Evaluating search engines

CE-324: Modern Information Retrieval

Sharif University of Technology

M. Soleymani

Fall 2015

Most slides have been adapted from: Profs. Manning, Nayak & Raghavan (CS-276, Stanford)

Why do we need system evaluation?

- How do we know which of the already introduced techniques are effective in which applications?
  - Should we use stop lists? Should we stem? Should we use inverse document frequency weighting?

- We need evaluation to demonstrate the superior performance of novel techniques on representative document collections.

User happiness is elusive to measure

- The key utility measure is user happiness.
  - How satisfied is each user with the obtained results?
  - The most common proxy to measure human satisfaction is ***relevance*** of search results to the posed information

- How do you measure relevance?

- Relevance measurement requires 3 elements:
  1. A benchmark doc collection
  2. A benchmark suite of information needs
  3. A usually binary assessment of either <u>Relevant</u> or <u>Nonrelevant</u> for each information needs and each document
     - Some work on more-than-binary, but not the standard

Evaluating an IR system

- Note: **information need** is translated into a **query**
- User happiness can only be measured by **relevance** to an information need, not by relevance to queries.
- Evaluate whether doc addresses information need
  - not whether it has these words

Standard relevance benchmarks

- TREC: NIST has run a large IR test bed for many years
- Reuters and other benchmark doc collections
- "Retrieval tasks" specified
  - sometimes as queries
- Human experts mark, for each query and for each doc, <u>Relevant</u> or <u>Nonrelevant</u>
  - or at least for subset of docs that some systems (participating in the competitions) returned for that query

TREC Relevance Judgements

Humans decide which document–query pairs are relevant.

Evaluation metrics

	Relevant	Non-relevant	Total
Retrieved	A	B	A+B
Not retrieved	C	D	C+D
Total	A+C	B+D	A+B+C+D

Recall: proportion of retrieved items amongst the relevant items ($\frac{A}{A+C}$)

Precision: proportion of relevant items amongst retrieved items ($\frac{A}{A+B}$)

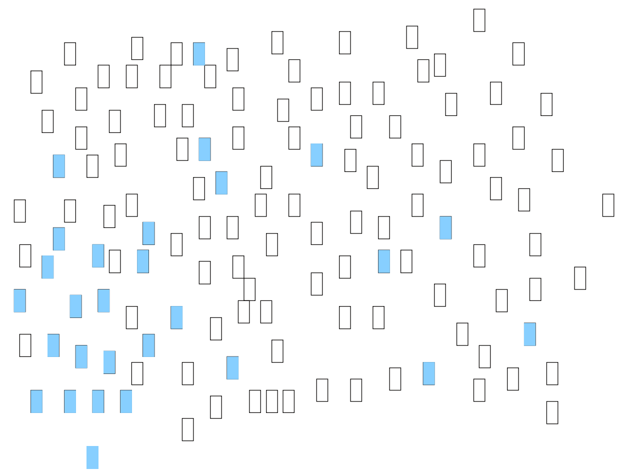
Accuracy: proportion of correctly classified items as relevant/irrelevant ($\frac{A+D}{A+B+C+D}$)

Recall: [0..1]; Precision: [0..1]; Accuracy: [0..1]

Accuracy is not a good measure for IR, as it conflates performance on relevant items (A) with performance on irrelevant (uninteresting) items (D)

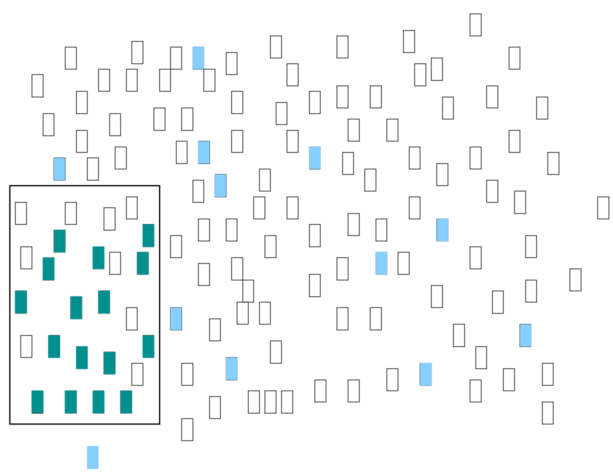
Recall and Precision

- All documents: A+B+C+D = 130
- Relevant documents for a given query: A+C = 28



Recall and Precision: System 1

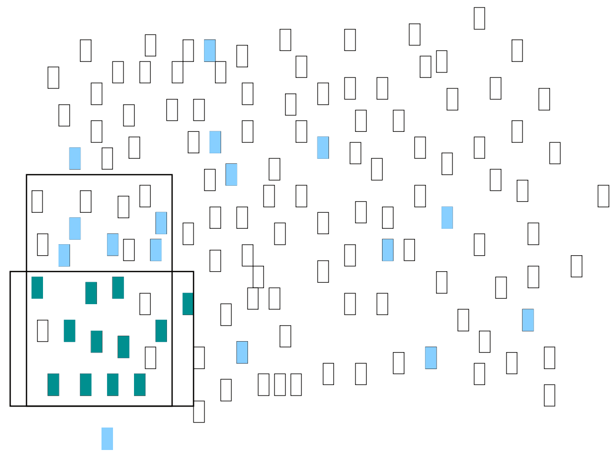
- System 1 retrieves 25 items: $(A{+}B)_1 = 25$
- Relevant and retrieved items: $A_1 = 16$



$R_1 = \frac{A_1}{A+C} = \frac{16}{28} = .57$

$P_1 = \frac{A_1}{(A+B)_1} = \frac{16}{25} = .64$

$A_1 = \frac{A_1+D_1}{A+B+C+D} = \frac{16+93}{130} = .84$



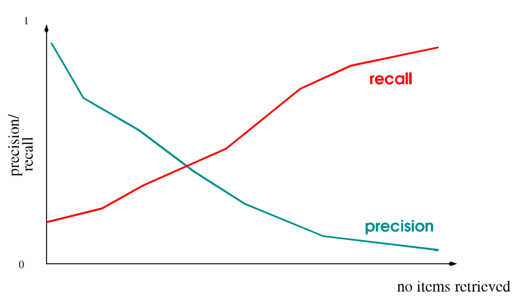
- System B retrieves set $(A+B)_2$ = 15 items
- A_2 = 12

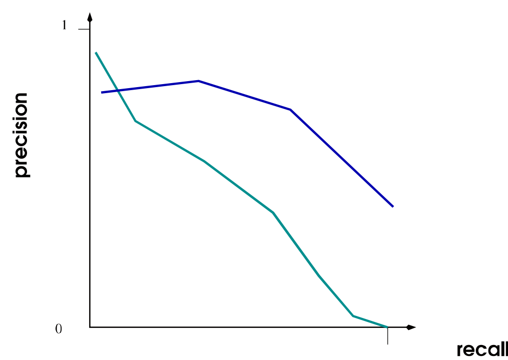
$R_2 = \frac{12}{28} = .43$

$P_2 = \frac{12}{15} = .8$

$A_2 = \frac{12+99}{130} = .85$

Recall-precision curve





- Plotting precision and recall (versus no. of documents retrieved) shows inverse relationship between precison and recall
- Precision/recall cross-over can be used as combined evaluation measure

- Plotting precision versus recall gives recall-precision curve
- Area under normalised recall-precision curve can be used as evaluation measure

Recall-criticality and precision-criticality

- Inverse relationship between precision and recall forces general systems to go for compromise between them
- But some tasks particularly need good precision whereas others need good recall:

Precision-critical task	Recall-critical task
Little time available	Time matters less
A small set of relevant documents answers the information need	One cannot afford to miss a single document
Potentially many documents might fill the information need (redundantly)	Need to see *each* relevant document
Example: web search for factual information	Example: patent search

The problem of determining recall

- Recall problem: for a collection of non-trivial size, it becomes impossible to inspect each document
- It would take 6500 hours to judge 800,000 documents for one query (30 sec/document)
- Pooling addresses this problem

Unranked retrieval evaluation: Precision and Recall

- **Precision**: P(relevant|retrieved)
  - fraction of retrieved docs that are relevant
- **Recall**: P(retrieved|relevant)
  - fraction of relevant docs that are retrieved

	Relevant	Nonrelevant
Retrieved	tp	fp
Not Retrieved	fn	tn

Precision P = tp/(tp + fp)

Recall R = tp/(tp + fn)

Accuracy measure for evaluation?

- **Accuracy:** fraction of classifications that are correct
  - evaluation measure in machine learning classification works
- The **accuracy** of an engine:
  - (tp + tn) / (tp + fp + fn + tn)
- Given a query, an engine classifies each doc as "Relevant" or "Nonrelevant"
- Why is this not a very useful evaluation measure in IR?

Why not just use accuracy?

- How to build a 99.9999% accurate search engine on a low budget….
  - The snoogle search engine below always returns 0 results (“No matching results found”), regardless of the query
  - Since many more non-relevant docs than relevant ones

- People *want to find something* and have a certain tolerance for junk.

Precision/Recall

- Retrieving all docs for all queries!
  - High recall but low precision

- Recall is a non-decreasing function of the number of docs retrieved

- In a good system, precision decreases as either the number of docs retrieved or recall increases
  - This is not a theorem, but a result with strong empirical confirmation

A combined measure: F

- Combined measure: **F measure**
  - allows us to trade off precision against recall
  - weighted harmonic mean of P and R

$$\beta^2 = \frac{1-\alpha}{\alpha}$$

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

- What value range of weights recall higher than precision?

A combined measure: *F*

- People usually use balanced F (β = 1 or α = ½)

$$F = F_{\beta=1}$$

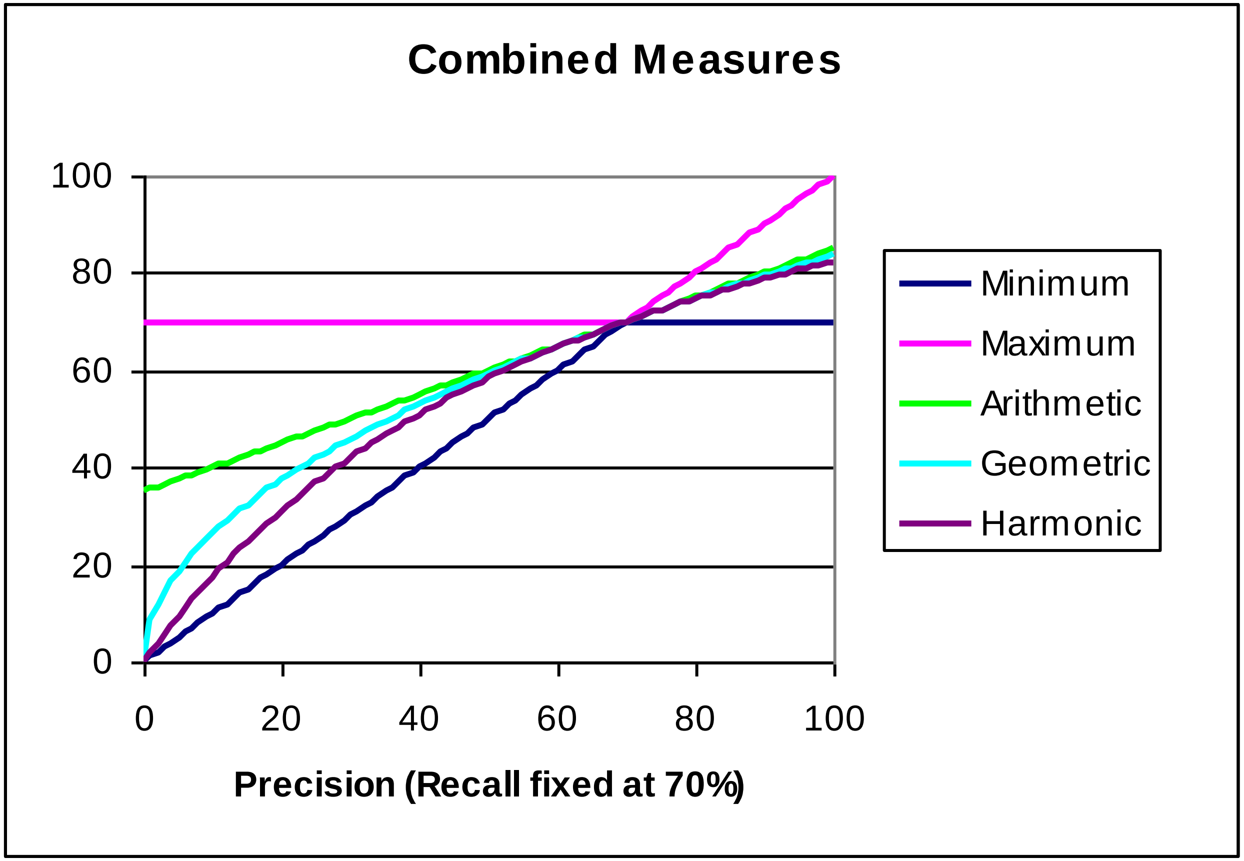
$$F = \frac{2PR}{P + R}$$

- harmonic mean of P and R: $\frac{1}{F} = \frac{1}{2}\left(\frac{1}{P} + \frac{1}{R}\right)$

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 50% for "return-everything" search 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.

F_1 and other averages



Harmonic mean is a conservative average

We can view the harmonic mean as a kind of soft minimum

Evaluation for unranked retrieval (example)

	Relevant	Not relevant	
Retrieved	20	40	60
Not retrieved	60	1,000,000	1,000,060
	80	1,000,040	1,000,120

$$Pr = \frac{tp}{tp + fp} = \frac{20}{20 + 40} = \frac{1}{3}$$

$$Re = \frac{tp}{tp + fn} = \frac{20}{20 + 60} = \frac{1}{4}$$

$$F_1 = \frac{2 \times \frac{1}{3} \times \frac{1}{4}}{\frac{1}{3} + \frac{1}{4}} = \frac{2}{7}$$

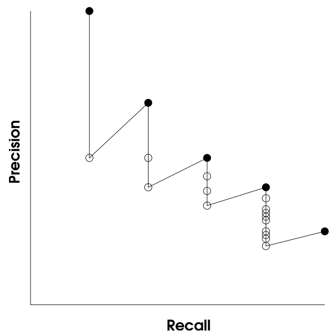
Evaluating ranked results

- Precision, recall and F are measures for (unranked) sets.
  - We can easily turn set measures into measures of ranked lists.

- Evaluation of ranked results:
  - Taking various numbers of top returned docs (recall levels)
    - Sets of retrieved docs are given by the top k retrieved docs.
      - Just compute the set measure for each "prefix": the top 1, top 2, top 3, top 4, and etc results
  - Doing this for precision and recall gives you a ***precision-recall curve***

Precision and recall in ranked IR engines

- With ranked list of return documents there are many P/R data points
- Sensible P/R data points are those after each new relevant document has been seen (black points)

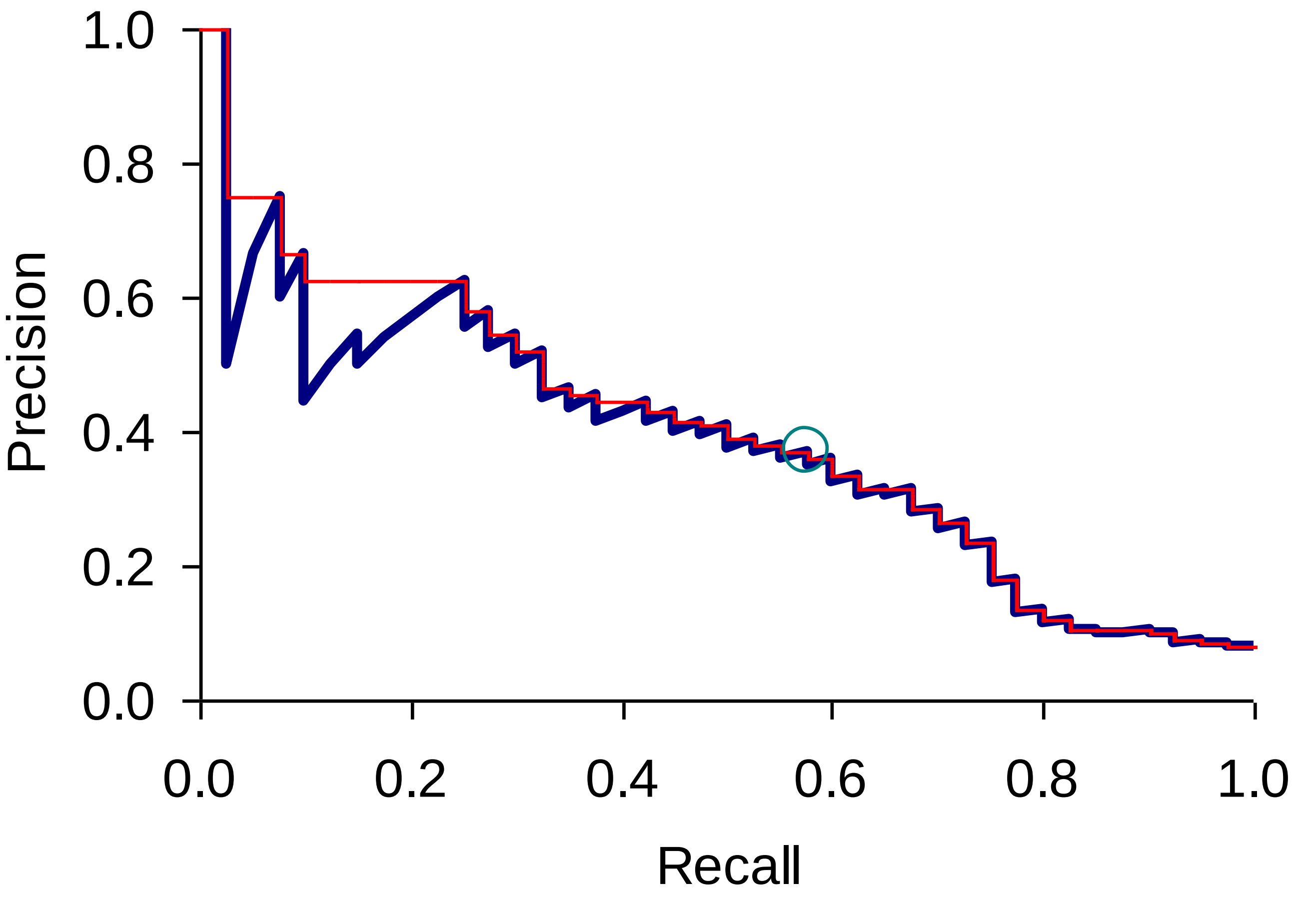


Query 1			
Rank	Relev.	R	P
1	X	0.20	1.00
2		“	0.50
3	X	0.40	0.67
4		”	0.50
5		”	0.40
6	X	0.60	0.50
7		”	0.43
8		”	0.38
9		”	0.33
10	X	0.80	0.40
11		”	0.36
12		”	0.33
13		”	0.31
14		”	0.29
15		”	0.27
16		”	0.25
17		”	0.24
18		”	0.22
19		”	0.21
20	X	1.00	0.25

Summary IR measures

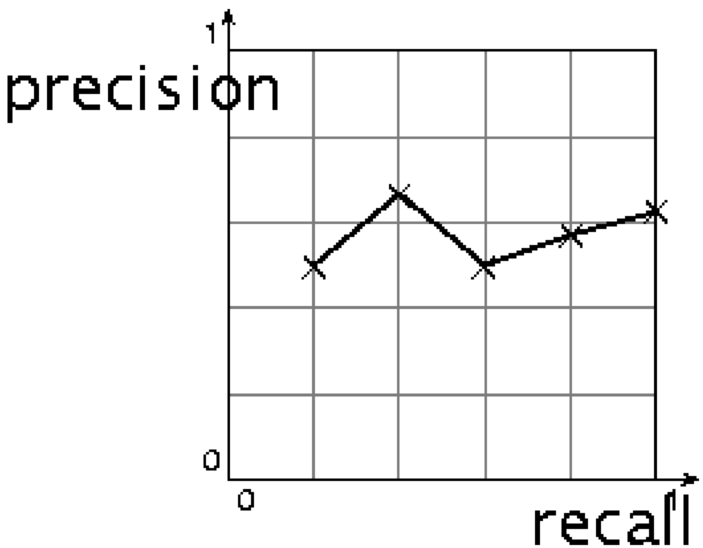
- Precision at a certain rank: P(100)
- Precision at a certain recall value: P(R=.2)
- Precision at last relevant document: P(last_relev)
- Recall at a fixed rank: R(100)
- Recall at a certain precison value: R(P=.1)

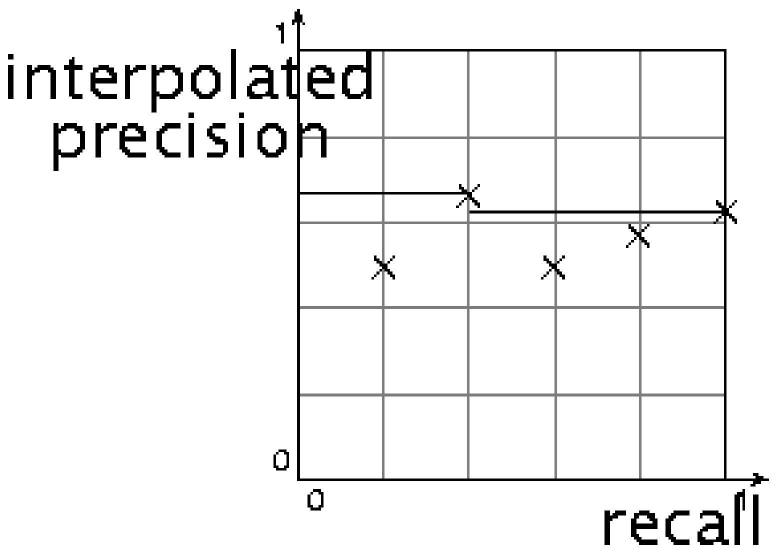
A precision-recall curve



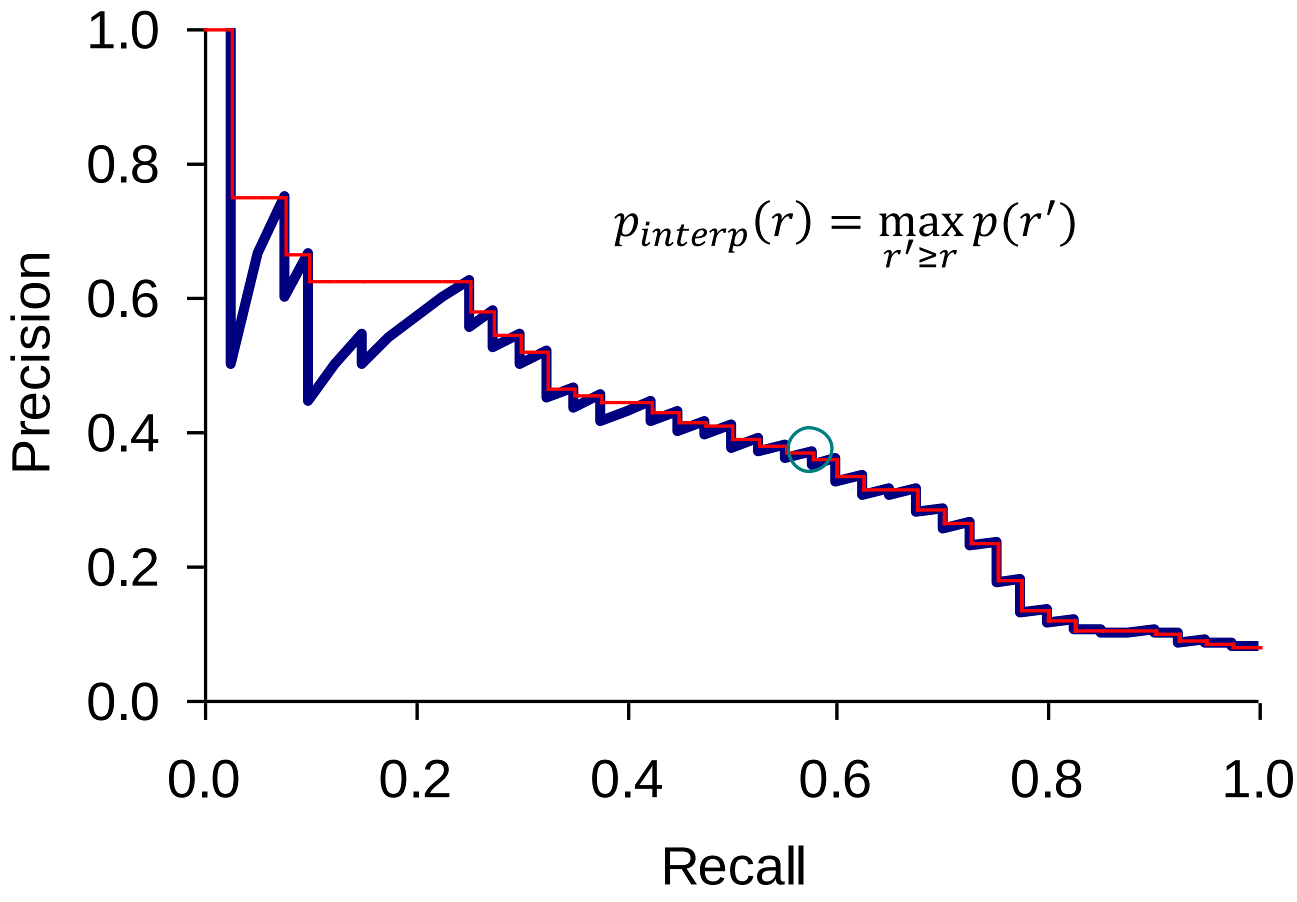
Interpolated precision

- Interpolation: 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.
  - If locally precision increases with increasing recall, then you should get to count that…





An interpolated precision-recall curve



Averaging over queries

- Precision-recall graph for <u>one query</u>
  - It isn't a very sensible thing to look at
- <u>Average</u> performance over a whole bunch of queries.
- But there's a technical issue:
  - Precision-recall: only place some points on the graph
  - How do you determine a value (interpolate) between the points?

Evaluation

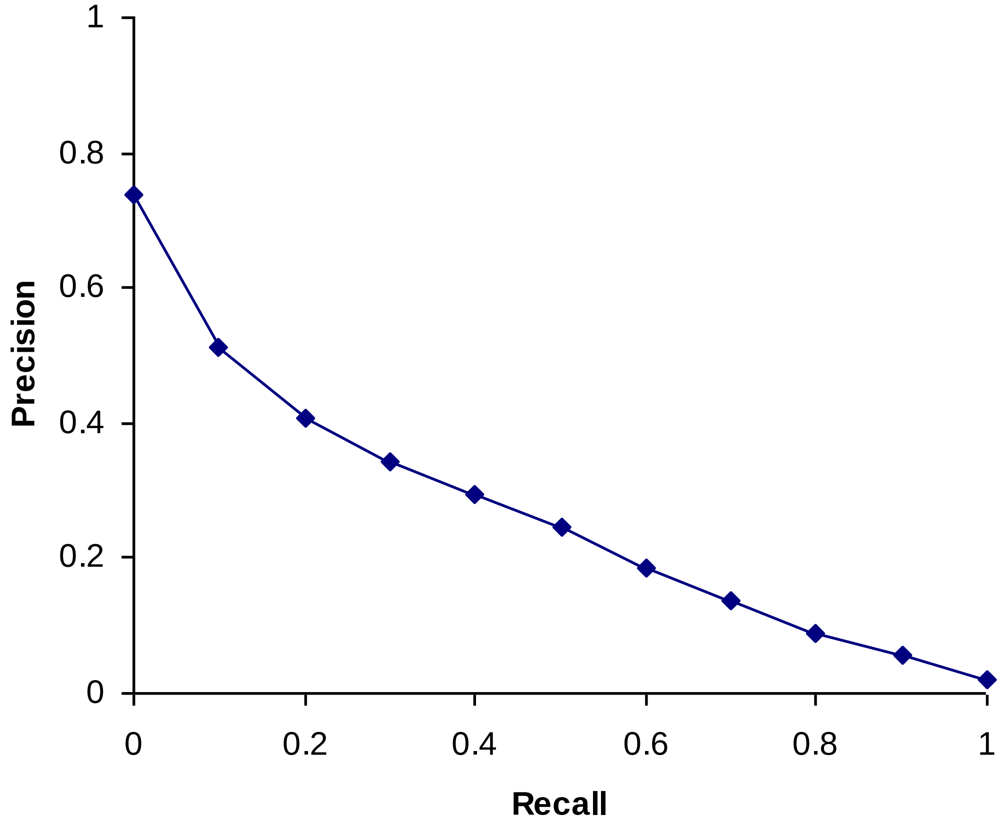
- Graphs are good, but people want summary measures!
  - 11-point interpolated average precision
    - Precision at fixed retrieval level
    - MAP
    - R-precision

11-point interpolated average precision

- The standard measure in the early TREC competitions
- Precision at 11 levels of recall varying from 0 to 1
  - by tenths of the docs using interpolation and average them
- Evaluates performance at all recall levels (0, 0.1, 0.2, …,1)

Typical (good) 11 point precisions

- SabIR/Cornell 8A1
  - 11pt precision from TREC 8 (1999)



Mean Average Precision (MAP)

- Mean Average Precision (MAP)
  - Average precision is obtained for the top *k* docs, each time a relevant doc is retrieved

  - MAP for query collection is arithmetic average
    - Macro-averaging: each query counts equally

Mean Average Precision (MAP)

- Q: set of information needs
- Set of relevant docs to $q_j \in Q$: $d_j^{(1)}, d_j^{(2)}, \dots, d_j^{(m_j)}$
- $R_j^{(i)}$: set of ranked retrieval results from the top until reaching $d_j^{(i)}$

$$MAP(Q) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m_j}\sum_{i=1}^{m_j}\text{Precision}(R_j^{(i)})$$

For queries for which $k' < k$ documents are retrieved, the last summation is done up to k'.

Mean Average Precision (MAP) (example)

Query 1		
Rank		$P(doc_i)$
1	X	1.00
2		
3	X	0.67
4		
5		
6	X	0.50
7		
8		
9		
10	X	0.40
11		
12		
13		
14		
15		
16		
17		
18		
19		
20	X	0.25
AVG:		0.564

Query 2		
Rank		$P(doc_i)$
1	X	1.00
2		
3	X	0.67
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15	X	0.2
AVG:		0.623

$$MAP = \frac{0.564 + 0.623}{2} = 0.594$$

R-precision

- Rel: A known (though perhaps incomplete) set of relevant docs

- Calculate precision of the top $|Rel|$ docs returned
  - r relevant among the top $|Rel|$ results $\Rightarrow$ for this set $P = R = \frac{r}{|Rel|}$

- Perfect system could score 1.0.

Precision-at-k

- **Precision-at-*k***: Precision of top *k* results
- Perhaps appropriate for most of web searches
  - people want good matches on the first one or two results pages
- Does not need any estimate of the size of relevant set
  - But: averages badly and has an arbitrary parameter of *k*

Precision at k (example)

Rank n	Doc
1	d_{12}
2	d_{123}
3	d_{4}
4	d_{57}
5	d_{157}
6	d_{222}
7	d_{24}
8	d_{26}
9	d_{77}
10	d_{90}

- Blue documents are relevant.
- P@n: P@3=0.33, P@5=0.2, P@8=0.25
- R@n: R@3=0.33, R@5=0.33, R@8=0.66

Variance of performance

"The variance in performance of the same system across queries"

is much greater than

"the variance of different systems on the same query."

- There are easy information needs and hard ones!

Creating Test Collections

for IR Evaluation

TREC

- TREC Ad Hoc task from first 8 TRECs is standard IR task
  - 50 detailed information needs for each year
  - Human evaluation of pooled results returned

- A TREC query (TREC 5): Example

  <top>

  <num> Number: 225

  <desc> Description:

  What is the main function of the Federal Emergency Management Agency (FEMA) and the funding level provided to meet emergencies? Also, what resources are available to FEMA such as people, equipment, facilities?

  </top>

Other standard relevance benchmarks

- GOV2
  - Another TREC/NIST collection
  - 25 million web pages
  - Largest collection that is easily available
  - But still 3 orders of magnitude smaller than what Google/Yahoo/MSN index
- NTCIR
  - East Asian language and cross-language information retrieval
- Cross Language Evaluation Forum (CLEF)
  - European languages and cross-language information retrieval.

From doc collections to test collections

- Test queries (information needs)
  - Must be germane to docs available
  - Best designed by domain experts
  - Random query terms generally not a good idea

- Relevance assessments
  - Human judges, time-consuming
  - Pooling
  - Are human panels perfect?

Kappa measure for inter-judge (dis)agreement

- **Kappa measure**
  - Agreement measure among judges
  - Designed for categorical judgments
  - Corrects for chance agreement

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

- $P(A)$: proportion of time judges agree
- $P(E)$: what agreement would be by chance

- $Kappa = 0$ for chance agreement, 1 for total agreement.

Kappa measure: example

$P(A)?\ P(E)?$

Number of docs	Judge 1	Judge 2
300	Relevant	Relevant
70	Nonrelevant	Nonrelevant
20	Relevant	Nonrelevant
10	Nonrelevant	Relevant

Kappa example

$$P(A) = 370/400 = 0.925$$

$$P(\text{nonrelevant}) = (10 + 20 + 70 + 70)/800 = 0.2125$$

$$P(\text{relevant}) = (10 + 20 + 300 + 300)/800 = 0.7878$$

$$P(E) = 0.2125^2 + 0.7878^2 = 0.665$$

$$Kappa = (0.925 - 0.665)/(1 - 0.665) = 0.776$$

Kappa

- $Kappa > 0.8$
  - good agreement
- $0.67 < Kappa < 0.8$
  - "tentative conclusions" (Carletta '96)
- $Kappa < 0.67$
  - A dubious basis for evaluation

- Precise cutoffs depends on purpose of study

- For >2 judges: average pairwise kappas

Impact of inter-judge agreement

- Impact on **absolute performance** measure can be significant (0.32 vs 0.39)

- Little impact on ranking of different systems or **relative** performance

- “Algorithm A is better than algorithm B?”
  - A standard information retrieval experiment will give us a reliable answer to this question.

Difficulties in (Precision/Recall) system evaluation

- Should average over large doc collection/query ensembles
- Need human relevance assessments
  - People aren't reliable assessors
- Assessments have to be binary
  - Nuanced assessments?
- Heavily skewed by collection/authorship
  - Results may not translate from one domain to another

Normalized Discounted Cumulative Gain (NDCG)

- Q: set of information needs
- List of relevant docs to $q_j \in Q$: $d_j^{(1)}, d_j^{(2)}, \ldots$
- $R(d, q)$: graded relevance of doc d to query q
- $Z_{j,k}$ is a normalization factor calculated to make it so that a perfect ranking's NDCG at k for query j is 1.
- For queries for which $k' < k$ docs are retrieved, the last sum is done up to k'.

$$NDCG(Q,k) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} Z_{j,k} \sum_{i=1}^{k} \frac{2^{R(d_j^{(i)}, q_j)} - 1}{\log_2 i + 1}$$

Highly relevant documents are more useful

The gain of each result discounted at lower ranks

A broader perspective for IR system evaluation

- System issues
- User utility

System issues

- How fast does it index?
  - Number of documents (or bytes) per hour
- How fast does it search?
  - Latency as a function of queries per second
- How large is its document collection?
- Expressiveness of query language
  - Ability to express complex information needs
  - Speed on complex queries

All of the preceding criteria are *measurable*: we can quantify speed/size
we can also make expressiveness precise

User utility

- The key measure is user happiness

- Factors of user happiness include:
  - Speed of response
  - Uncluttered User Interface
  - Most important: **relevance**
    - Speed of response and size of index are factors but blindingly fast, useless answers won't make a user happy

- Quantifying aggregate user happiness based on relevance, speed, and user interface of the system

- User satisfaction can be measured by running user studies

Measuring user happiness

"Issue: who is the user we are trying to make happy?"

- Web search engine: searcher
- Web search engine: advertiser
- Ecommerce: buyer
- Ecommerce: seller
- Enterprise: CEO

Measuring user happiness

"Issue: 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
- Ecommerce: seller
- Enterprise: CEO

Measuring user happiness

"Issue: who is the user we are trying to make happy?"

- Web search engine: searcher
- Web search engine: advertiser
- **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

Measuring user happiness

"Issue: who is the user we are trying to make happy?"

- Web search engine: searcher
- Web search engine: advertiser
- Ecommerce: buyer
- Ecommerce: seller
- **Enterprise: CEO**
  - Success: Employees are more productive (because of effective search)
  - Measure: profit of the company

Resources for this lecture

- IIR 8
- MIR Chapter 3
- MG 4.5
- Carbonell and Goldstein 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. SIGIR 21.