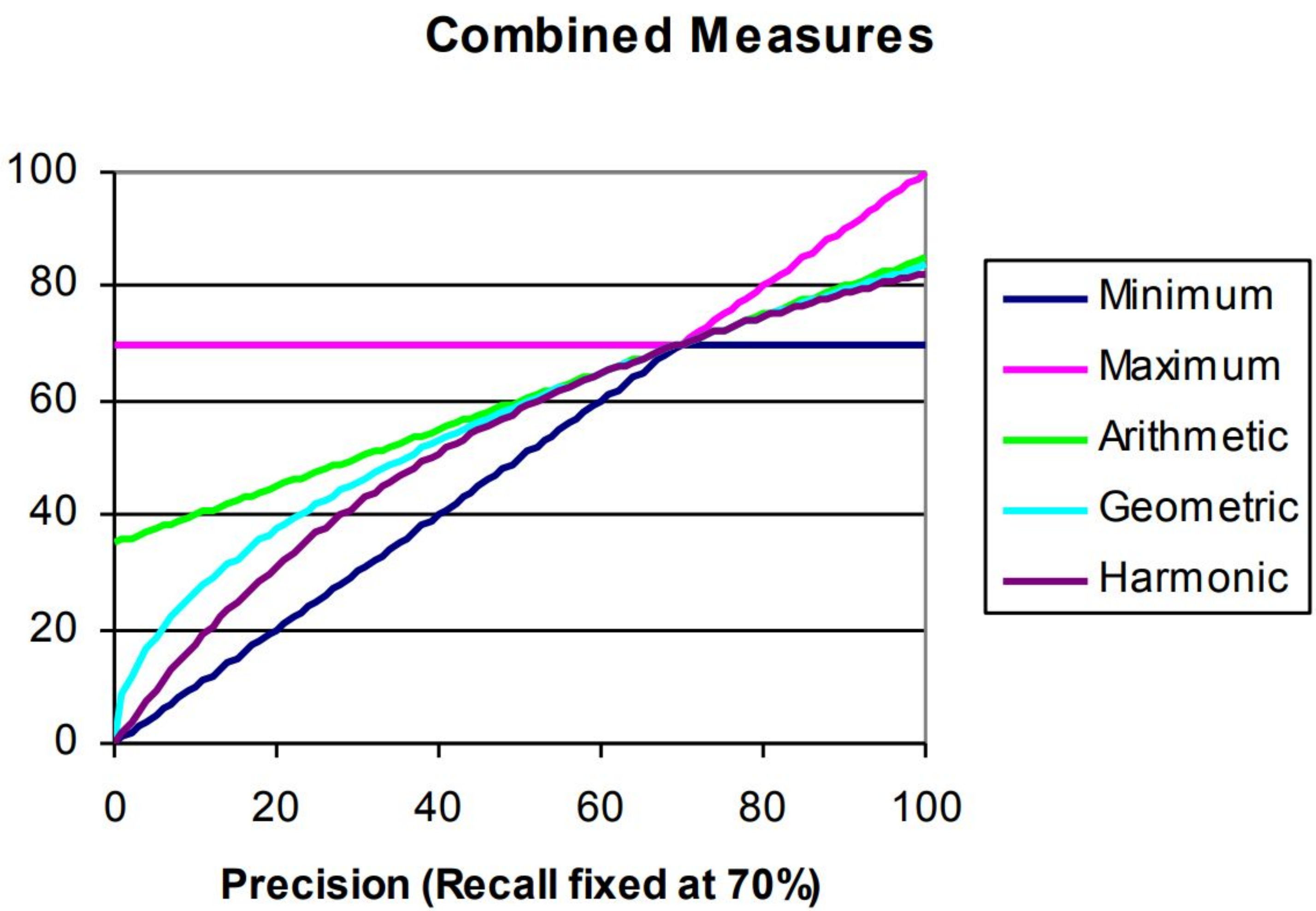
Why harmonic mean?

- Why don’t we use a different mean of P and R as a measure?
- e.g., the arithmetic mean

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 snoogle search engine – which is too high.
- 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.

F1 and other averages



Evaluations in Ranked Retrieval

Ranked Based Measures

Ranked Based Measures

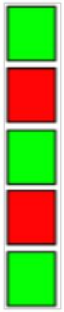
- **Binary relevance**
  - Precision@K (P@K)
  - Mean Average Precision (MAP)
  - Mean Reciprocal Rank (MRR)

- **Multiple levels of relevance**
  - Normalized Discounted Cumulative Gain (NDCG)

Precision@K

Precision@K

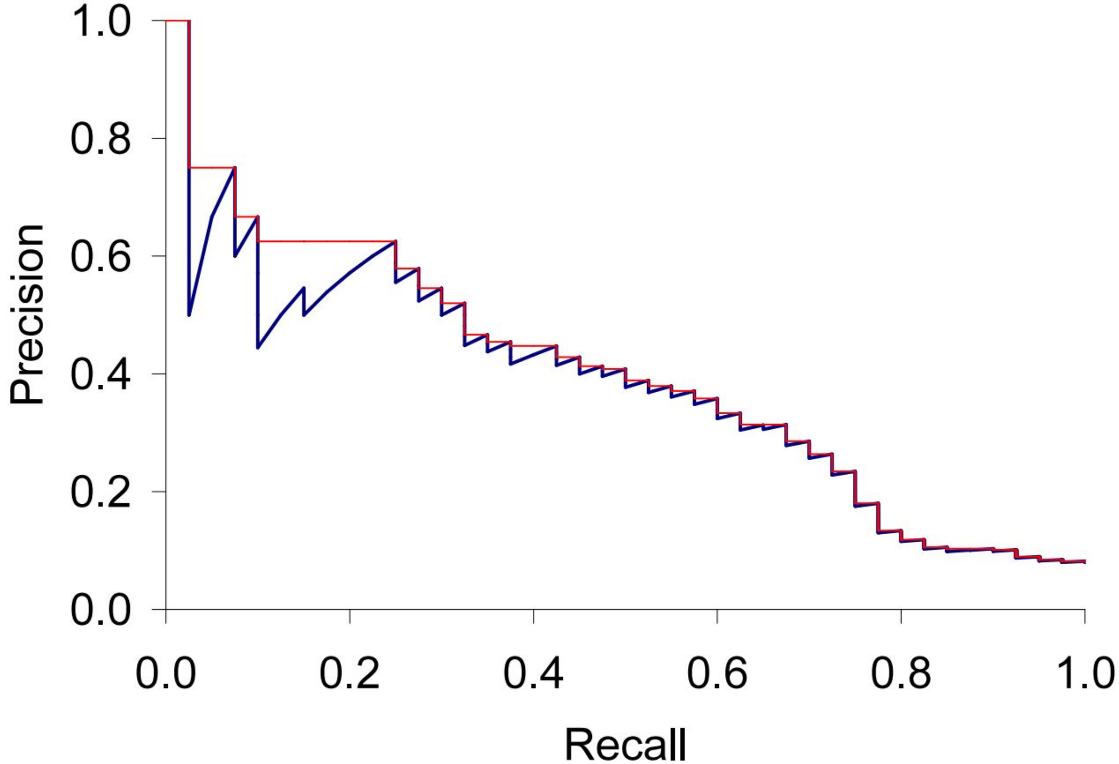
- Set a rank threshold K.
- Compute % relevant in top K.
- Ignores documents ranked lower than K.
- Example:
  - Prec@3 of 2/3.
  - Prec@4 of 2/4.
  - Prec@5 of 3/5.



- In similar fashion we have Recall@K.

Precision Recall Curve

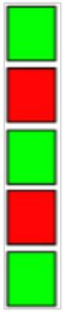
Precision Recall Curve



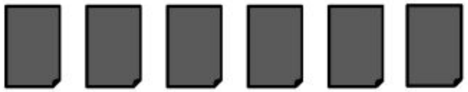
- Each point corresponds to a result for the top k ranked hits (k = 1, 2, 3, ..).
- Interpolation (in red): Take maximum of all future points.
- Rationale for interpolation: The user is willing to look at more stuff if both precision and recall get better.

Mean Average Precision

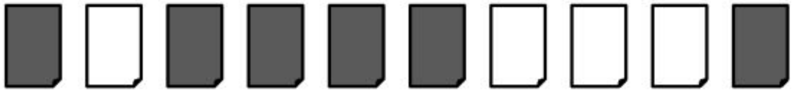
Mean Average Precision

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

Average Precision

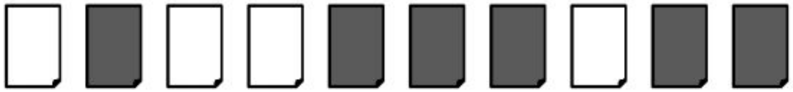
 = the relevant documents

Ranking #1



Recall	0.17	0.17	0.33	0.5	0.67	0.83	0.83	0.83	0.83	1.0
Precision	1.0	0.5	0.67	0.75	0.8	0.83	0.71	0.63	0.56	0.6

Ranking #2

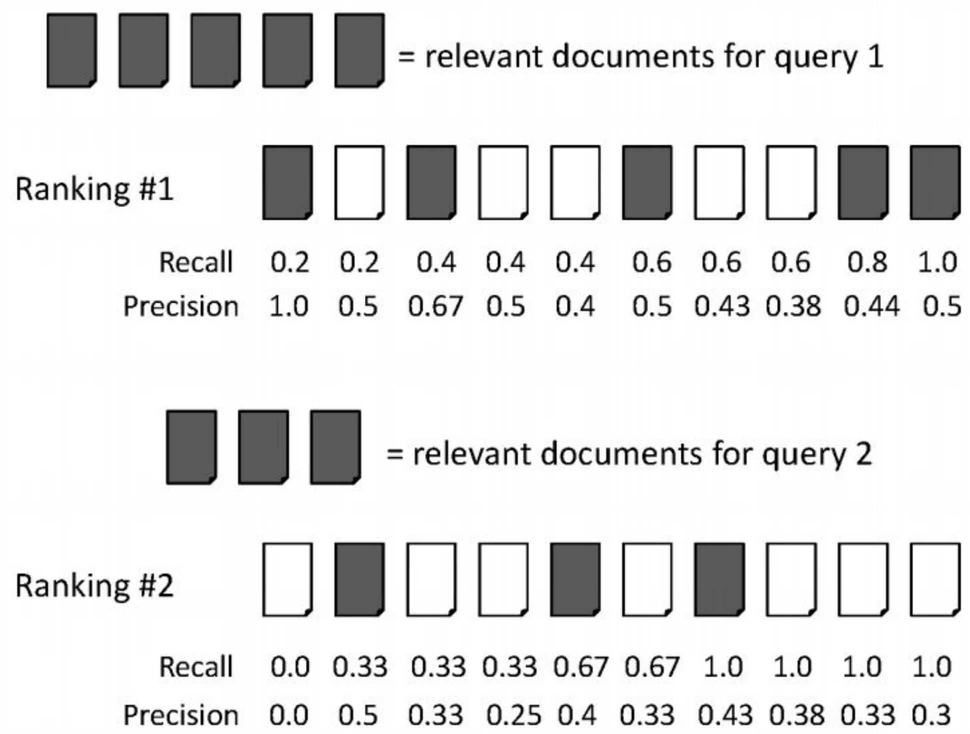


Recall	0.0	0.17	0.17	0.17	0.33	0.5	0.67	0.67	0.83	1.0
Precision	0.0	0.5	0.33	0.25	0.4	0.5	0.57	0.5	0.56	0.6

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

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

Mean Average Precision (MAP)



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

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

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

Mean Average Precision (MAP)

- If a relevant document never gets retrieved, we assume the precision corresponding to that relevant document 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.

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.

Discounted Cumulative Gain

- Popular measure for evaluating web search and related tasks.
- Two assumptions:
  - Highly relevant documents are more useful than marginally relevant documents
  - 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(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, …r_n (in ranked order)
  - CG = $r_1+r_2+\ldots r_n$
- Discounted Cumulative Gain (DCG) at rank n
  - DCG = $r_1 + r_2/\log_2 2 + r_3/\log_2 3 + \ldots r_n/\log_2 n$
  - We may use any base for the logarithm

Discounted Cumulative Gain

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

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

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

NDCG for summarizing rankings

- Normalized Discounted 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.
- Normalization useful for contrasting queries with varying numbers of relevant results.
- NDCG is now quite popular in evaluating Web search

NDCG: Example

i	Ground Truth		Ranking Function1		Ranking Function2	
	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
	$NDCG_{GT}=1.00$		$NDCG_{RF1}=1.00$		$NDCG_{RF2}=0.9203$	

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

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

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

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

What if the results are not in a list?

- Suppose there’s only one Relevant Document.
- Scenarios:
  - known-item search
  - navigational queries
  - looking for a fact

- Search duration ~ Rank of the answer
  - measures a user’s effort

Mean Reciprocal Rank

- Consider rank position, K, of first relevant doc
  - Could be – only clicked doc

- Reciprocal Rank score = **1 / K**
- **MRR** is the mean **RR** across multiple queries

Evaluation at large search engines

- Search engines have test collections of queries and hand-ranked results.
- Recall is difficult to measure on the web (why?)
- Search engines often use top k precision, e.g., k=10
- … or measures that reward you more for getting rank 1 right than for getting rank 10 right: NDCG (Normalized Cumulative Discounted Gain)
- Search engines also use non-relevance-based measures:
  - Clickthrough on first result: Not very reliable if you look at a single clickthrough … but pretty reliable in the aggregate.
  - Studies of user behavior in the lab.
  - 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.

Recommended Readings

- Introduction to Google Search Quality: https://googleblog.blogspot.com/2008/05/introduction-to-google-search-quality.html

Thank You!