Evaluation and Relevance

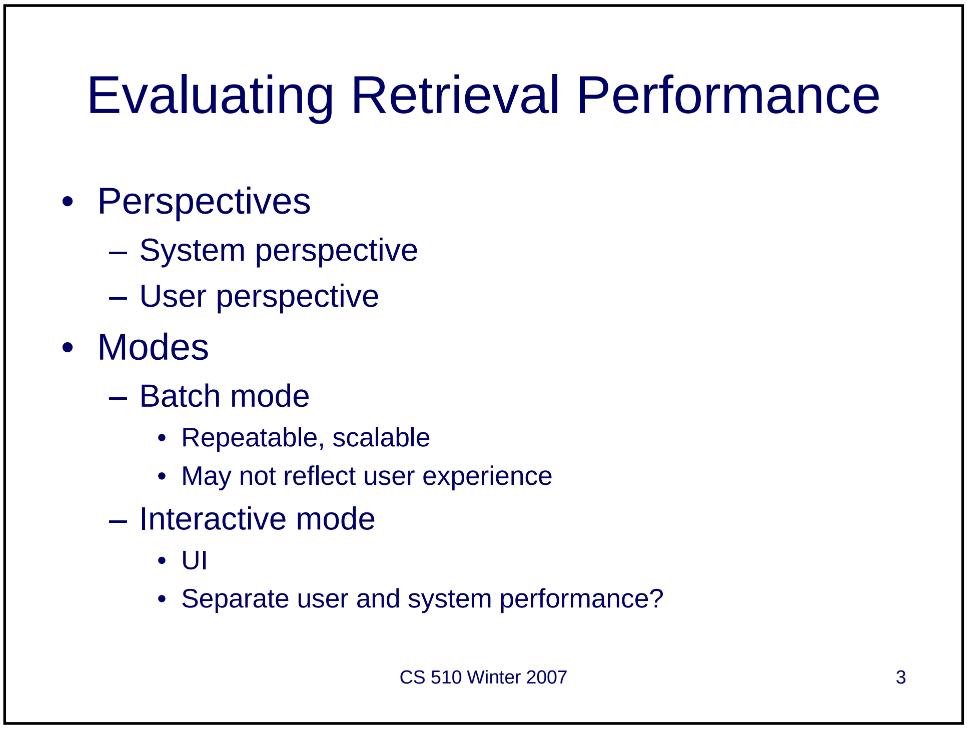
Lecture 8

CS 410/510

Information Retrieval on the Internet

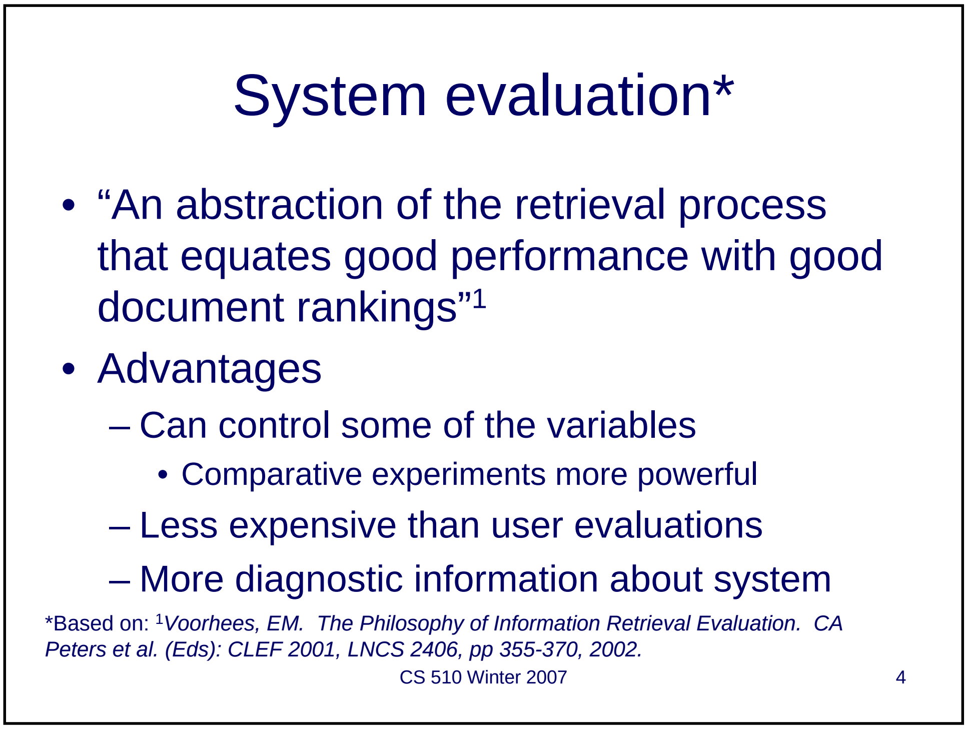
What should we evaluate?

- Time
- Space
- Cost
- Usability
- Retrieval performance



Evaluating Retrieval Performance

- Perspectives
  - System perspective
  - User perspective
- Modes
  - Batch mode
    - Repeatable, scalable
    - May not reflect user experience
  - Interactive mode
    - UI
    - Separate user and system performance?



System evaluation*

- "An abstraction of the retrieval process that equates good performance with good document rankings"[1]
- Advantages
  - Can control some of the variables
    - Comparative experiments more powerful
  - Less expensive than user evaluations
  - More diagnostic information about system

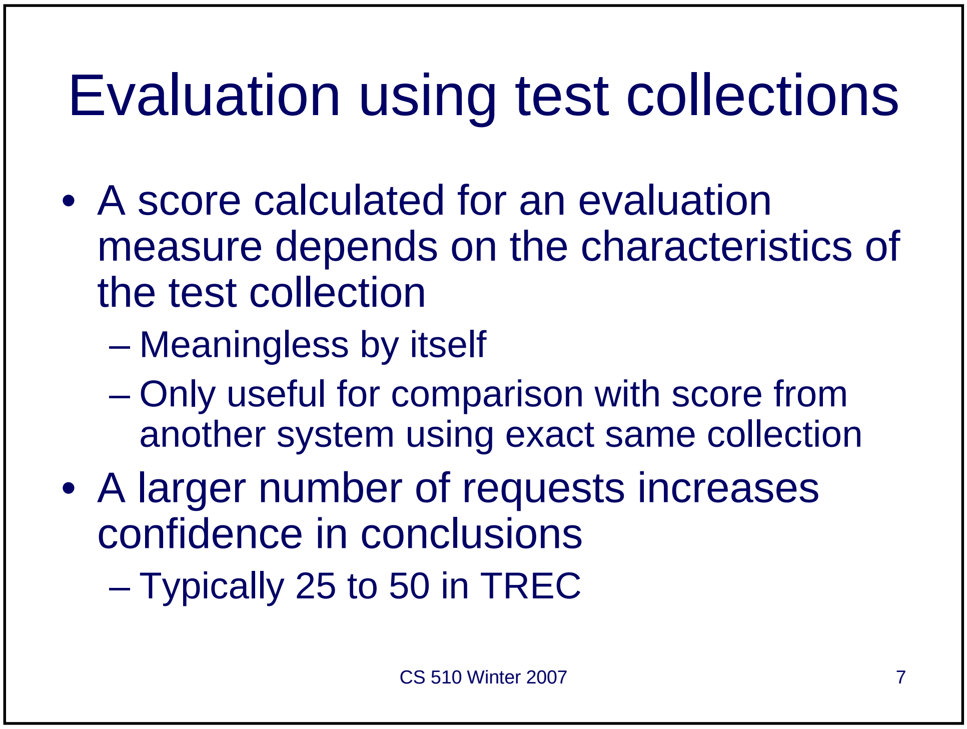
*Based on: [1]*Voorhees, EM. The Philosophy of Information Retrieval Evaluation. CA Peters et al. (Eds): CLEF 2001, LNCS 2406, pp 355-370, 2002.*

Test collections

- Cranfield paradigm
- Components
  - – Documents
  - – Requests
  - – Relevance judgments
- Advantages
  - – Allow comparing performance of retrieval algorithms while controlling other variables
  - – Less expensive than user evaluations

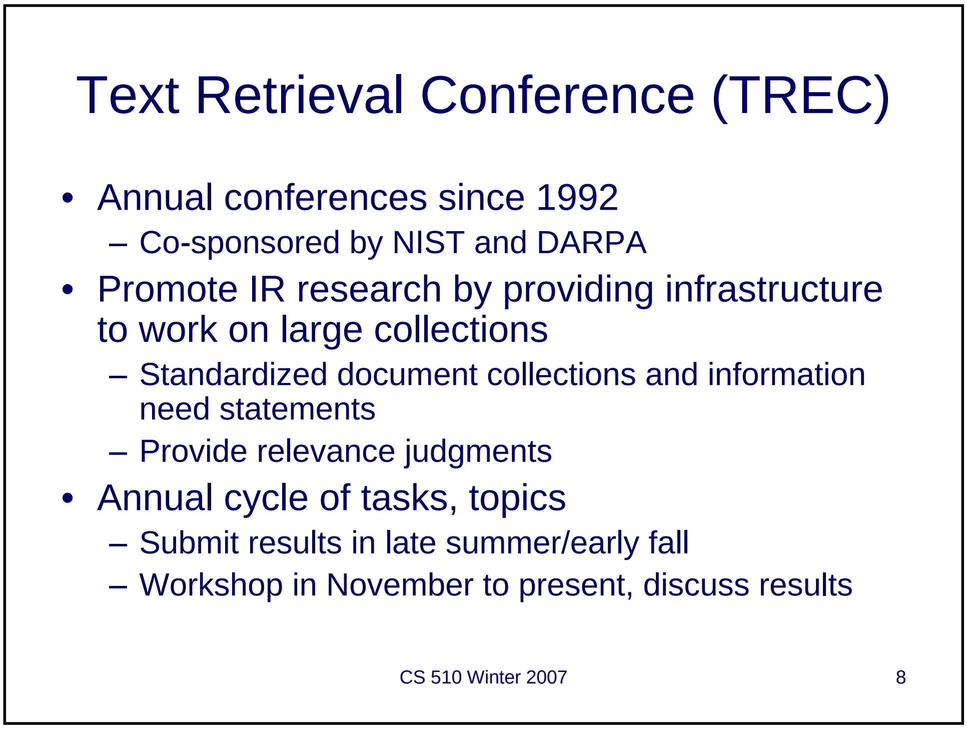
Properties of test collections

- Number of documents
- Kinds of documents
  - – Domain
  - – Format/purpose/language
  - – Full text or not? Indexed?
- Number of requests
  - – Representative of real requests?
- Relevance judgments
  - – Complete? By who? Binary? Using what standard?



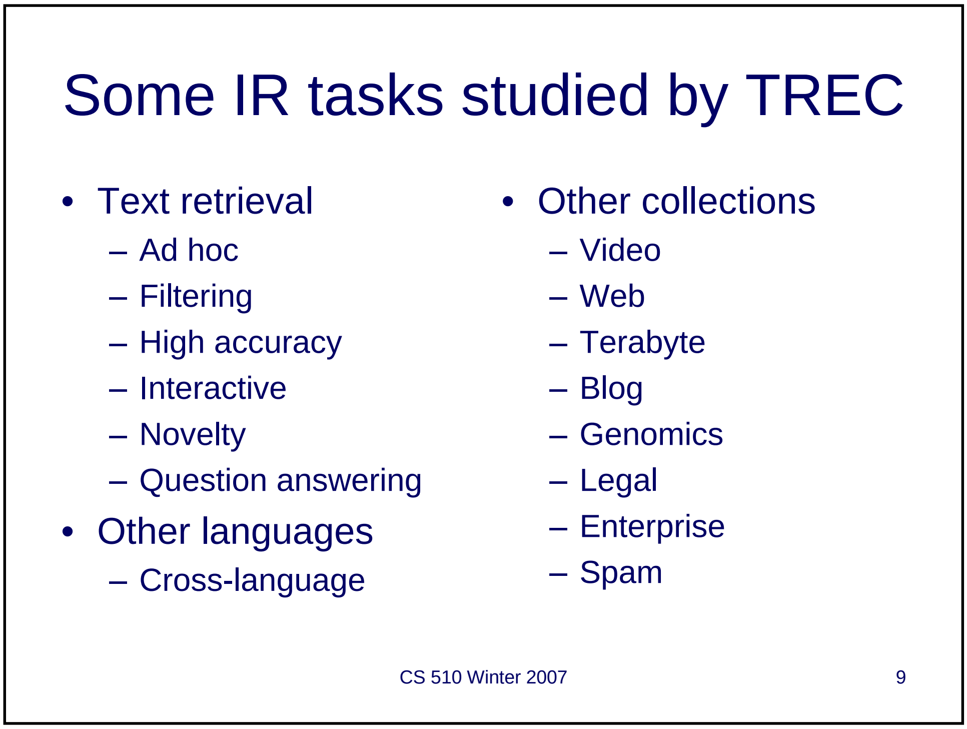
Evaluation using test collections

- A score calculated for an evaluation measure depends on the characteristics of the test collection
  - Meaningless by itself
  - Only useful for comparison with score from another system using exact same collection
- A larger number of requests increases confidence in conclusions
  - Typically 25 to 50 in TREC



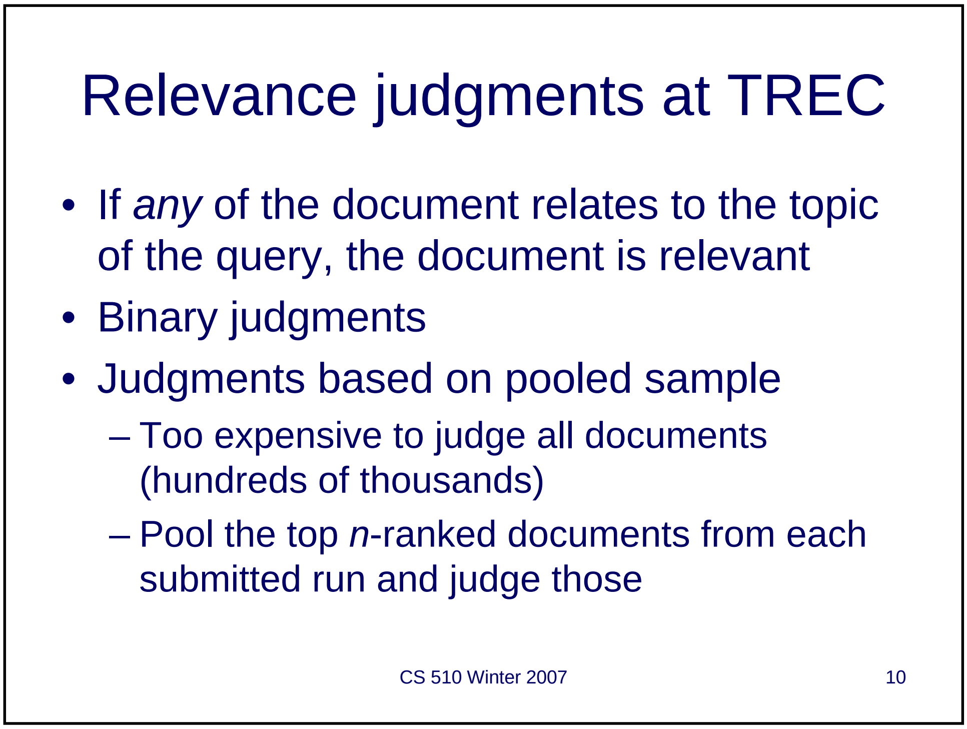
Text Retrieval Conference (TREC)

- Annual conferences since 1992
  - Co-sponsored by NIST and DARPA
- Promote IR research by providing infrastructure to work on large collections
  - Standardized document collections and information need statements
  - Provide relevance judgments
- Annual cycle of tasks, topics
  - Submit results in late summer/early fall
  - Workshop in November to present, discuss results



Some IR tasks studied by TREC

- Text retrieval
  - Ad hoc
  - Filtering
  - High accuracy
  - Interactive
  - Novelty
  - Question answering
- Other languages
  - Cross-language
- Other collections
  - Video
  - Web
  - Terabyte
  - Blog
  - Genomics
  - Legal
  - Enterprise
  - Spam

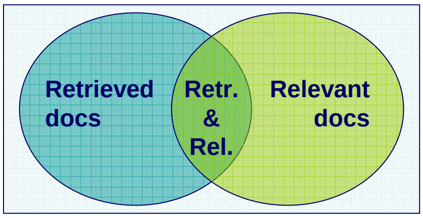


Relevance judgments at TREC

- If *any* of the document relates to the topic of the query, the document is relevant
- Binary judgments
- Judgments based on pooled sample
  - Too expensive to judge all documents (hundreds of thousands)
  - Pool the top *n*-ranked documents from each submitted run and judge those

Evaluation: Metrics

- Two basics:



$$\textbf{Recall} = \frac{\text{\# documents retrieved and relevant}}{\text{\# documents relevant}}$$

$$\textbf{Precision} = \frac{\text{\# documents retrieved and relevant}}{\text{\# documents retrieved}}$$

Evaluation: Metrics

- What about ranked results?
  - Recall and precision fit the Boolean model
  - A relevant document first on a list is more useful than 89th on a list
- Two main approaches
  - Consider precision at various levels of recall
    - Plot precision as a function of recall
  - Summarize performance with a single statistic

Plotting recall and precision

- Typically reported at 11 standard levels of recall
  - 0, 10, 20 ... 100 percent
  - Allows averaging over multiple topics with different numbers of relevant documents
- Interpolate based on actual values
  - For any standard recall level *i*, take maximum precision at any **actual** recall level >= *i*
  - This defines a precision at the standard recall of 0 even though precision at actual recall 0 is undefined

Plotting recall and precision

Relevant docs
0123
0132
0241
0256
0299
0311
0324
0357
0399

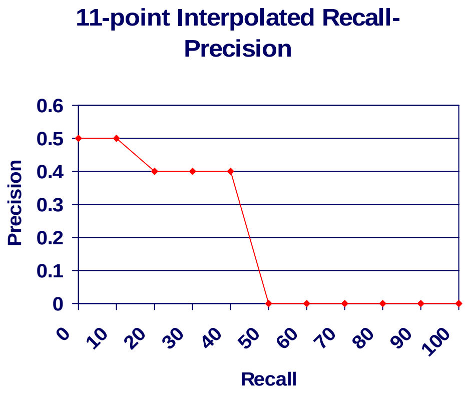
Rank	DocID	Recall	Precision at this recall
1	0234	0	
2	0132	0.111	0.5
3	0115	0.111	
4	0193	0.111	
5	0123	0.222	0.4
6	0345	0.222	
7	0387	0.222	
8	0256	0.333	0.375
9	0078	0.333	
10	0311	0.444	0.4
11	0231	0.444	
12	0177	0.444	

Recall level	Interpolated precision
0	0.5
10	0.5
20	0.4
30	0.4
40	0.4
50	0
60	0
70	0
80	0
90	0
100	0

Plotting recall and precision

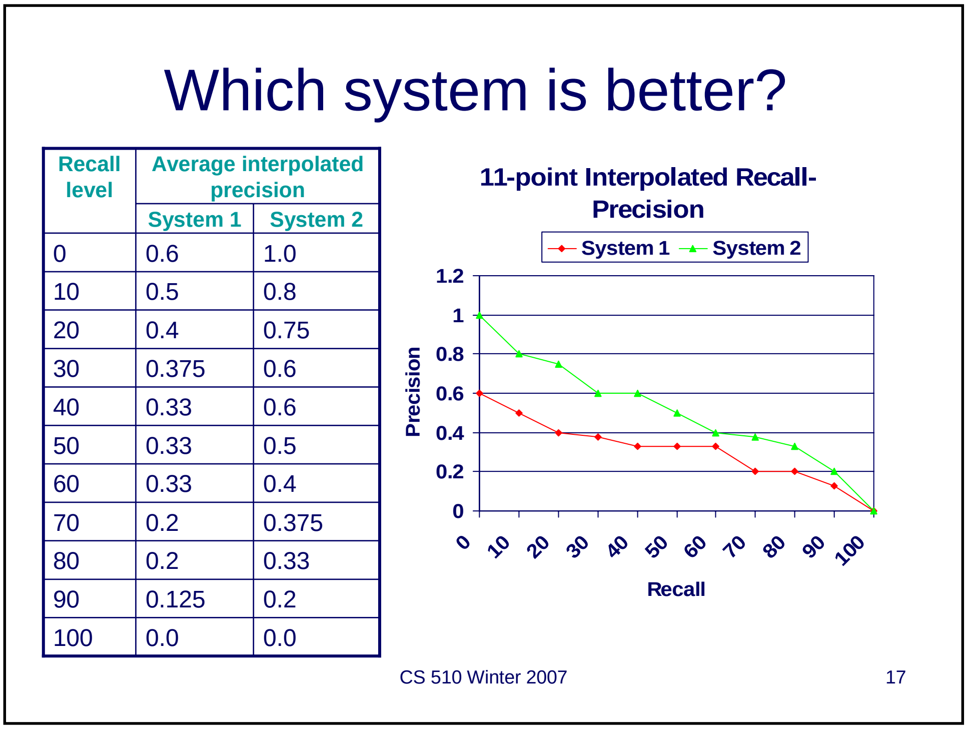
Recall level	Interpolated precision
0	0.5
10	0.5
20	0.4
30	0.4
40	0.4
50	0
60	0
70	0
80	0
90	0
100	0

Recall and precision for a single query



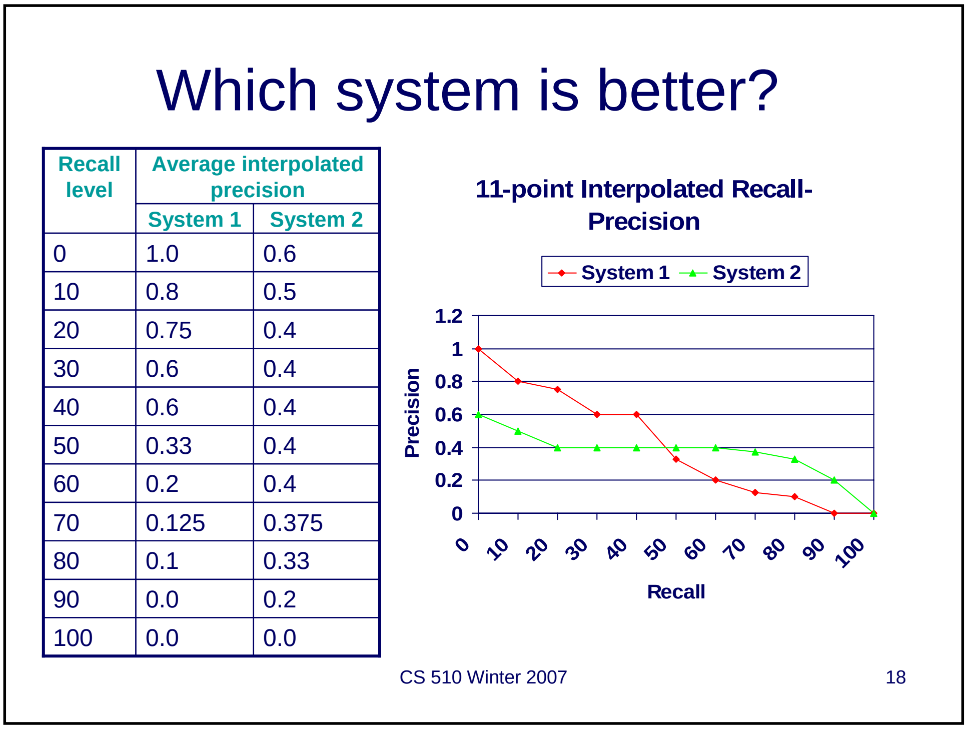
Plotting recall and precision

- Single query performance not necessarily representative of system
  - Compute recall and precision for multiple queries
  - Average the interpolated values at each recall level



Which system is better?

Recall level	Average interpolated precision	
	System 1	System 2
0	0.6	1.0
10	0.5	0.8
20	0.4	0.75
30	0.375	0.6
40	0.33	0.6
50	0.33	0.5
60	0.33	0.4
70	0.2	0.375
80	0.2	0.33
90	0.125	0.2
100	0.0	0.0



Which system is better?

Recall level	Average interpolated precision	
	System 1	System 2
0	1.0	0.6
10	0.8	0.5
20	0.75	0.4
30	0.6	0.4
40	0.6	0.4
50	0.33	0.4
60	0.2	0.4
70	0.125	0.375
80	0.1	0.33
90	0.0	0.2
100	0.0	0.0

Mean average precision (MAP)

- Calculate average precision (AP) for each query
  - Calculate precision at each “seen” relevant doc
    - Not interpolated
    - For each relevant doc not returned, precision = 0
  - Calculate the average for the precisions for each relevant doc $AP = (\sum_{i=1}^{R} \frac{i}{rank_i}) / R$

    where *R* = number of relevant docs for that query and *i/rank*$_i$ = 0 if document *i* was not retrieved
- Calculate the mean of the APs for all the queries

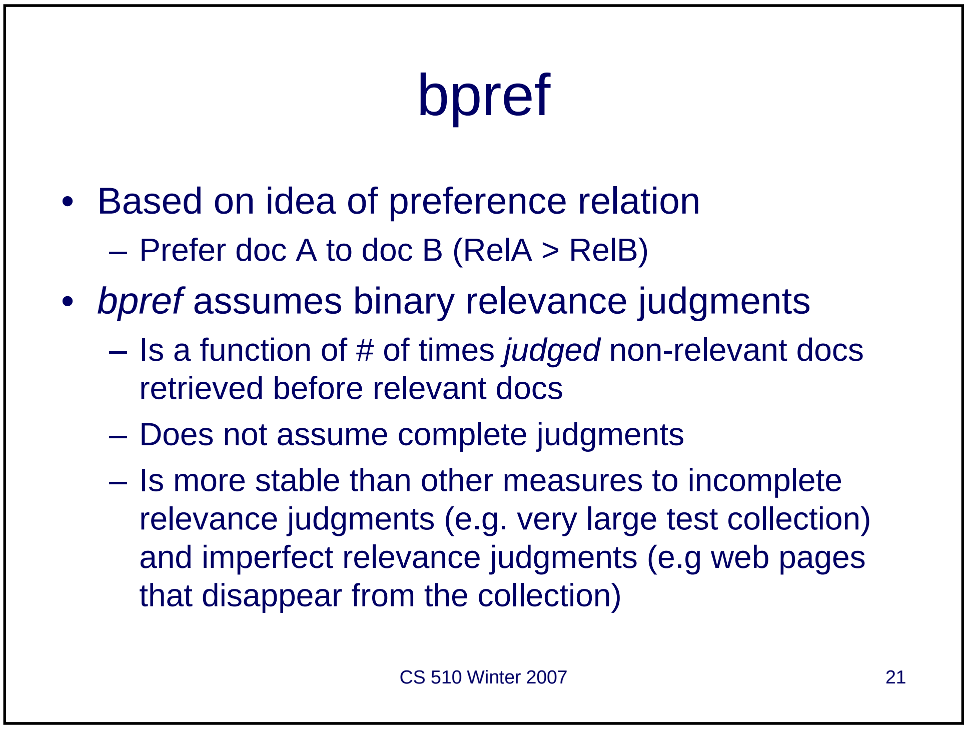
Mean Average Precision

Average precision (AP)

Docs: (9 relevant)	Precision	Docs	Precision
1 Not relevant		6 Not relevant	
2 Relevant	1/2 = 0.5	7 Not relevant	
3 Not relevant		8 Relevant	3/8=0.375
4 Not relevant		9 Not relevant	
5 Relevant	2/5 = 0.4	10 Relevant	4/10=0.4
Not found	0		
AP	(0.5 + 0.4 + 0.375 + 0.4 + 0 + 0 + 0 + 0 + 0) / 9 = 0.1861		

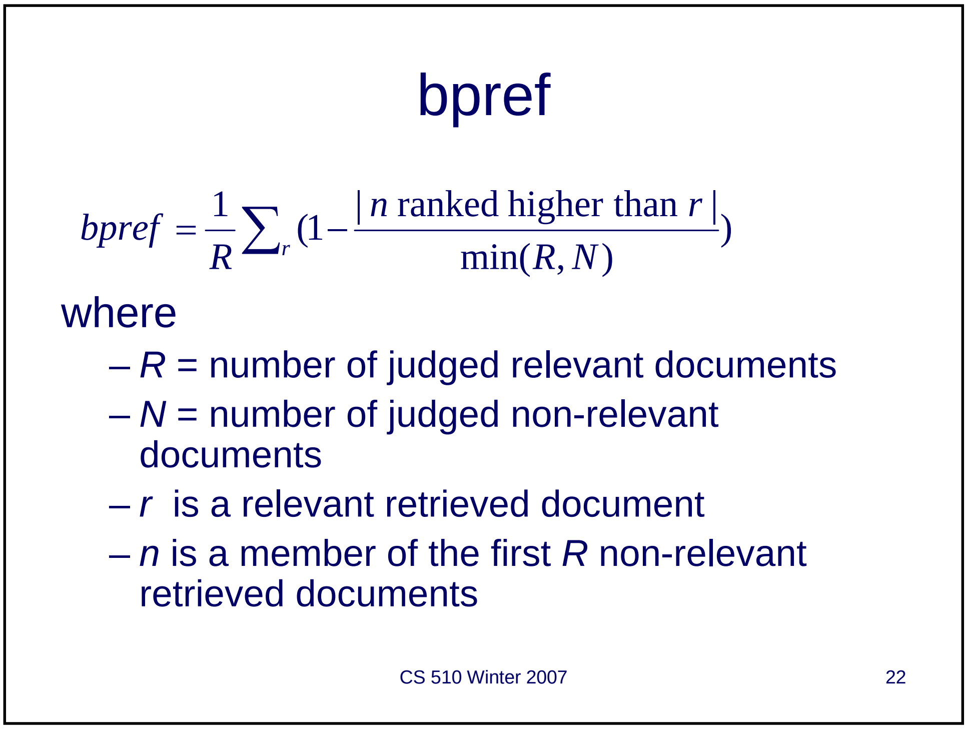
Mean average precision (MAP)

- calculated for a batch of queries
- $MAP = (\sum_{i=1}^{Q} AP_i) / Q$ where *Q* = number of queries in a batch



bpref

- Based on idea of preference relation
  - Prefer doc A to doc B (RelA > RelB)
- *bpref* assumes binary relevance judgments
  - Is a function of # of times *judged* non-relevant docs retrieved before relevant docs
  - Does not assume complete judgments
  - Is more stable than other measures to incomplete relevance judgments (e.g. very large test collection) and imperfect relevance judgments (e.g web pages that disappear from the collection)

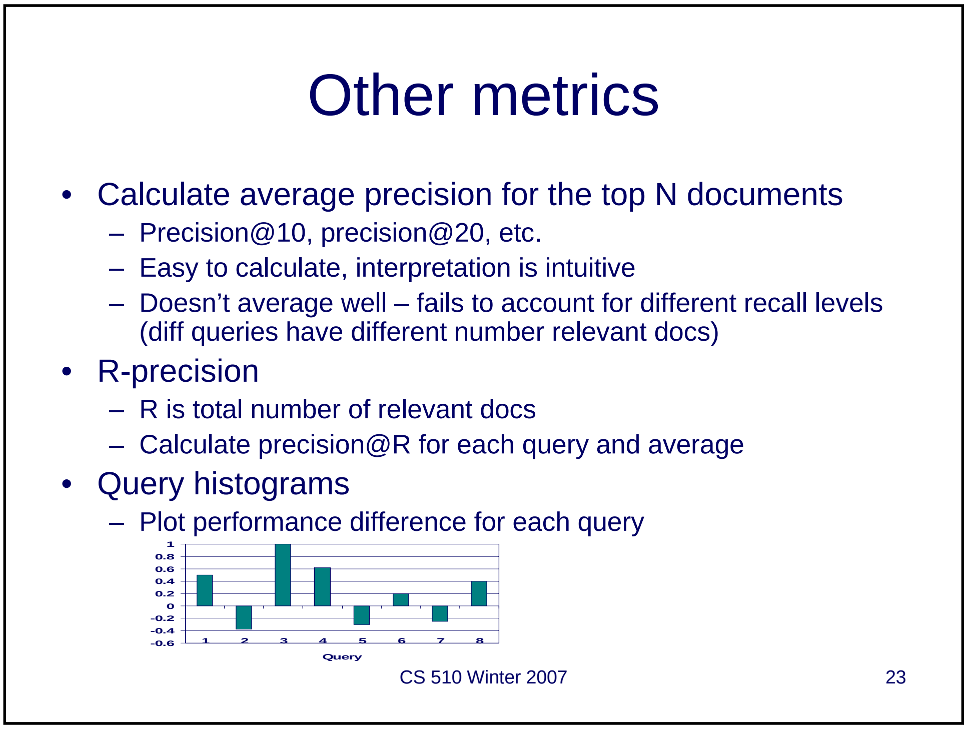


bpref

$$bpref = \frac{1}{R}\sum_r \left(1 - \frac{|n \text{ ranked higher than } r|}{\min(R, N)}\right)$$

where

- R = number of judged relevant documents
- N = number of judged non-relevant documents
- r is a relevant retrieved document
- n is a member of the first R non-relevant retrieved documents



Other metrics

- Calculate average precision for the top N documents
  - Precision@10, precision@20, etc.
  - Easy to calculate, interpretation is intuitive
  - Doesn’t average well – fails to account for different recall levels (diff queries have different number relevant docs)
- R-precision
  - R is total number of relevant docs
  - Calculate precision@R for each query and average
- Query histograms
  - Plot performance difference for each query