

Evaluation

Rong Jin

Evaluation

- Evaluation is key to building *effective* and *efficient* search engines
 - usually carried out in controlled experiments
 - *online* testing can also be done

- Effectiveness and efficiency are related
 - High efficiency may be obtained at the price of effectiveness



Evaluation Corpus

- *Test collections* consisting of documents, queries, and relevance judgments, e.g.,
 - CACM: Titles and abstracts from the Communications of the ACM from 1958-1979. Queries and relevance judgments generated by computer scientists.
 - AP: Associated Press newswire documents from 1988-1990 (from TREC disks 1-3). Queries are the title fields from TREC topics 51-150. Topics and relevance judgments generated by government information analysts.
 - GOV2: Web pages crawled from websites in the .gov domain during early 2004. Queries are the title fields from TREC topics 701-850. Topics and relevance judgments generated by government analysts.

Test Collections

Collection	Number of documents	Size	Average number of words/doc.
CACM	3,204	2.2 Mb	64
AP	242,918	0.7 Gb	474
GOV2	25,205,179	426 Gb	1073

Collection	Number of queries	Average number of words/query	Average number of relevant docs/query
CACM	64	13.0	16
AP	100	4.3	220
GOV2	150	3.1	180



TREC Topic Example

<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>

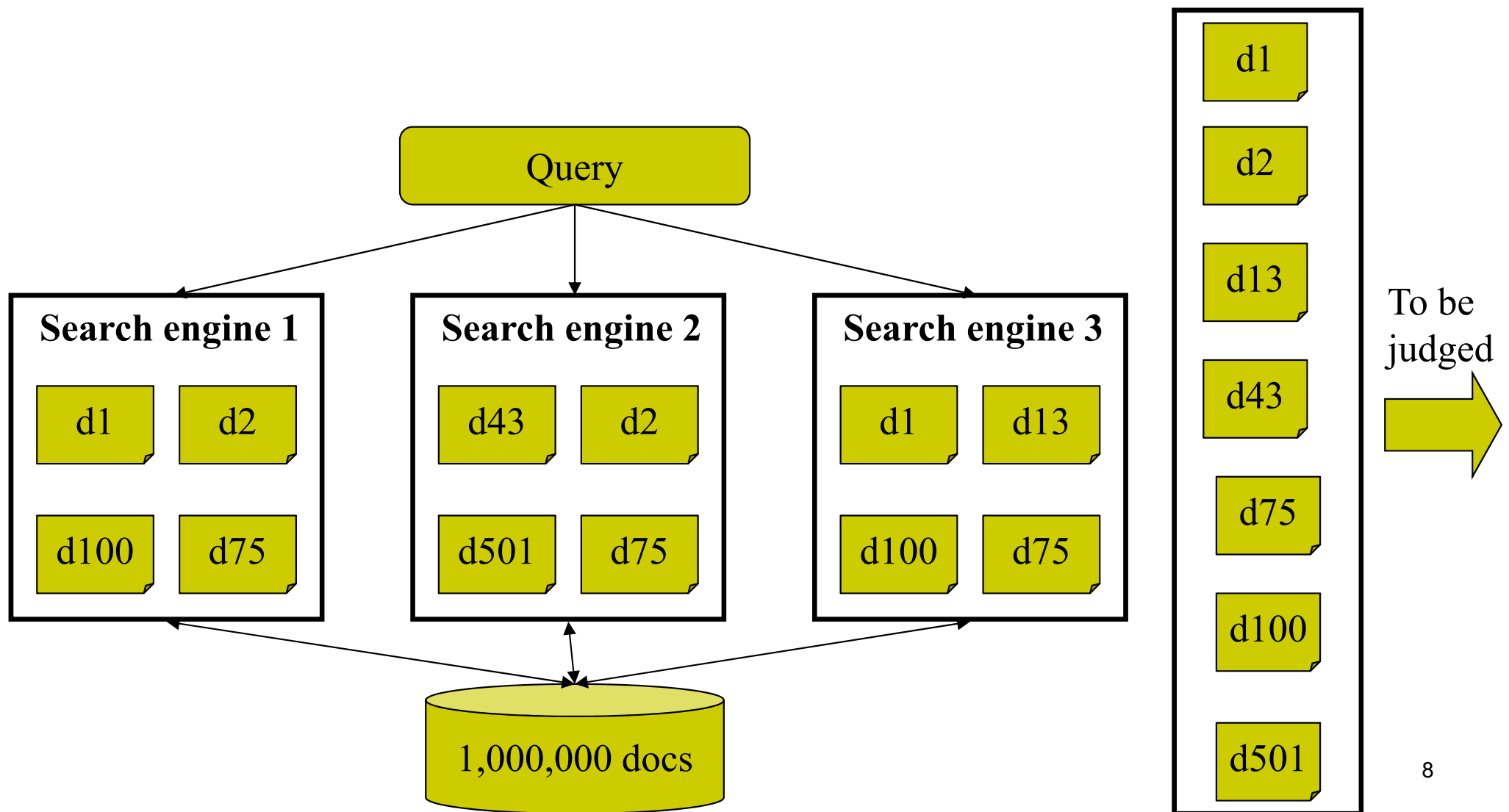
Relevance Judgments

- Obtaining relevance judgments is an expensive, time-consuming process
 - who does it?
 - what are the instructions?
 - what is the level of agreement?
- TREC judgments
 - depend on task being evaluated
 - generally binary
 - reasonable agreement because of “narrative”

Pooling

- Exhaustive judgments for all documents in a collection is not practical
- Pooling technique is used in TREC
 - top *k results* (*k varied between 50 and 200*) from the rankings obtained by different search engines are merged into a pool
 - duplicates are removed
 - documents are presented in some random order to the relevance judges
- Produces a large number of relevance judgments for each query, although still incomplete

Pooling



Bias in Relevance Judgments

- Relevance judgment is subjective
- Disagreement among assessors

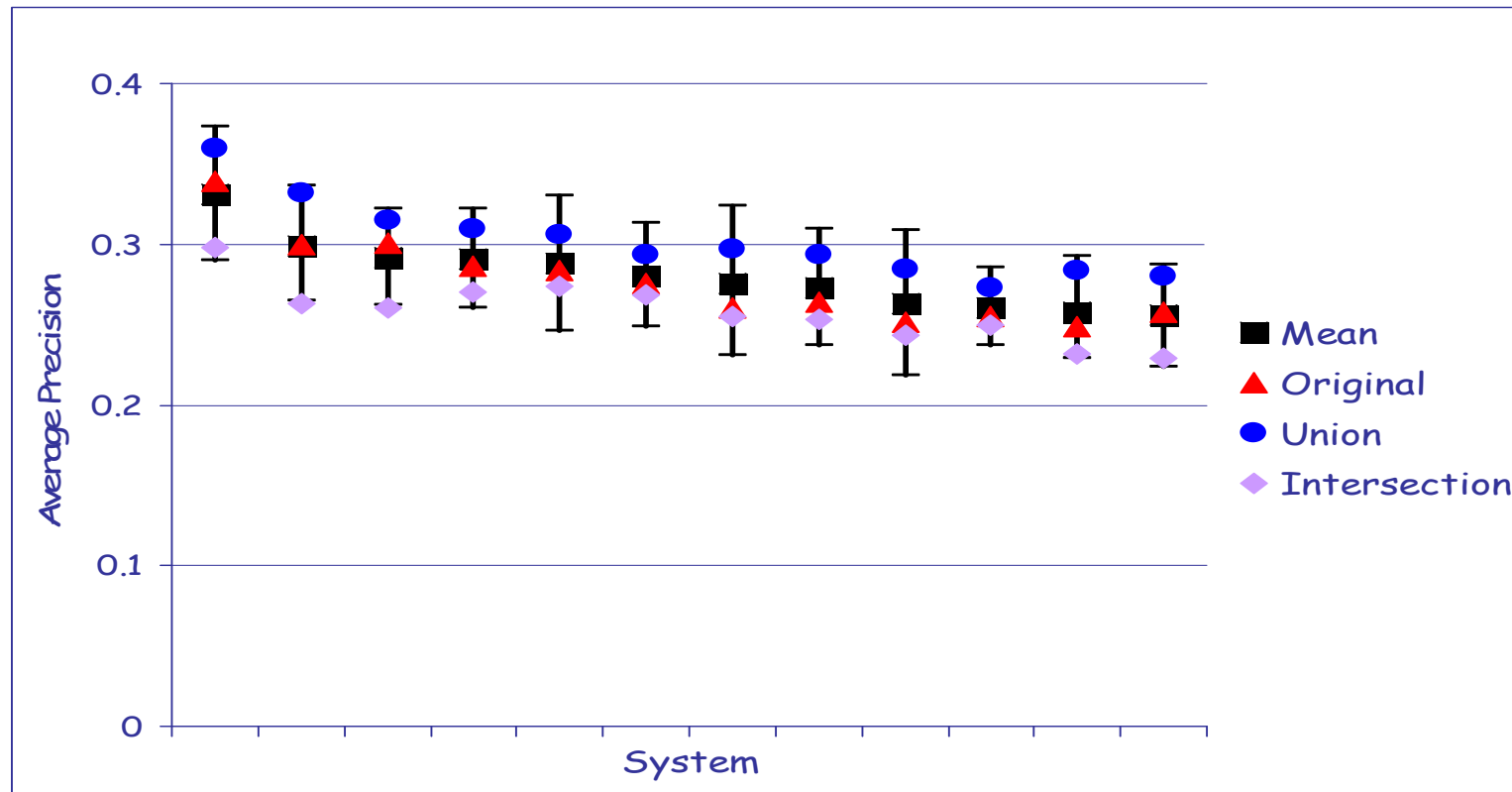
information need	number of docs judged	disagreements	NR	R
51	211	6	4	2
62	400	157	149	8
67	400	68	37	31
95	400	110	108	2
127	400	106	12	94



Combine Multiple Judgments

- Judges disagree a lot. How to combine judgments from multiple reviewers ?
 - Union
 - Intersection
 - Majority vote

Combine Multiple Judgments



- Large impact on absolute performance numbers
- Virtually no impact on ranking of systems

Query Logs

- Used for tuning and evaluating search engines
 - also for techniques such as query suggestion and spell checking
- Typical contents
 - User identifier or user session identifier
 - Query terms - stored exactly as user entered
 - List of URLs of results, their ranks on the result list, and whether they were clicked on
 - Timestamp(s) - records the time of user events such as query submission, clicks

Query Logs

- Clicks are not relevance judgments
 - although they are correlated
 - biased by a number of factors such as rank on result list
- Can use clickthrough data to predict *preferences* between pairs of documents
 - appropriate for tasks with multiple levels of relevance, focused on user relevance
 - various “policies” used to generate preferences

Example Click Policy

- *Skip Above and Skip Next*

- click data

d_1

d_2

d_3 (clicked)

d_4

- generated preferences

$d_3 > d_2$

$d_3 > d_1$

$d_3 > d_4$

Query Logs

- Click data can also be aggregated to remove noise
- *Click distribution* information
 - can be used to identify clicks that have a higher frequency than would be expected
 - high correlation with relevance
 - e.g., using *click deviation* to filter clicks for preference-generation policies

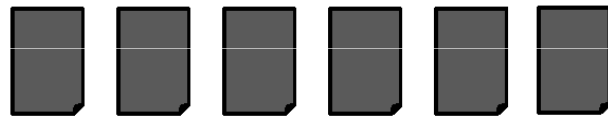
Evaluation Metrics: Classification View

Action Doc	Retrieved	Not Retrieved
Relevant	Relevant Retrieved	Relevant Rejected
Not relevant	Irrelevant Retrieved	Irrelevant Rejected

$$\text{Precision} = \frac{\text{Relevant Retrieved}}{\text{Retrieved}}$$

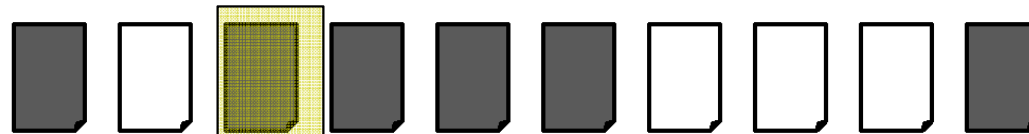
$$\text{Recall} = \frac{\text{Relevant Retrieved}}{\text{Relevant}}$$

Evaluation Metrics: Example



= 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

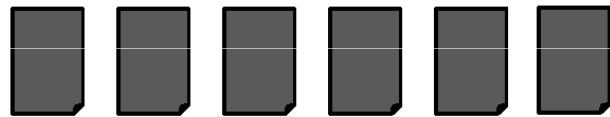
$$\text{Recall} = 2/6 = 0.33$$

$$\text{Precision} = 2/3 = 0.67$$

$$\text{Precision} = \frac{\text{Relevant Retrieved}}{\text{Retrieved}}$$

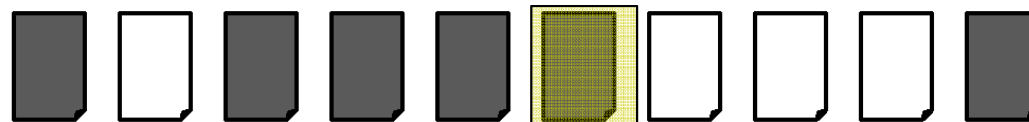
$$\text{Recall} = \frac{\text{Relevant Retrieved}}{\text{Relevant}}$$

Evaluation Metrics: Example



= 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

$$\text{Recall} = 5/6 = 0.83$$

$$\text{Precision} = 5/6 = 0.83$$

$$\text{Precision} = \frac{\text{Relevant Retrieved}}{\text{Retrieved}}$$

$$\text{Recall} = \frac{\text{Relevant Retrieved}}{\text{Relevant}}$$

F Measure

- *Harmonic mean* of recall and precision

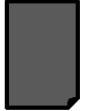

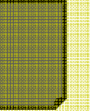



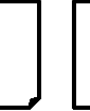
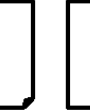


$$F = \frac{1}{\frac{1}{2} \left(\frac{1}{R} + \frac{1}{P} \right)} = \frac{2RP}{(R+P)}$$

- Why harmonic mean?
- harmonic mean emphasizes the importance of small values, whereas the arithmetic mean is affected more by outliers that are unusually large

Evaluation Metrics: Example

 = 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

$$\text{Recall} = 2/6 = 0.33$$

$$\text{Precision} = 2/3 = 0.67$$








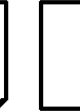
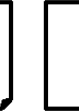

$$F = 2 * \text{Recall} * \text{Precision} / (\text{Recall} + \text{Precision})$$

$$= 2 * 0.33 * 0.67 / (0.33 + 0.67) = 0.22$$

Evaluation Metrics: Example

 = 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

$$\text{Recall} = 5/6 = 0.83$$

$$\text{Precision} = 5/6 = 0.83$$

$$F = 2 * \text{Recall} * \text{Precision} / (\text{Recall} + \text{Precision})$$

$$= 2 * 0.83 * 0.83 / (0.83 + 0.83) = 0.83$$













Evaluation for Ranking

- Average precision
 - Averaging the precision values from the rank positions where a relevant document was retrieved
 - Set precision values to be zero for the not retrieved documents











Average Precision: Example

 = the relevant documents

Ranking #1

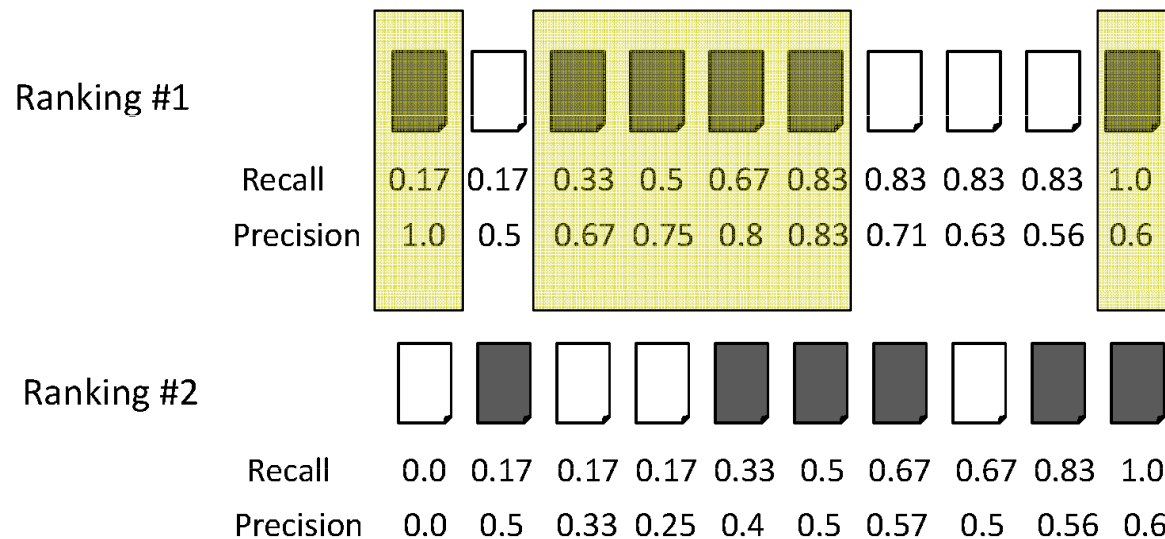
										
Recall	0.17	0.17	0.33	0.5	0.67	0.83	0.83	0.83	0.83	1.0
Precision	1.0	0.5	0.67	0.75	0.8	0.83	0.71	0.63	0.56	0.6

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

Average Precision: Example

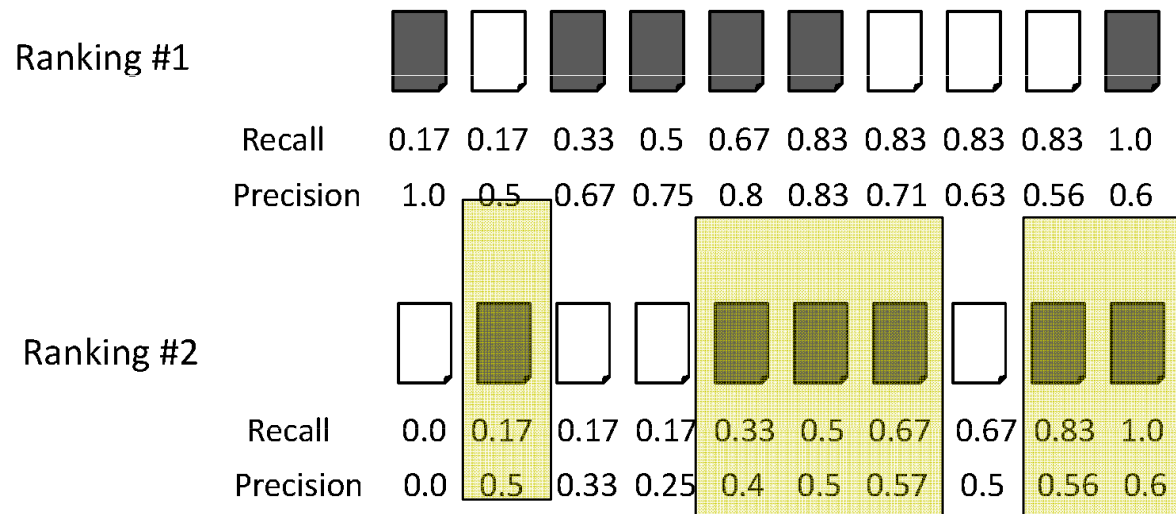
 = the relevant documents



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

Average Precision: Example

 = the relevant documents










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

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








Average Precision: Example

 = the relevant documents

Ranking #1

							
Recall	0.17	0.17	0.33	0.5	0.67	0.83	0.83
Precision	1.0	0.5	0.67	0.75	0.8	0.83	0.71

Ranking #2

							
Recall	0.0	0.17	0.17	0.17	0.33	0.5	0.67
Precision	0.0	0.5	0.33	0.25	0.4	0.5	0.57

Miss one relevant document








$$\text{Rank 1} = (1 + 0.67 + 0.75 + 0.8 + 0.83 + 0) / 6 = 0.675$$

$$\text{Rank 2} = (0.5 + 0.4 + 0.5 + 0.57 + 0 + 0) / 6 = 0.328$$








Average Precision: Example

 = the relevant documents

Ranking #1

							
Recall	0.17	0.17	0.33	0.5	0.67	0.83	0.83
Precision	1.0	0.5	0.67	0.75	0.8	0.83	0.71

Ranking #2


							
Recall	0.0	0.17	0.17	0.17	0.33	0.5	0.67
Precision	0.0	0.5	0.33	0.25	0.4	0.5	0.57

Miss two relevant documents











$$\text{Rank 1} = (1 + 0.67 + 0.75 + 0.8 + 0.83 + 0) / 6 = 0.675$$


$$\text{Rank 2} = (0.5 + 0.4 + 0.5 + 0.57 + 0 + 0) / 6 = 0.328$$

Mean Average Precision (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

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

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

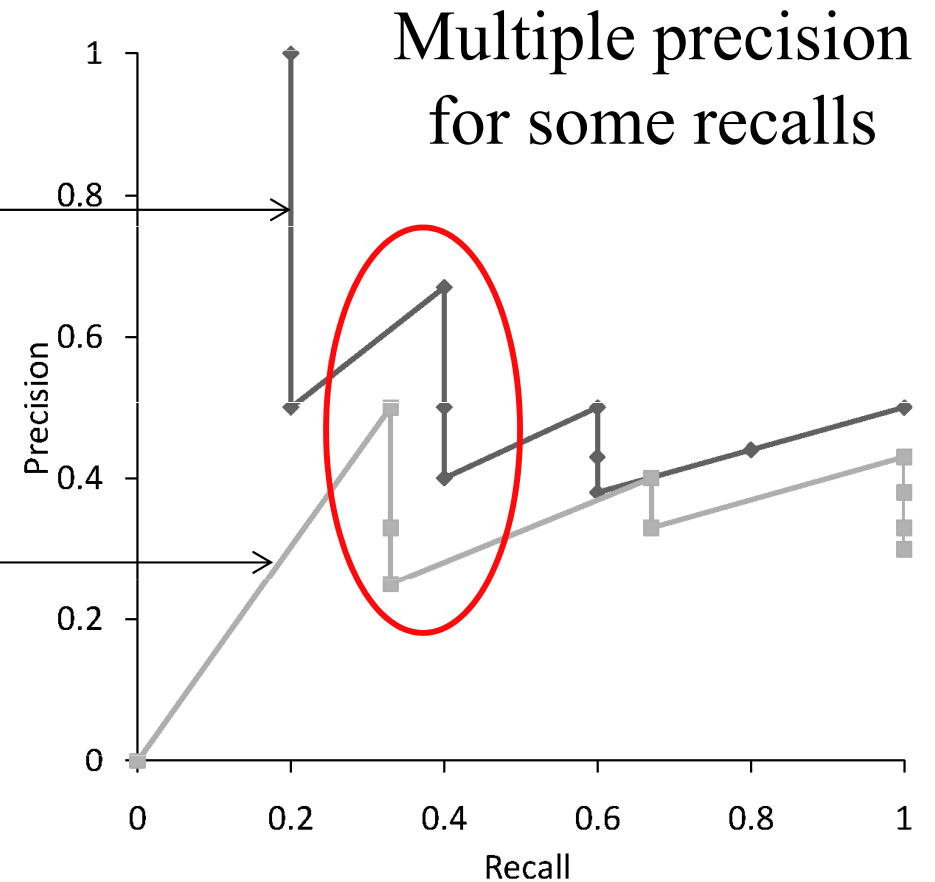
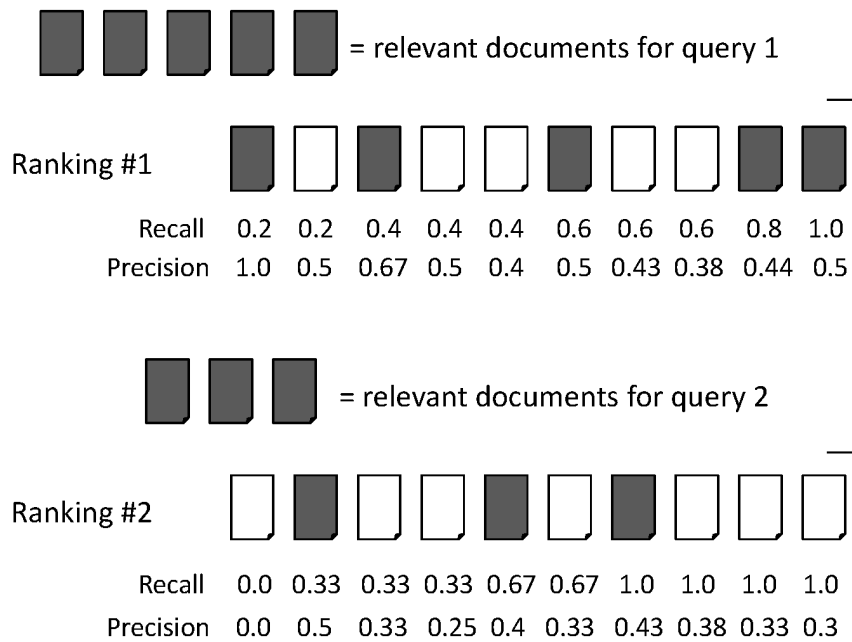
mean average precision = $(0.62 + 0.44)/2 = 0.53$



Mean Average Precision (MAP)

- Summarize rankings from multiple queries by averaging average precision
- Most commonly used measure in research papers
- Assumes user is interested in finding **many** relevant documents for each query
- Requires **many** relevance judgments in text collection

Recall-Precision Graph



Interpolation

$$P(R) = \max\{P' : R' \geq R \wedge (R', P') \in S\}$$

- where S is the set of observed (R, P) points
- Defines precision at any recall level as the *maximum* precision observed in any recall-precision point at a higher recall level
 - produces a step function
 - defines precision at recall 0.0

Interpolation

 = 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







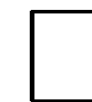



Precision 1.0 0.5 0.67 0.5 0.4 0.5 0.43 0.38 0.44 0.5

Recall 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

Interpolated
Precision 1.0

Interpolation











 = 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
Recall	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Interpolated Precision	1.0										

Interpolation

 = 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

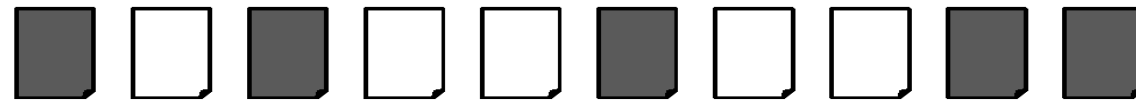
Recall 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

Interpolated
Precision 1.0 1.0

Interpolation

 = 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











Recall 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

Interpolated
Precision 1.0 1.0 1.0

Interpolation

 = 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

Recall 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

Interpolated
Precision 1.0 1.0 1.0

Interpolation

 = 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











Recall 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

Interpolated
Precision 1.0 1.0 1.0 0.67

Interpolation

 = relevant documents for query 1

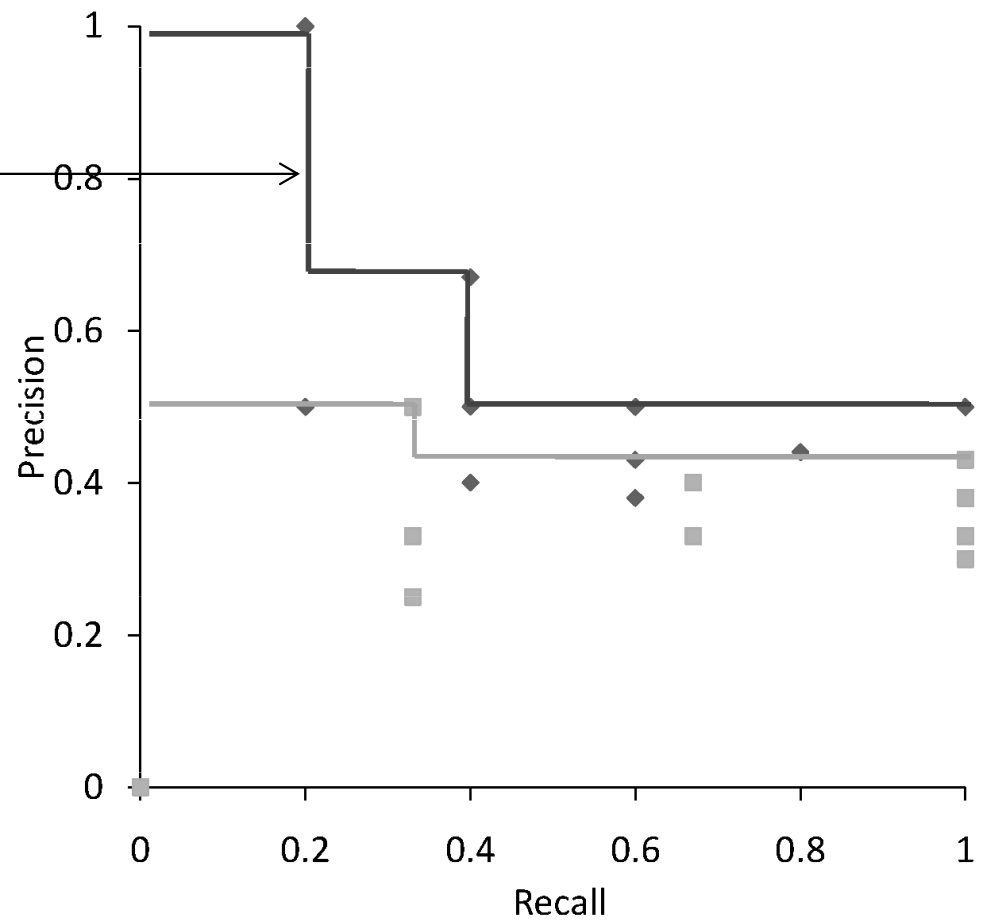
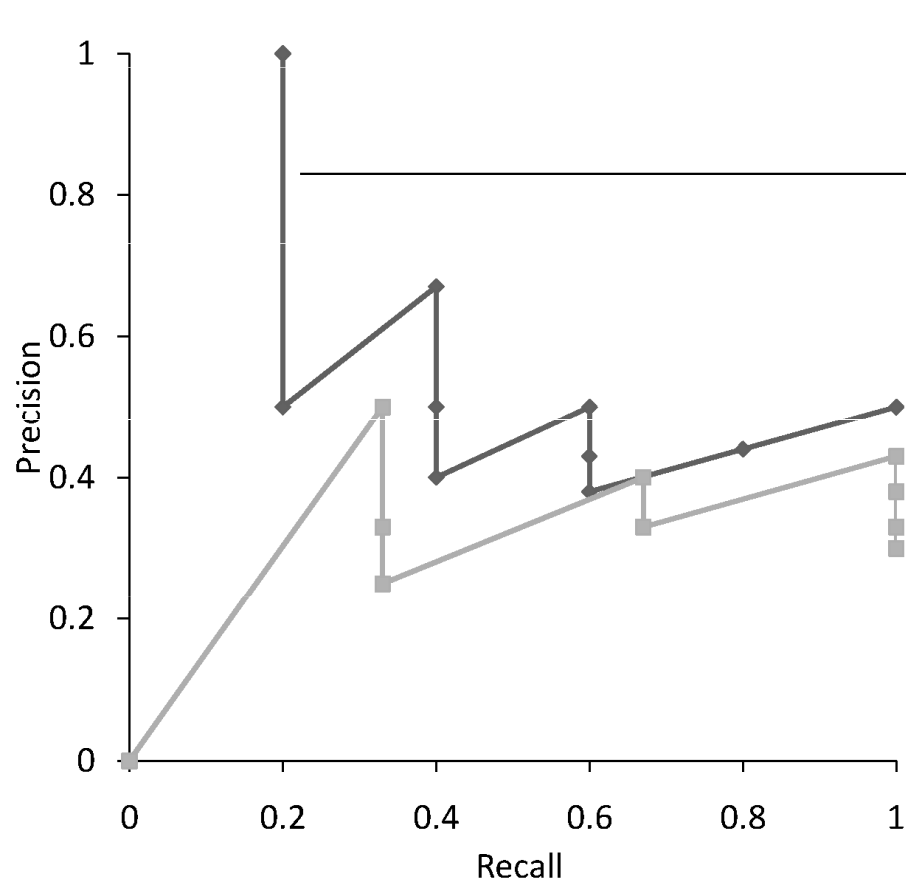
Ranking #1

										
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

Recall 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

Interpolated
Precision 1.0 1.0 1.0 0.67 0.67 0.5 0.5 0.5 0.5 0.5 0.5

Interpolation

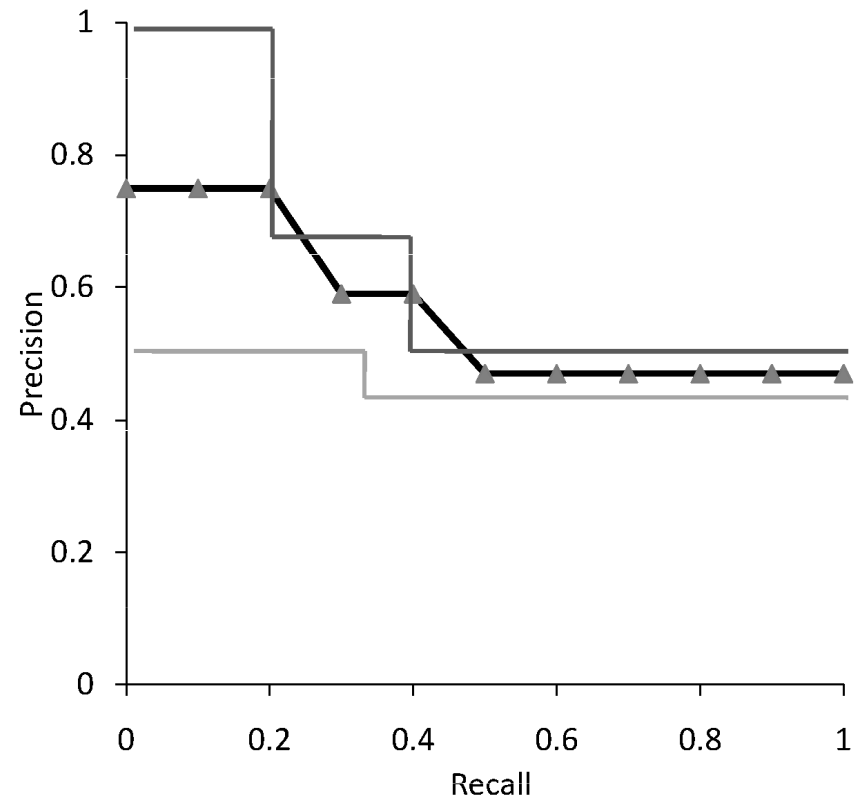
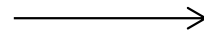
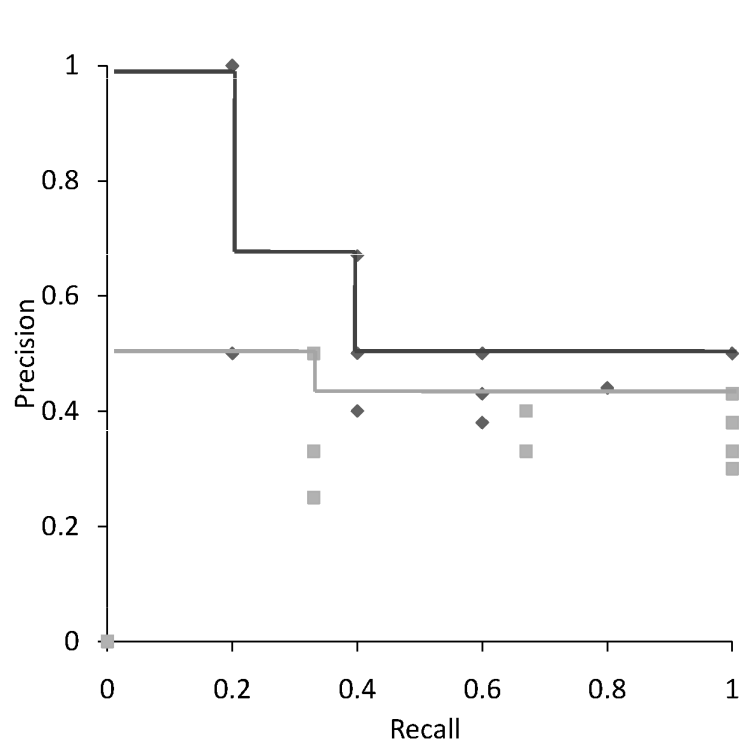


Average Precision at Standard Recall Levels

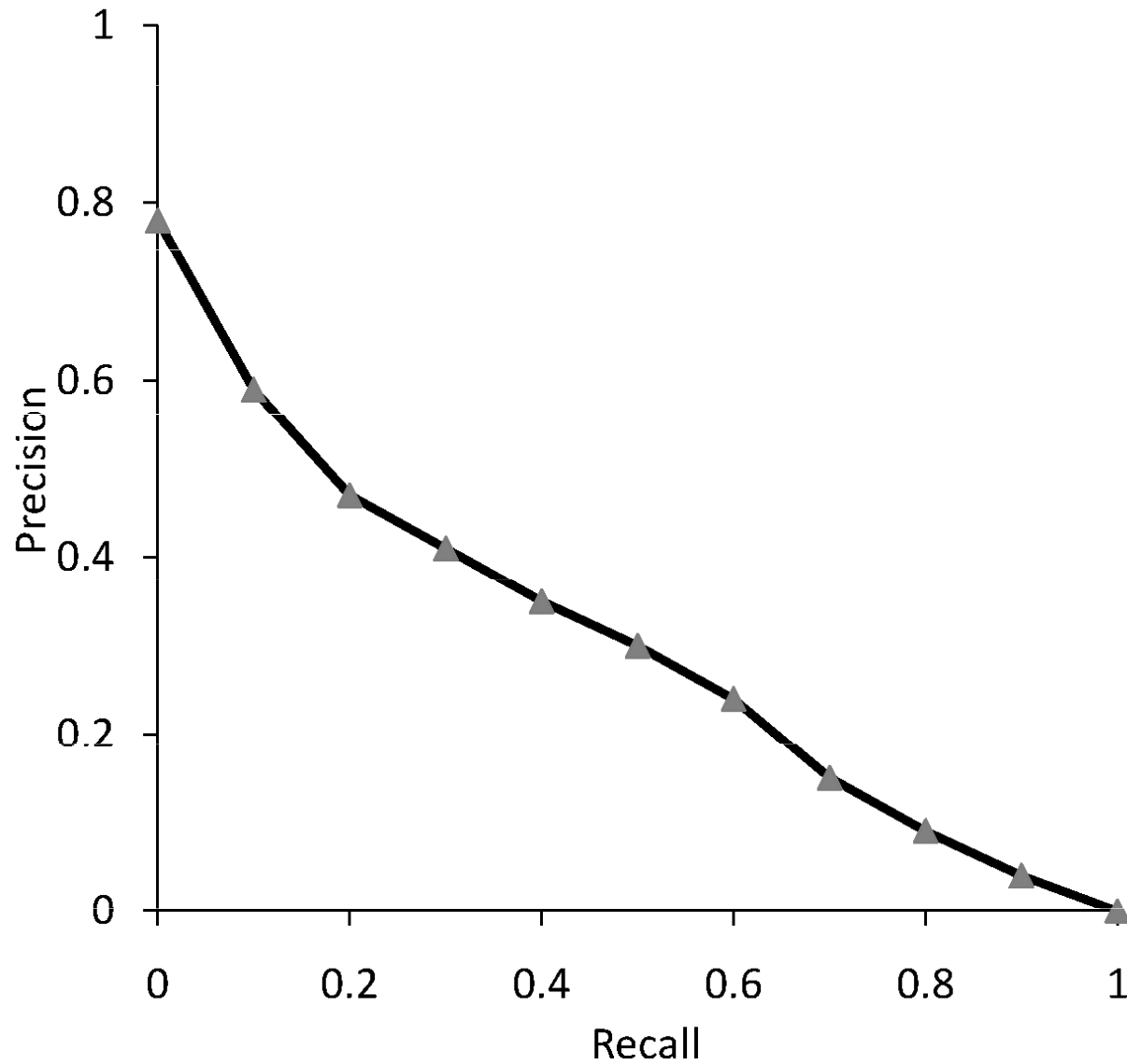
Recall	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Ranking 1	1.0	1.0	1.0	0.67	0.67	0.5	0.5	0.5	0.5	0.5	0.5
Ranking 2	0.5	0.5	0.5	0.5	0.43	0.43	0.43	0.43	0.43	0.43	0.43
Average	0.75	0.75	0.75	0.59	0.47	0.47	0.47	0.47	0.47	0.47	0.47

- Only consider standard recall levels: varying from 0.0 to 1.0 at the incremental of 0.1
- Recall-precision graph plotted by simply joining the average precision points at the standard recall levels

Average Recall-Precision Graph



Graph for 50 Queries






Focusing on Top Documents

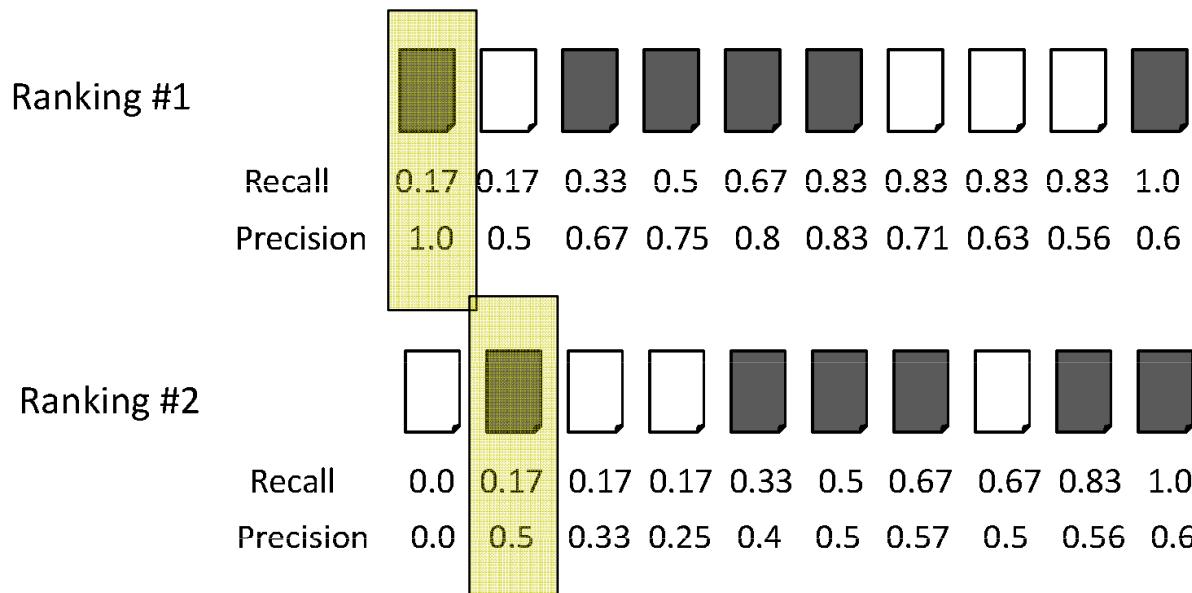
- Users tend to look at only the top part of the ranked result list to find relevant documents
- Some search tasks have only one relevant document
 - e.g., navigational search, question answering
- Recall not appropriate
 - instead need to measure how well the search engine does at retrieving relevant documents at very high ranks

Focusing on Top Documents

- Precision at Rank R
 - R typically 5, 10, 20
 - easy to compute, average, understand
 - not sensitive to rank positions less than R
- Reciprocal Rank
 - reciprocal of the rank at which the *first* relevant document is retrieved
 - *Mean Reciprocal Rank (MRR)* is the average of the reciprocal ranks over a set of queries
 - very sensitive to rank position

MRR

 = the relevant documents



$$RR = 1/1 = 1$$

$$RR = 1/2 = 0.5$$

$$MRR = (1+0.5)/2 = 0.75$$

Discounted Cumulative Gain (DCG)

- Popular measure for evaluating web search and related tasks
 - Use graded relevance
- 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

- Gain is accumulated starting at the top of the ranking and is *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 $1/3$

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

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

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

Efficiency Metrics

- Query throughput
 - The number of queries processed per second
- Query latency
 - The time between issuing a query and receiving a response, measured in millisecond
 - Users consider instantaneous if the latency is less than 150 millisecond
- Relation between query throughput and latency
 - High throughput → handle multiple queries simultaneously → high latency

Significance Tests

- Given the results from a number of queries, how can we conclude that ranking algorithm A is better than algorithm B?
- A significance test
 - *null hypothesis*: no difference between A and B
 - *alternative hypothesis*: B is better than A
 - the *power* of a test is the probability that the test will reject the null hypothesis correctly
 - increasing the number of queries in the experiment also increases power of test

Example Experimental Results

Query	A	B	B-A
1	25	35	10
2	43	84	41
3	39	15	-24
4	75	75	0
5	43	68	25
6	15	85	70
7	20	80	60
8	52	50	-2
9	49	58	9
10	50	75	25

Significance level: $\alpha = 0.05$

Probability for B=A

Example Experimental Results

Query	A	B	B-A
1	25	35	10
2	43	84	41
3	39	15	-24
4	75	75	0
5	43	68	25
6	15	85	70
7	20	80	60
8	52	50	-2
9	49	58	9
10	50	75	25
Avg	41.1	62.5	

$$\text{t-test} \quad t = \frac{\overline{B-A}}{\sigma_{B-A}} \cdot \sqrt{N}$$

$$t = 2.33 \quad \text{p-value} = 0.02$$

Probability for B=A is 0.02

→ B is better than A

Significance level: $\alpha = 0.05$

Probability for B=A

Online Testing

- Test (or even train) using live traffic on a search engine
- Benefits:
 - real users, less biased, large amounts of test data
- Drawbacks:
 - noisy data, can degrade user experience
- Often done on small proportion (1-5%) of live traffic

Summary

- No single measure is the correct one for any application
 - choose measures appropriate for task
 - use a combination
 - shows different aspects of the system effectiveness
- Use significance tests (t-test)
- Analyze performance of individual queries