Information Retrieval 4

System-oriented evaluation (*batch-mode* evaluation)

Credits

- Raymond Mooney
- Dik Lee
- Joydeep Ghosh
- Francesco Ricci
- Ricardo Baeza-Yates
- Berthier Riberiro-Neto
- Marco de Gemmis
- Pasquale Lops
- Giovanni Semeraro

Why System Evaluation?

- There are many retrieval models/ algorithms/ systems, which one is the best?
- What is the best component for:
  - Ranking function (dot-product, cosine, …)
  - Term selection (stopword removal, stemming…)
  - Term weighting (TF, TF-IDF,…)
- How far down the ranked list will a user need to look to find some/all relevant documents?

Difficulties in Evaluating IR Systems

- Effectiveness is related to the relevance of retrieved items
- Relevance is not typically binary but continuous
- Even if relevancy is binary, it can be a difficult judgment to make
- Relevance, from a human standpoint, is:
  - Subjective: depends upon a specific user's judgment
  - Situational: relates to user's current needs
  - Cognitive: depends on human perception and behavior
  - Dynamic: changes over time
  - Contextual: influenced by user state (i.e. variables such as mood, group/alone, available time,…)

The Cranfield Paradigm

- Evaluation of IR systems is the result of early experimentation initiated at Cranfield College of Aeronautics between 1958 and 1966 by librarian Cyril Cleverdon (Cleverdon, 1960)
- Insights derived from these experiments provide a foundation for the evaluation of IR systems
- Back in 1952, Cleverdon took notice of a new indexing system called Uniterm, proposed by Mortimer Taube
  - ✓ analysis of 40,000 subject headings, which resulted in 7,000 distinct words
  - ✓ Cleverdon thought it appealing and with Bob Thorne, a colleague, did a small test
  - ✓ he manually indexed 200 documents using Uniterm and asked Thorne to run some queries
  - ✓ this experiment put Cleverdon on a life trajectory of reliance on experimentation for evaluating indexing systems

The Cranfield Paradigm

- Cleverdon obtained a grant from the National Science Foundation (NSF) to compare distinct indexing systems
- these experiments provided <u>interesting insights</u>, that culminated in the modern metrics of Precision and Recall
  - ✓ Precision ratio: the fraction of retrieved documents that are relevant
  - ✓ Recall ratio: the fraction of relevant documents that are retrieved
- f.i., it became clear that, in practical situations, the <u>majority of searches</u> does <u>not</u> require <u>high recall</u>
- instead, the vast majority of the users <u>require just a few relevant answers</u> and <u>are more likely to select documents higher up in the ranking</u> (*rank bias*)

The Cranfield Paradigm

- use of a **test reference collection** composed of documents, queries, and relevance judgments
- **Relevance = Topical relevance**
  whether a document contains information on the same topic as the query
- it became known as the ***Cranfield 2* collection**
  - 1400 research papers on aeronautics (single domain) in English
  - 221 topics (queries)
- the reference collection allows using the same set of documents and queries to evaluate different ranking systems
- the uniformity of this setup allows quick evaluation of new ranking functions

The Cranfield Paradigm

1. select different retrieval strategies/systems to compare
2. use these to produce ranked lists of documents (*runs*) for each query (*topics*)
3. compute the effectiveness of each strategy for every query in the test collection as a function of relevant documents retrieved
4. average the scores over all queries to compute overall effectiveness of the strategy or system
5. use the scores to rank the strategies/systems relative to each other
6. (optional, to determine the 'best' approach) perform statistical tests to determine whether the differences between effectiveness scores for strategies/systems and their rankings are significant

Human Labeled Corpus (Test collection)

- Start with a corpus of documents
- Collect a set of information needs (not queries) for this corpus
- Have one or more human experts exhaustively label the relevant documents for each information need
- Typically assumes **binary relevance judgments**
- Requires considerable human effort for large corpus

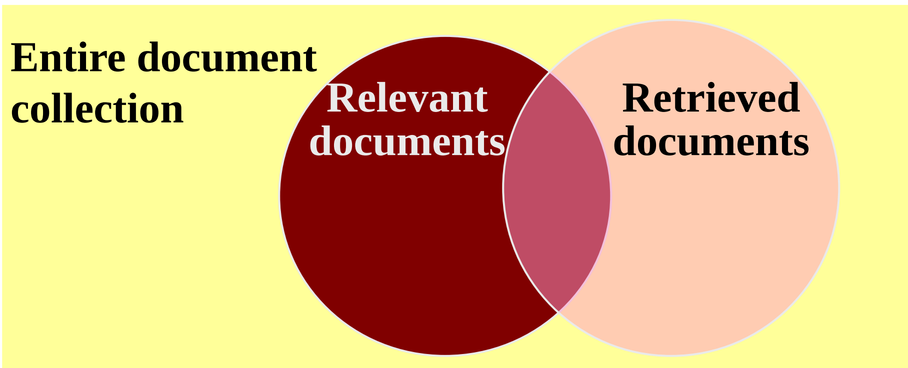
Standard relevance benchmarks

- **TREC (Text REtrieval Conference)**
  - National Institute of Standards and Technology (NIST) has run a large IR test bed (1M docs) for many years since 1992
  - annual series of workshops http://trec.nist.gov/data.html
  - In 1970s, the idea of "ideal" test collection was proposed by Karen Sparck Jones, but such test collection was not built until the TREC project began in 1992. TREC makes use of Cranfield paradigm evaluation to evaluate IR systems for various tasks
- ***ClueWeb09 collection*** (TREC Web track)
  - > 1B web pages in 10 languages on several domains
  - http://lemurproject.org/clueweb09/
- **ISILT** (Keen and Digger, 1972), **UKCIS** (Barker et al., 1974), **MEDLARS** (Barraclough et al., 1972) (Lancaster, 1968)

Standard relevance benchmarks

- **Reuters** and other benchmark doc collections used
- "Retrieval tasks" specified, sometimes as queries
- Human experts mark, for each query and for each doc, Relevant or Nonrelevant
  - at least for **subsets** of docs (***pooling***) that some system returned for that query (in the TREC-style version of the Cranfield approach)

Precision and Recall (Kent et al. 1955)

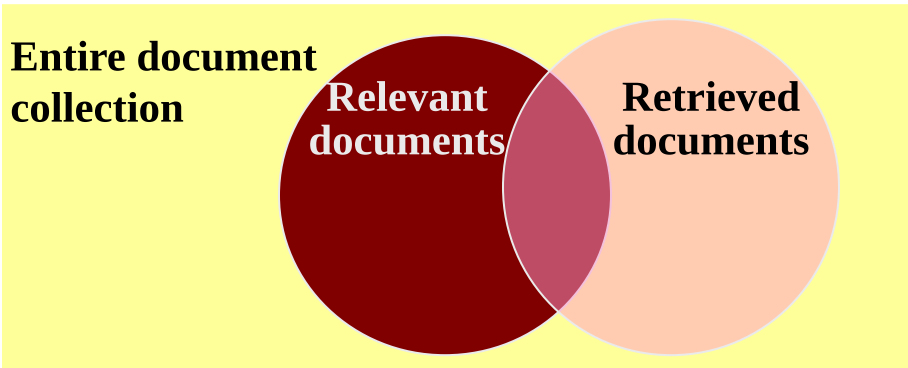


	retrieved	not retrieved
irrelevant	retrieved & irrelevant	not retrieved & irrelevant
relevant	retrieved & relevant	not retrieved but relevant

$$precision = \frac{Number\ of\ relevant\ documents\ retrieved}{Total\ number\ of\ documents\ retrieved}$$

$$recall = \frac{Number\ of\ relevant\ documents\ retrieved}{Total\ number\ of\ relevant\ documents}$$

Precision and Recall



	retrieved	not retrieved
irrelevant	False Positive (FP)	True Negative (TN)
relevant	True Positive (TP)	False Negative (FN)

$$precision = \frac{Number\ \ of\ \ relevant\ documents\ \ retrieved}{Total\ number\ of\ \ documents\ \ retrieved} = \frac{TP}{TP + FP}$$

$$recall = \frac{Number\ of\ \ relevant\ documents\ retrieved}{Total\ number\ of\ \ relevant\ documents} = \frac{TP}{TP + FN}$$

Precision and Recall

- Precision
  - Fraction of retrieved docs that are relevant
  - The ability to retrieve top-ranked documents that are mostly relevant
  - Precision = P(relevant | retrieved)
- Recall
  - Fraction of relevant docs that are retrieved
  - The ability of the search to find (***all*** of) the relevant items in the corpus
  - Recall = P(retrieved | relevant)

Accuracy

- Given a query, an engine (**classifier**) classifies each doc as “Relevant” or “Nonrelevant”
  - What is retrieved is classified by the engine as "relevant" and what is not retrieved is classified as "nonrelevant"
- The **accuracy** of the engine: the fraction of these classifications that are correct
  - (TP + TN) / (TP + FP + TN + FN)
- **Accuracy** is a commonly used evaluation measure in **machine learning** classification work
- Why is this not a very useful evaluation measure in IR?

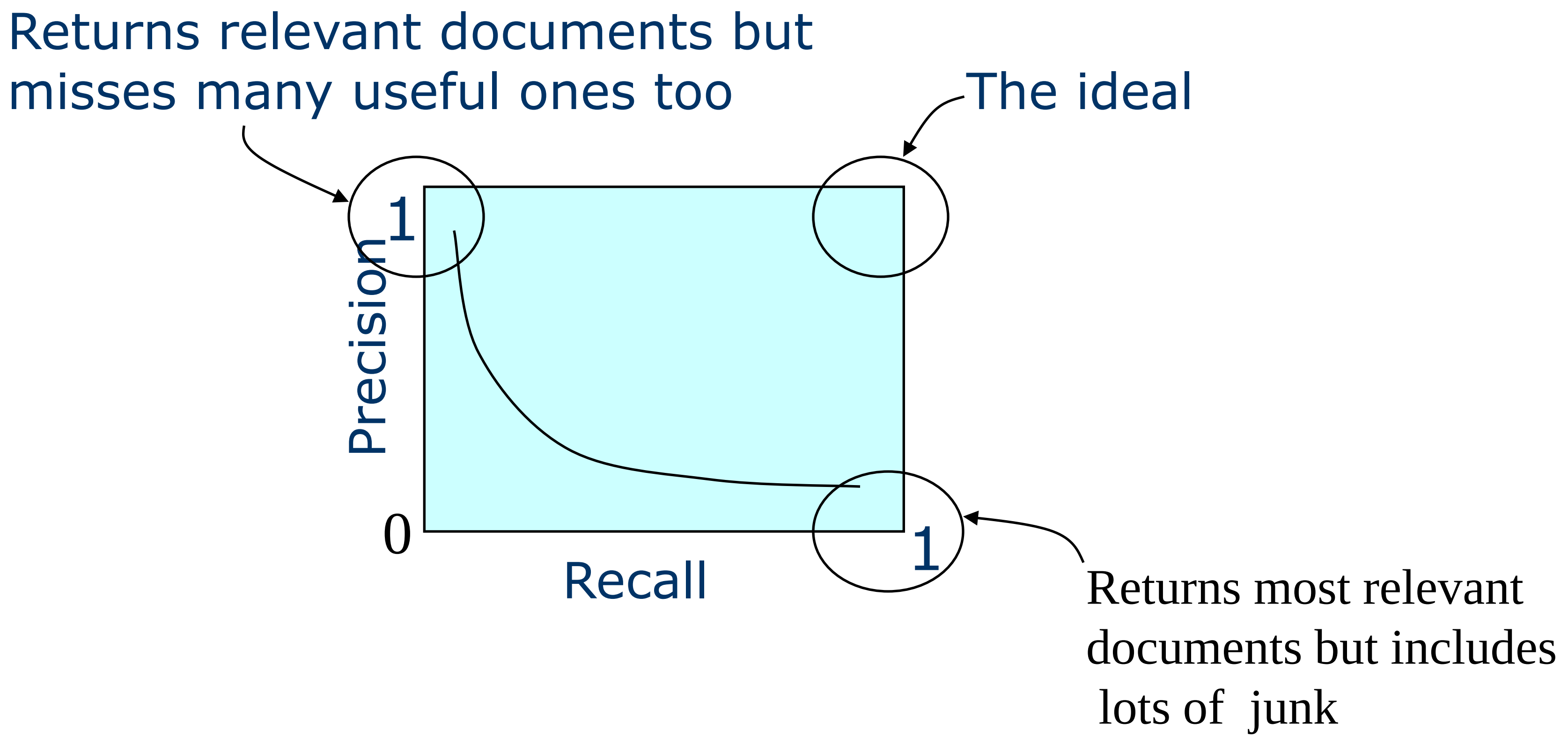
Precision/Recall

- What is the recall of a query if you retrieve all the documents?
- You can get high recall (but low precision) by retrieving all docs for all queries!
- **Recall is a non-decreasing function of the number of docs retrieved. Why?**
  - ✓ By increasing the number of retrieved documents, for instance by 1:
    if it is **relevant** then TP = TP + 1 e FN = FN -1
    if it is **irrelevant** then FP = FP + 1 e TN = TN -1
    In both cases **recall** TP /(TP + FN) **does not change**.
- In a good system, **precision decreases as either the number of docs retrieved increases or recall increases**
  - ✓ This is not a theorem, but a result with strong empirical confirmation.

Determining Recall is Difficult

- Total number of relevant items is sometimes not available:
  - Sample across the database and perform relevance judgment on these items.
  - Apply different retrieval algorithms to the same database for the same query. The aggregate of relevant items is taken as the total relevant set.

Recall vs. Precision



Trade-off between Recall and Precision

Difficulties in using precision/recall

- Should average over large document 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.

F1-Measure

- One measure of performance that takes into account both recall and precision.
- Harmonic mean of recall and precision:

$$F = \frac{2PR}{P+R} = \frac{2}{\frac{1}{R}+\frac{1}{P}}$$

- Compared to arithmetic mean, both need to be high for harmonic mean to be high.

F_β Measure (parameterized F Measure)

- A variant of F measure that allows weighting emphasis on precision over recall:

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

- Value of β controls trade-off:
  - $\beta = 1$: Equally weight precision and recall (F_β=F)
  - $\beta > 1$: Weight recall more
  - $\beta < 1$: Weight precision more
  - $\beta = 0$: F_β =P

- E Measure = 1- F_β

Evaluating ranked results

- Evaluation of ranked results:
  - The system can return any number of results – by varying its behavior or
  - By taking various numbers of the top returned documents (levels of recall), the evaluator can produce a *precision-recall curve.*

Precision-Recall

What is 1000?

Google copland Sea

Web Show options...

Cop Land (1997)
Do you know that he was paid only $60.000 for his acting in **Cop Land**, ... To me **Cop land** is the kind of movie Stallone should have made after First Blood. ...
www.imdb.com/title/tt0118887/ - 13 hours ago - Cached - Similar

P=0/1, R=0/1000

Aaron **Copland** - Wikipedia, the free encyclopedia
Before emigrating from Scotland to the United States, **Copland's** father, ... Travels to Italy, Austria, and Germany rounded out **Copland's** musical education. ...
Biography - Composer - Film composer - Critic, writer, and teacher
en.wikipedia.org/wiki/Aaron_**Copland** - Cached - Similar

P=1/2, R=1/1000

Copland - Wikipedia, the free encyclopedia
From Wikipedia, the free encyclopedia. Jump to: navigation, search. **Copland** can mean: [ec
Surname. Aaron **Copland** (1900–1990), American composer ...
en.wikipedia.org/wiki/**Copland** - Cached - Similar

Show more results from en.wikipedia.org

Books by **Aaron Copland**
What to Listen for in Music - 2002 - 308 pages
Music and Imagination - 1980 - 134 pages
Aaron Copland: A Reader Selected Writings 1923 ... - 2004 - 416 pages
books.google.it - More book results »

P=2/3, R=2/1000

COPLAND
Maker and one line of products: stereo and multi-channel valve amplifier, stereo and multi-channel power amplifier and cd player.
www.**copland**.dk/ - Cached - Similar

P=2/4, R=2/1000

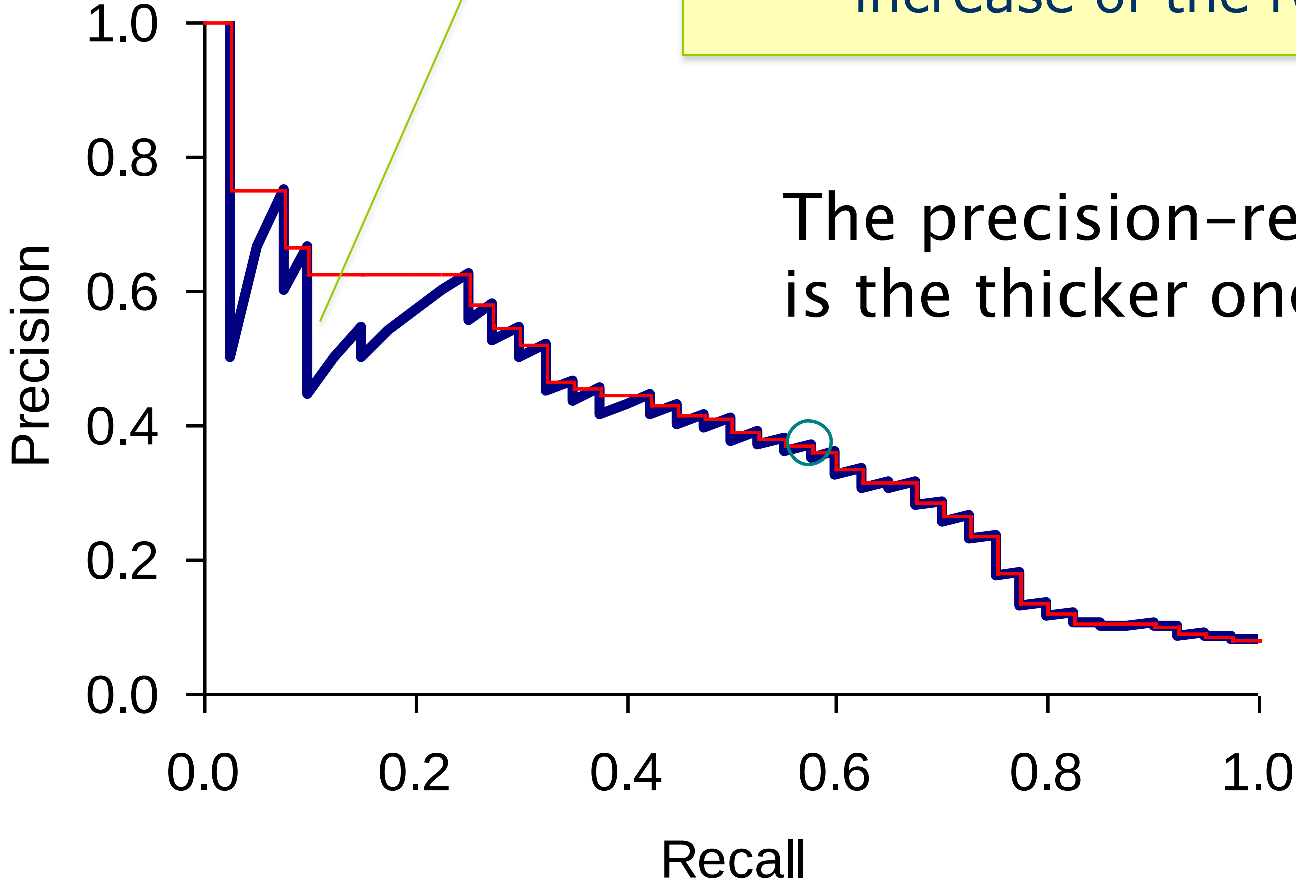
Aaron **Copland** | American Composer
4 Jan 2010 ... Lucidcafé's profile noting life, works, and style with photograph and links.
www.lucidcafe.com/library/95nov/**copland**.html - Cached - Similar

P=3/5, R=3/1000

Classical Net - Basic Repertoire List - **Copland**
As much as anyone, Aaron **Copland** established American concert music through his

A precision-recall curve

What is happening here where precision decreases without an increase of the recall?



The precision-recall curve is the thicker one

Averaging over queries

- A precision-recall graph for one query isn't a very sensible thing to look at
- You need to **average** performance over a whole bunch of queries
- But there's a technical issue:
  - Precision-recall calculations place some points on the graph
  - How do you determine a value (interpolate) between the points?

Computing Recall/Precision points (example 1)

n	doc #	relevant
1	588	x
2	589	x
3	576	
4	590	x
5	986	
6	592	x
7	984	
8	988	
9	578	
10	985	
11	103	
12	591	
13	772	x
14	990	

Let total # of relevant docs be = 6

Check each new recall point:

R=1/6=0.167; P=1/1=1

R=2/6=0.333; P=2/2=1

R=3/6=0.5; P=3/4=0.75

R=4/6=0.667; P=4/6=0.667

R=5/6=0.833; P=5/13=0.38

Missing one relevant document. Never reach 100% recall

Computing Recall/Precision points (example 2)

n	doc #	relevant
1	588	x
2	576	
3	589	x
4	342	
5	590	x
6	717	
7	984	
8	772	x
9	321	x
10	498	
11	113	
12	628	
13	772	
14	592	x

Let total # of relevant docs = 6

Check each new recall point:

R=1/6=0.167; P=1/1=1

R=2/6=0.333; P=2/3=0.667

R=3/6=0.5; P=3/5=0.6

R=4/6=0.667; P=4/8=0.5

R=5/6=0.833; P=5/9=0.556

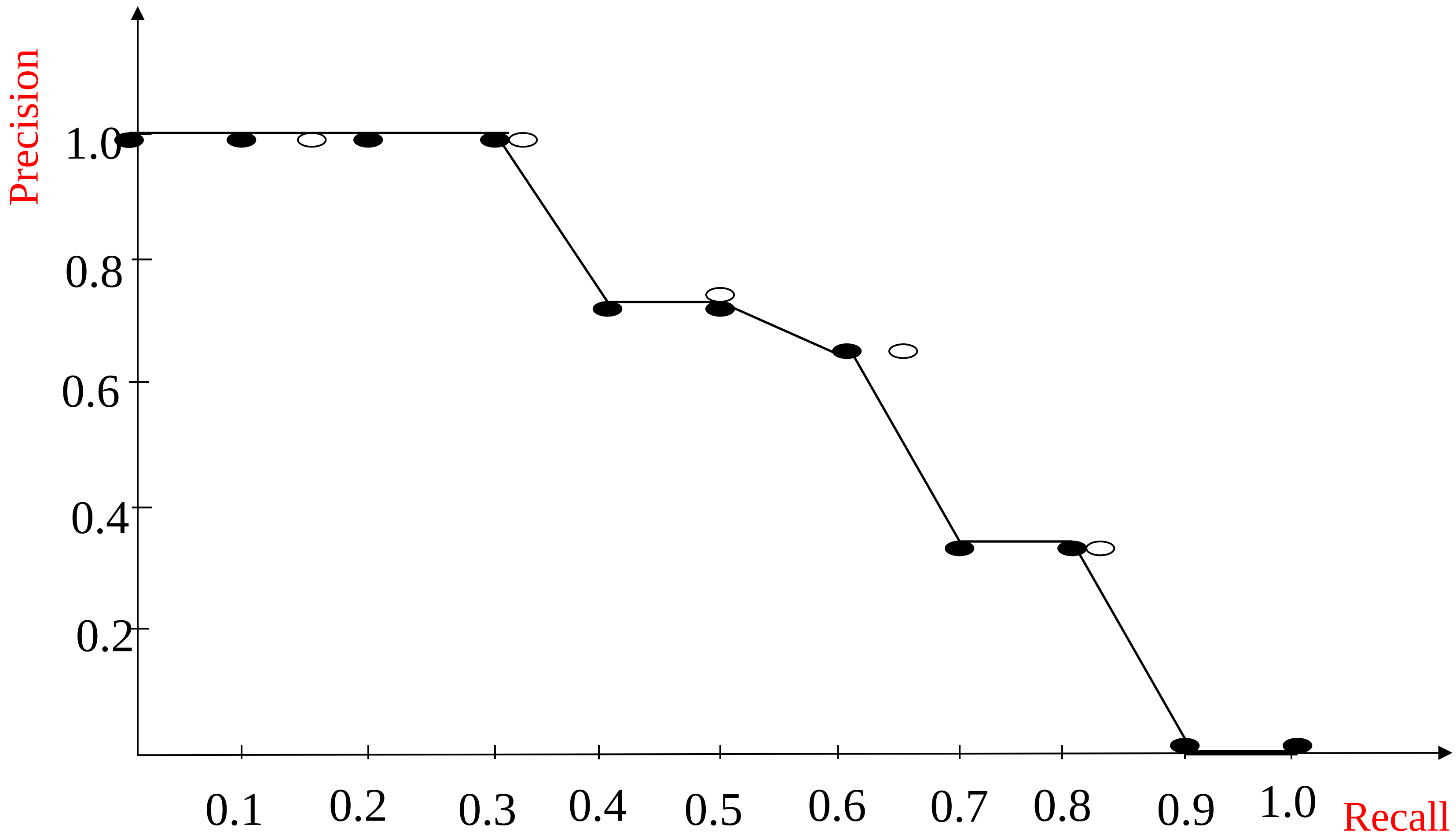
R=6/6=1.0; p=6/14=0.429

Interpolating a Recall/Precision Curve

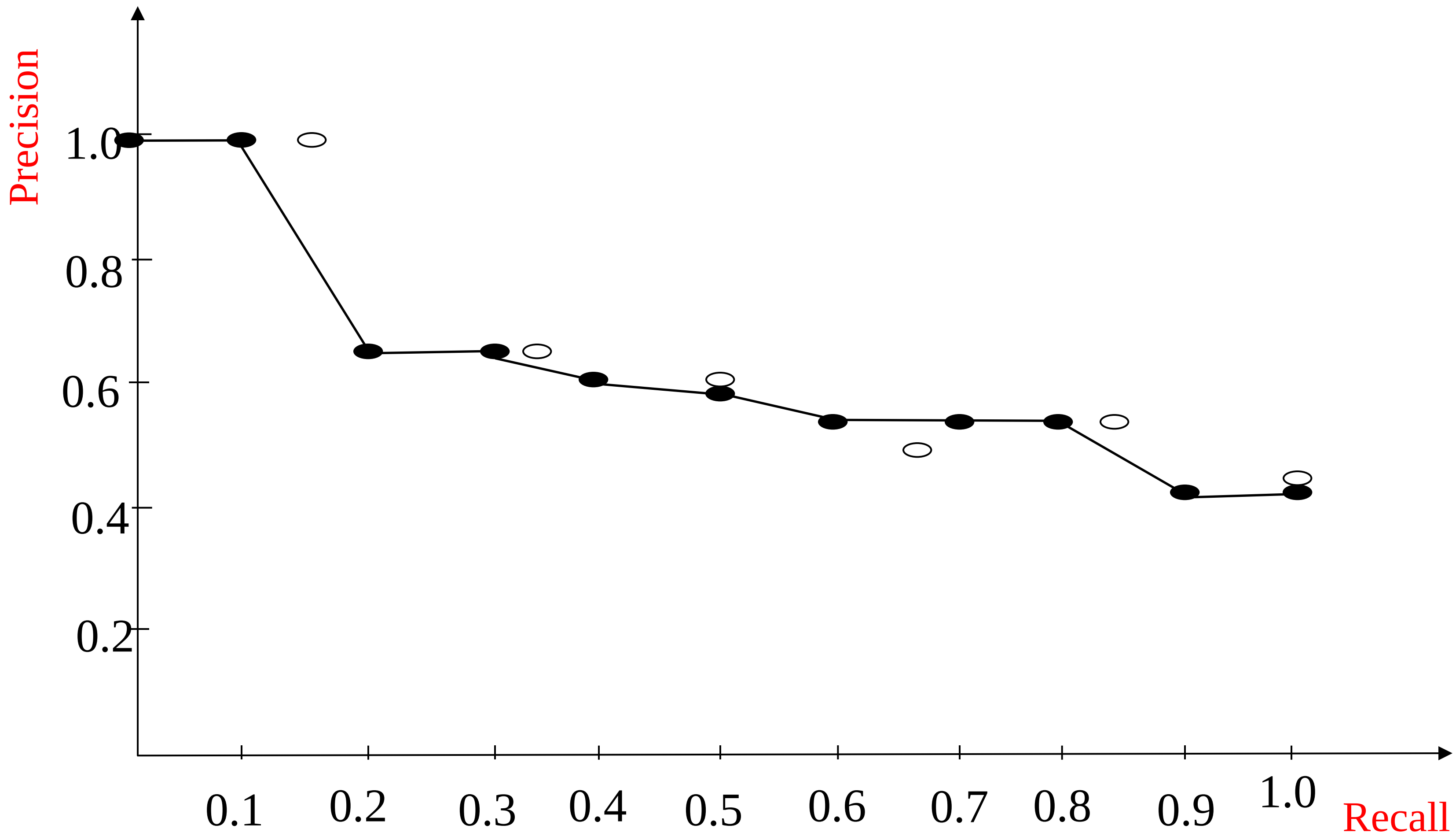
- Interpolate a precision value for each *standard recall level*:
  - ✓ $r_j \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$
  - ✓ $r_0 = 0.0$, $r_1 = 0.1$, …, $r_{10}=1.0$
- The interpolated precision at the *j*-th standard recall level is the maximum known precision among all recall levels above r_j:

$$P(r_j) = \max_{\forall r | r_j \le r} P(r)$$

Recall/Precision Curve: Example 1

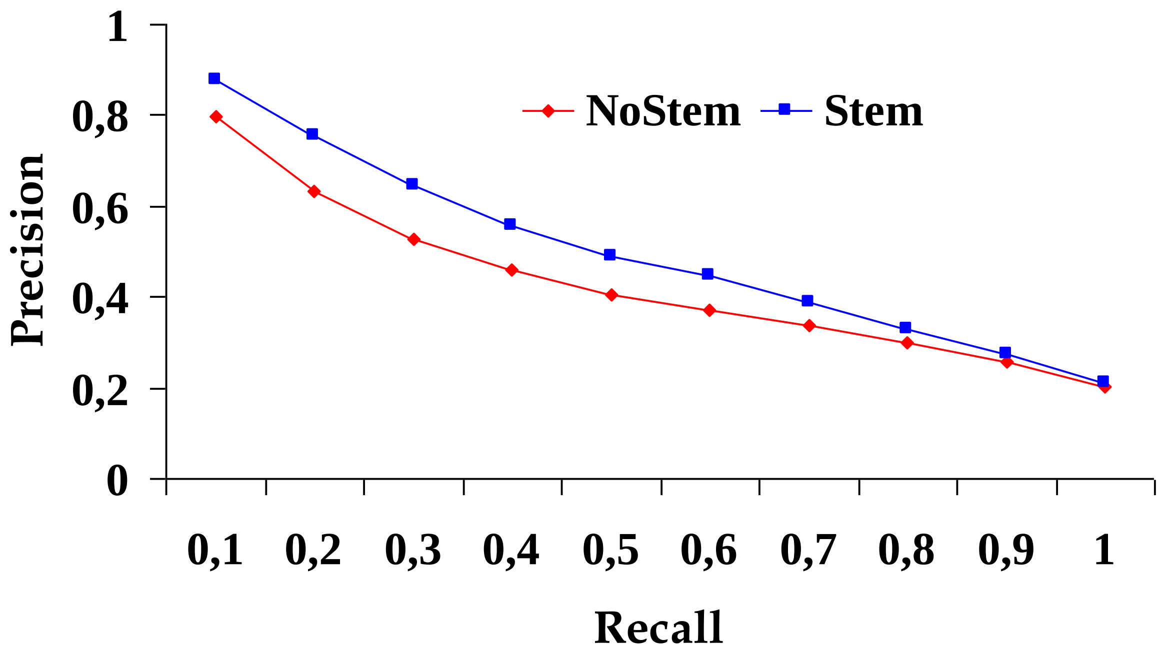


Recall/Precision Curve: Example 2



Compare Two or More Systems

- The curve closest to the upper right-hand corner of the graph indicates the best performance



Evaluation: Precision@k

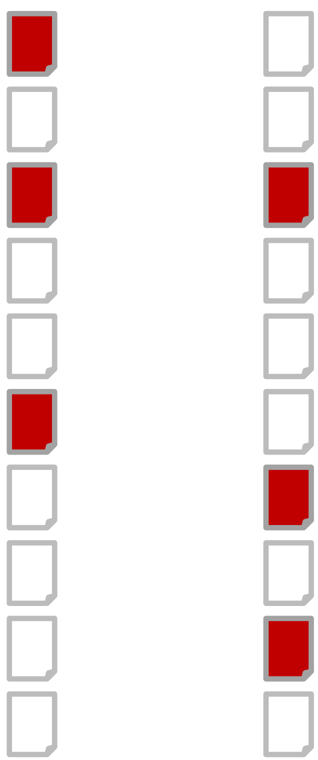
- Graphs are good, but people want summary measures!
- Precision at fixed retrieval level
- **Precision@*k*:** Precision at rank *k* = Precision of top *k* results (commonly used values *k*=10, 20)
  - Pro 1: perhaps appropriate for most of web search - all people want are good matches on the first one or two result pages
  - Pro 2: useful to estimate a ***cutoff*** *value k*
  - Cons 1: averages badly and has an arbitrary parameter *k*
  - Cons 2: no distinction between different rankings of the same number of relevant documents (***set-based*** measure)
  - Cons 3: the choice of *k* may be misleading, influences the results and the reliability of an evaluation (e.g., if *k*=10 and #rel_docs(q)=5, P@10 will never reach 1 even for the perfect IR system)

Unranked vs. Ranked Effectiveness Measures

Relevant (5 docs)

Not relevant

IRS_1 IRS_2



IRS_1 vs. IRS_2:

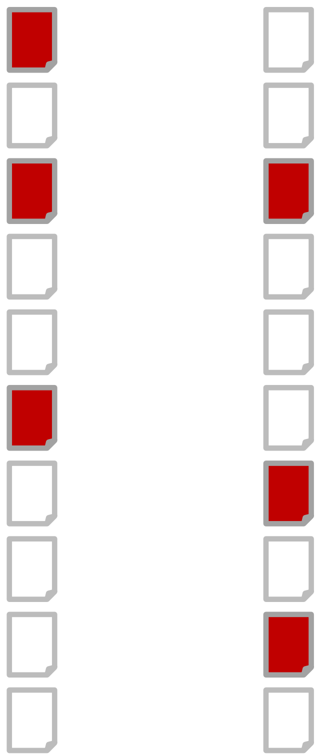
which is the better?

Unranked vs. Ranked Effectiveness Measures

Relevant (5 docs)

Not relevant

IRS_1 IRS_2



IRS_1 vs. IRS_2:

which is the better?

$$\text{RECALL} = \frac{\text{Retrieved \& Relevant}}{\text{Tot. Relevant}}$$

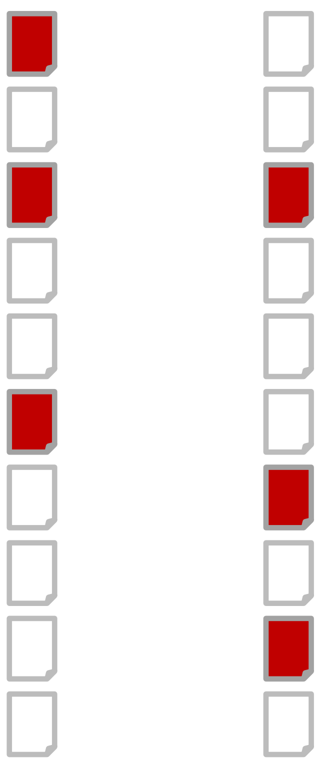
$$\text{RECALL}(\text{IRS}_1) = \frac{3}{5} = \text{RECALL}(\text{IRS}_2)$$

Unranked vs. Ranked Effectiveness Measures

Relevant (5 docs)

Not relevant

IRS_1 IRS_2



IRS_1 vs. IRS_2:

which is the better?

$$\text{PRECISION} = \frac{\text{Retrieved \& Relevant}}{\text{Tot. Retrieved}}$$

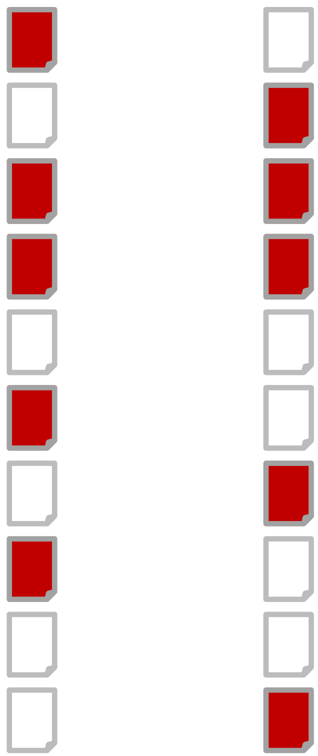
$$\text{PRECISION}(\text{IRS}_1) = \frac{3}{10} = \text{PRECISION}(\text{IRS}_2)$$

Ranked Effectiveness Measures

Relevant (5 docs)

Not relevant

IRS$_1$ IRS$_2$



- Computing precision at different ranking position *p*

Ranked Effectiveness Measures

- Precision at rank position *p*

IRS_1

P=1	R 0.2	P 1
P=2	R 0.2	P 0.5
P=3	R 0.4	P 0.67
P=4	R 0.6	P 0.75
P=5	R 0.6	P 0.6
P=6	R 0.8	P 0.67
P=7	R 0.8	P 0.57
P=8	R 1	P 0.63
P=9	R 1	P 0.56
P=10	R 1	P 0.5

IRS_2

P=1	R 0	P 0
P=2	R 0.2	P 0.5
P=3	R 0.4	P 0.67
P=4	R 0.6	P 0.75
P=5	R 0.6	P 0.6
P=6	R 0.6	P 0.5
P=7	R 0.8	P 0.57
P=8	R 0.8	P 0.5
P=9	R 0.8	P 0.44
P=10	R 1	P 0.5

Ranked Effectiveness Measures

- Precision at rank position *p*

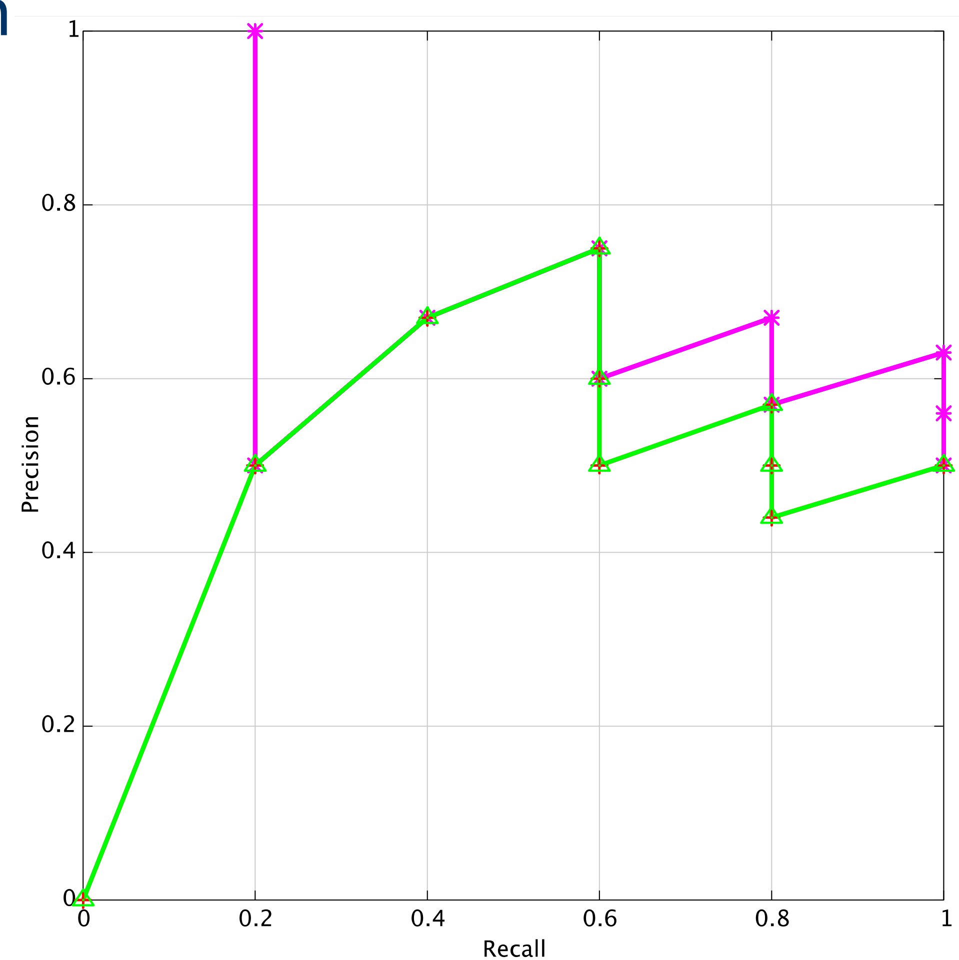
> If Precision $IRS_1 > IRS_2$ at p then also recall $IRS_1 > IRS_2$

	IRS_1		IRS_2
P=1	R 0.2 P 1	*P*=1	R 0 P 0
P=2	R 0.2 P 0.5	*P*=2	R 0.2 P 0.5
P=3	R 0.4 P 0.67	*P*=3	R 0.4 P 0.67
P=4	R 0.6 P 0.75	*P*=4	R 0.6 P 0.75
P=5	R 0.6 P 0.6	*P*=5	R 0.6 P 0.6
P=6	R 0.8 P 0.67	*P*=6	R 0.6 P 0.5
P=7	R 0.8 P 0.57	*P*=7	R 0.8 P 0.57
P=8	R 1 P 0.63	*P*=8	R 0.8 P 0.5
P=9	R 1 P 0.56	*P*=9	R 0.8 P 0.44
P=10	R 1 P 0.5	*P*=10	R 1 P 0.5

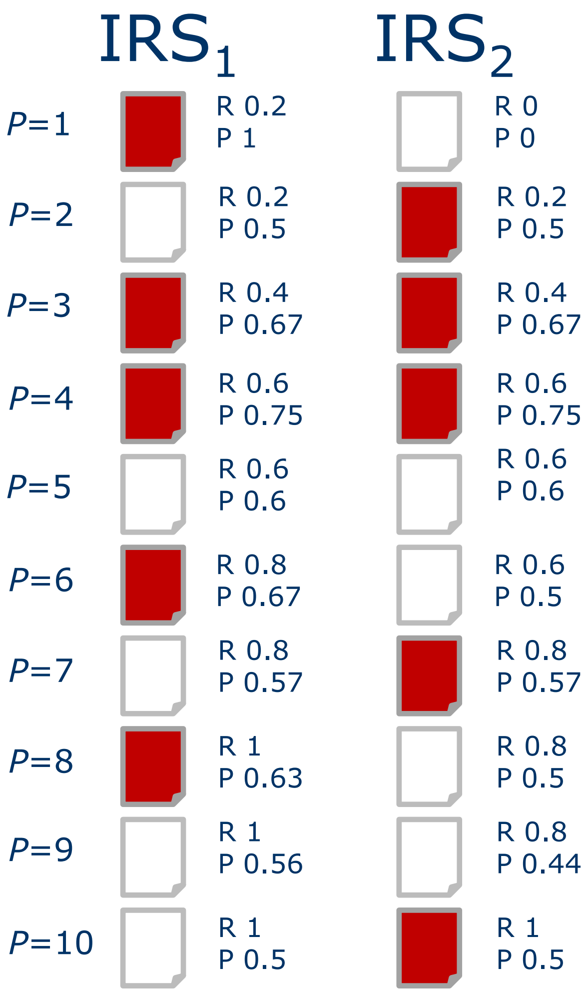
Ranked Effectiveness Measures

- Recall-Precision Graph

	IRS_1	IRS_2
$P=1$	R 0.2 P 1	R 0 P 0
$P=2$	R 0.2 P 0.5	R 0.2 P 0.5
$P=3$	R 0.4 P 0.67	R 0.4 P 0.67
$P=4$	R 0.6 P 0.75	R 0.6 P 0.75
$P=5$	R 0.6 P 0.6	R 0.6 P 0.6
$P=6$	R 0.8 P 0.67	R 0.6 P 0.5
$P=7$	R 0.8 P 0.57	R 0.8 P 0.57
$P=8$	R 1 P 0.63	R 0.8 P 0.5
$P=9$	R 1 P 0.56	R 0.8 P 0.44
$P=10$	R 1 P 0.5	R 1 P 0.5



Ranked Effectiveness Measures



Average Precision

- Averaging precision values from the *p* positions where a relevant document is retrieved

Rel. doc.

$$AP = \frac{1}{m} \sum_{k=1}^{m} \text{Precision}(p = k)$$

- AP(IRS$_1$) = (1+0.67+0.75+0.67+0.63)/5 = 0.74
- AP(IRS$_2$) = (0.5+0.67+0.75+0.57+0.5)/5 = 0.6

MAP

- Mean Average Precision: averaging the average precision over a set of queries
  - *n*: number of queries
  - m_j: number of relevant documents for the *j*-th query

$$\mathrm{MAP} = \frac{1}{n}\sum_{1}^{n}\frac{1}{m_j}\sum_{k=1}^{m_j}\mathrm{PRECISION}(p = k)$$

MAP

Why MAP is not enough?

	IRS1	IRS2
Q1	0.02	0.06
Q2	0.4	0.3
Q3	0.3	0.3
Q4	0.18	0.24
MAP	0.225	0.225

Differences in query AP are summed to same values

GMAP

- GMAP: Geometric Mean Average Precision
  - Increases in small values have a stronger impact on the final value (product, rather than sum, AP values)
  - Ideal for testing systems on “difficult” queries, e.g. queries where few relevant document are retrieved

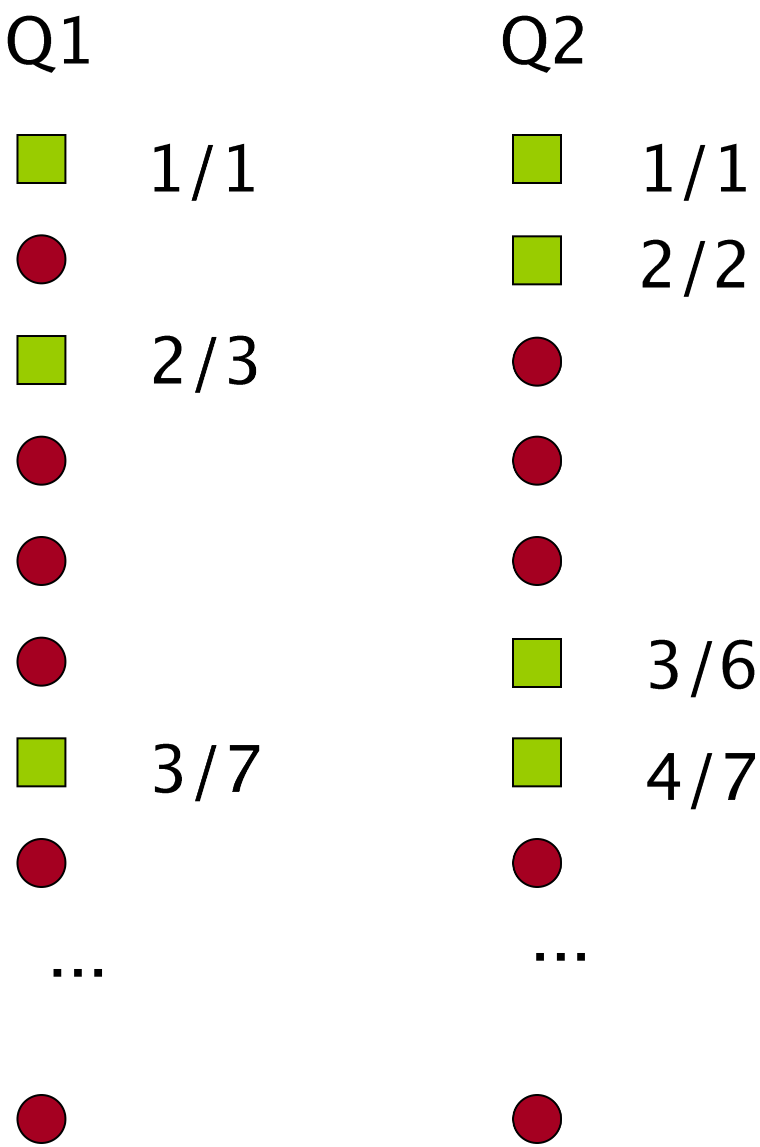
$$\mathrm{GMAP} = \sqrt[n]{\prod_{n} AP_n}$$

GMAP

	IRS1	IRS2
Q1	0.02	0.06
Q2	0.4	0.3
Q3	0.3	0.3
Q4	0.18	0.24
MAP	0.225	0.225
GMAP	0.144	0.19

Differences in query AP are reflected in GMAP values

MAP Example



$(1+2/3+3/7)/3 = 0.69$

$(1+1+3/6+4/7)/4 = 0.76$

Mean Average precision =

$(0.69 + 0.76)/2 = 0.72$

MAP Example

- **Average Precision** (**AP**): Average of the precision values at the points at which each relevant document is retrieved (equal to the areas under the precision-recall curves)
  - ✓ Ex1: (1 + 1 + 0.75 + 0.667 + 0.38 + 0)/6 = 0.633
  - ✓ Ex2: (1 + 0.667 + 0.6 + 0.5 + 0.556 + 0.429)/6 = 0.625

- **Mean Average Precision** (**MAP**): Arithmetic mean of average precision values for a set of queries.

R-precision

- Precision at the R-th position in the ranking, where *R* is the total number of relevant documents for the query
- Perfect system could score 1.0.
- The R-precision measure is useful for observing the behavior of an algorithm for individual queries
- R-precision could be averaged over all queries

R-Precision

- Precision at the R-th position in the ranking of results for a query that has R relevant documents.

n	doc #	relevant
1	588	x
2	589	x
3	576	
4	590	x
5	986	
6	592	x
7	984	
8	988	
9	578	
10	985	
11	103	
12	591	
13	772	x
14	990	

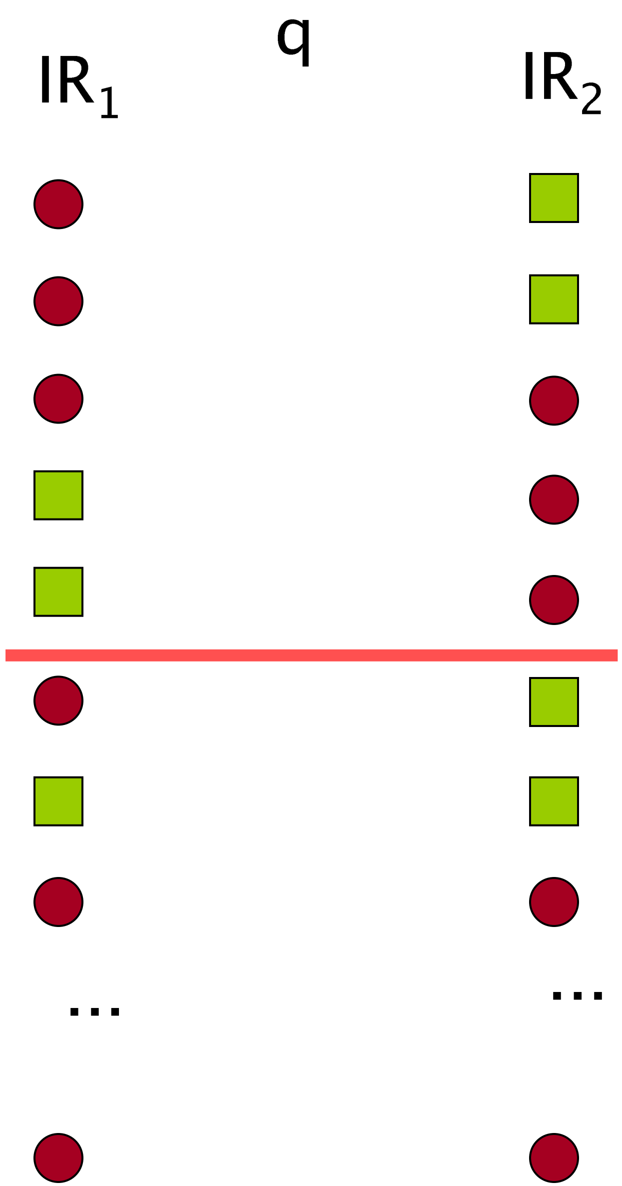
R = # of relevant docs = 6

R-Precision = 4/6 = 0.67

Mean Reciprocal Rank (MRR)

- Focus/need:
  measure how well the search engine retrieves relevant documents at very high ranks
- Recall is not an appropriate measure
- Is Precision@*k* what we are looking for?

Mean Reciprocal Rank (MRR) *cont'd*



IR_2 better than IR_1

but

Precision@5(q,IR_2)=2

Precision@5(q,IR_1)=2

● nonrelevant

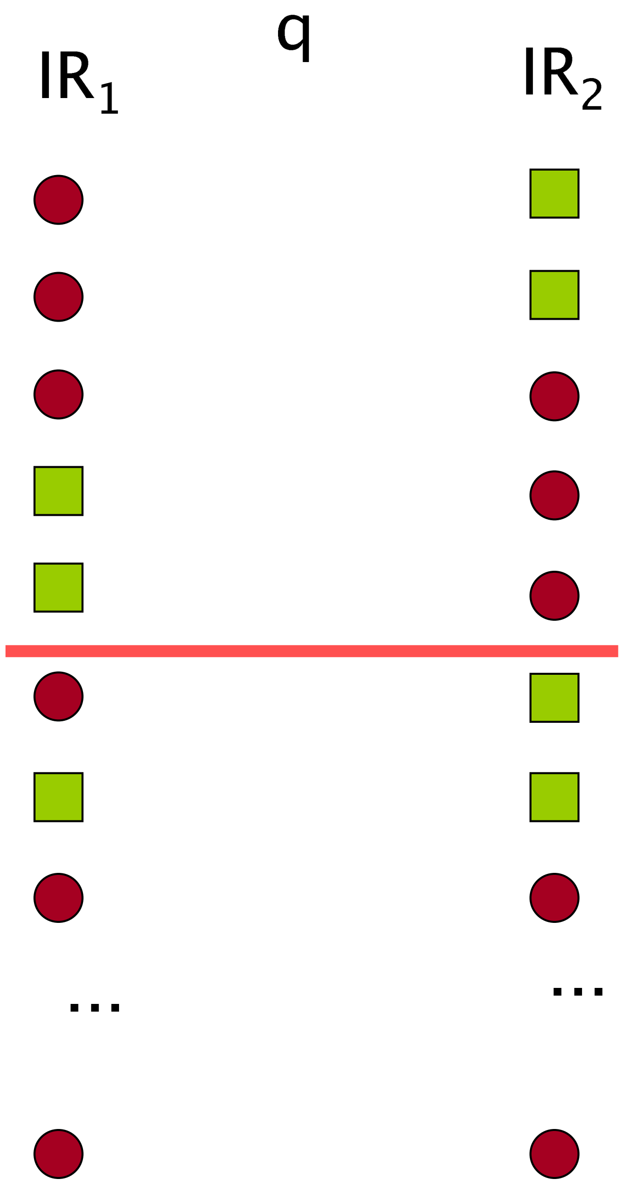
■ relevant

Mean Reciprocal Rank (MRR) *cont'd*

- we need a measure more sensitive to the rank position
- the Reciprocal Rank is defined as the reciprocal of the rank at which the 1st relevant document is retrieved
- the Mean Reciprocal Rank (MRR) (Kantor and Voorhees, 2000) is the average of the Reciprocal Ranks (RR) over a set of queries

$$\mathrm{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\mathrm{rank}_i}.$$

Mean Reciprocal Rank (MRR) *cont'd*



IR_2 better than IR_1

Indeed

$RR(q, IR_2) = 1/1$

$RR(q, IR_1) = 1/4$

Mean Reciprocal Rank (MRR) *cont'd*

Query	Results	Correct response	Rank	Reciprocal Rank
cat	catten, cati, **cats**	cats	3	1/3
torus	torii, **tori**, toruses	tori	2	1/2
virus	**viruses**, virii, viri	viruses	1	1

MRR = (1/3 + 1/2 + 1)/3 = 11/18 ≈ 0.61

- MRR is a good metric for those cases in which we are interested in the 1st correct answer
  - Question-Answering (QA) systems
  - Navigational search
    search engine queries that look for specific sites
    - URL queries
    - Home-page queries
    - Named-page queries

Graded (Non-Binary) Relevance

- Precision and Recall allow only binary relevance assessments
  - Documents are rarely entirely relevant or non-relevant to a query
  - As a result, there is no distinction between highly relevant docs and mildly relevant docs
- These limitations can be overcome by adopting **graded relevance metrics/measures** that combine them
  - many sources of *graded relevance judgments* (Relevance judgments on a 5-point Likert scale)
- In the case of **graded relevance** a document is judged for relevance on a **scale with multiple categories**, e.g., highly relevant, partially relevant or non-relevant

Discounted Cumulative Gain (DCG)

- The **Discounted Cumulated Gain** (**DCG**) (Järvelin and Kekäläinen, 2002) is a metric that combines graded relevance assessments effectively
- When examining the results of a query, 2 key observations can be made:
  - ✓ highly relevant documents are more useful than marginally relevant documents
  - ✓ the lower the ranked position of a relevant document (i.e., further down the ranked list), the less useful it is for the user, since it is less likely to be examined

Discounted Cumulative Gain (DCG)

- suppose that the results of the queries are graded on a scale 0–3 (0 for non-relevant, 3 for strong relevant docs)
- for queries q_1 and q_2, suppose that the graded relevance scores are as follows:

R_{q1} = { *[d3, 3], [d5, 3], [d9, 3], [d25, 2], [d39, 2], [d44, 2], [d56, 1], [d71, 1], [d89, 1], [d123, 1]* }

R_{q2} = { *[d3, 3], [d56, 2], [d129, 1]* }

that is, while document d_3 is highly relevant to query q_1, document d_{56} is just mildly relevant

DCG: Gain vector (G)

- given these graded-relevance judgments (assessments), the results of a new ranking algorithm can be evaluated as follows
- Specialists associate a graded-relevance judgment to the top 10-20 results produced for a given query
  - *this list of relevance scores is referred to as the **Gain vector G***
- Considering the top 15 docs in the ranking produced for queries q_1 and q_2, the gain vectors for these queries are:

$$G_1 = (1, 0, 1, 0, 0, 3, 0, 0, 0, 2, 0, 0, 0, 0, 3)$$
$$G_2 = (0, 0, 2, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 3)$$

DCG: Cumulated Gain (CG)

- By summing up the graded scores up to any point in the ranking, we obtain the **direct Cumulated Gain** (**CG**) (Järvelin and Kekäläinen, 2000)
- For query q_1, for instance, the cumulated gain at the first position is 1, at the second position is 1+0, and so on
- Thus, the cumulated gain vectors for queries q_1 and q_2 are given by

  CG_1 = (1, 1, 2, 2, 2, 5, 5, 5, 5, 7, 7, 7, 7, 7, 10)

  CG_2 = (0, 0, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 6)
- For instance, the cumulated gain at position 8 of CG_1 is equal to 5

Discounted Cumulative Gain

- In formal terms, we define
  - ✓ Given the gain vector G_j for a test query q_j , the CG_j associated with it is defined as

$$CG_j[i] = \begin{cases} G_j[1] & \texttt{if } i = 1; \\ G_j[i] + CG_j[i-1] & \texttt{otherwise} \end{cases}$$

where CG_j [i] refers to the cumulated gain at the i-th position of the ranking for query q_j

Discounted Cumulative Gain

- We also introduce a **Discount factor** (**D**) that reduces the impact of the gain as we move upper in the ranking
- A simple discount factor is the logarithm of the ranking position
- If we consider logs in base 2, this discount factor will be $\log_2 2$ at position 2, $\log_2 3$ at position 3, and so on
- By dividing a gain by the corresponding discount factor, we obtain the **Discounted Cumulated Gain** (**DCG**)

Discounted Cumulative Gain

- More formally,
  - Given the gain vector G_j for a test query q_j , the vector DCG_j associated with it is defined as

$$DCG_j[i] = \begin{cases} G_j[1] & \texttt{if } i = 1; \\ \frac{G_j[i]}{\log_2 i} + DCG_j[i-1] & \texttt{otherwise} \end{cases}$$

  where DCG_j [i] refers to the discounted cumulated gain at the i-th position of the ranking for query q_j

Discounted Cumulative Gain

- For the example queries q_1 and q_2, the DCG vectors are given by

  DCG_1 = (1.0, 1.0, 1.6, 1.6, 1.6, 2.8, 2.8, 2.8, 2.8, 3.4, 3.4, 3.4, 3.4, 3.4, 4.2)

  DCG_2 = (0.0, 0.0, 1.3, 1.3, 1.3, 1.3, 1.3, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 2.4)

- Discounted cumulated gains are much less affected by relevant documents at the end of the ranking
- By adopting logs in lower bases the discount factor can be accentuated

DCG Curves

- To produce CG and DCG curves over a set of test queries, we need to average them over all queries
- Given a set of N_q queries, average CG[i] and DCG[i] over all queries are computed as follows

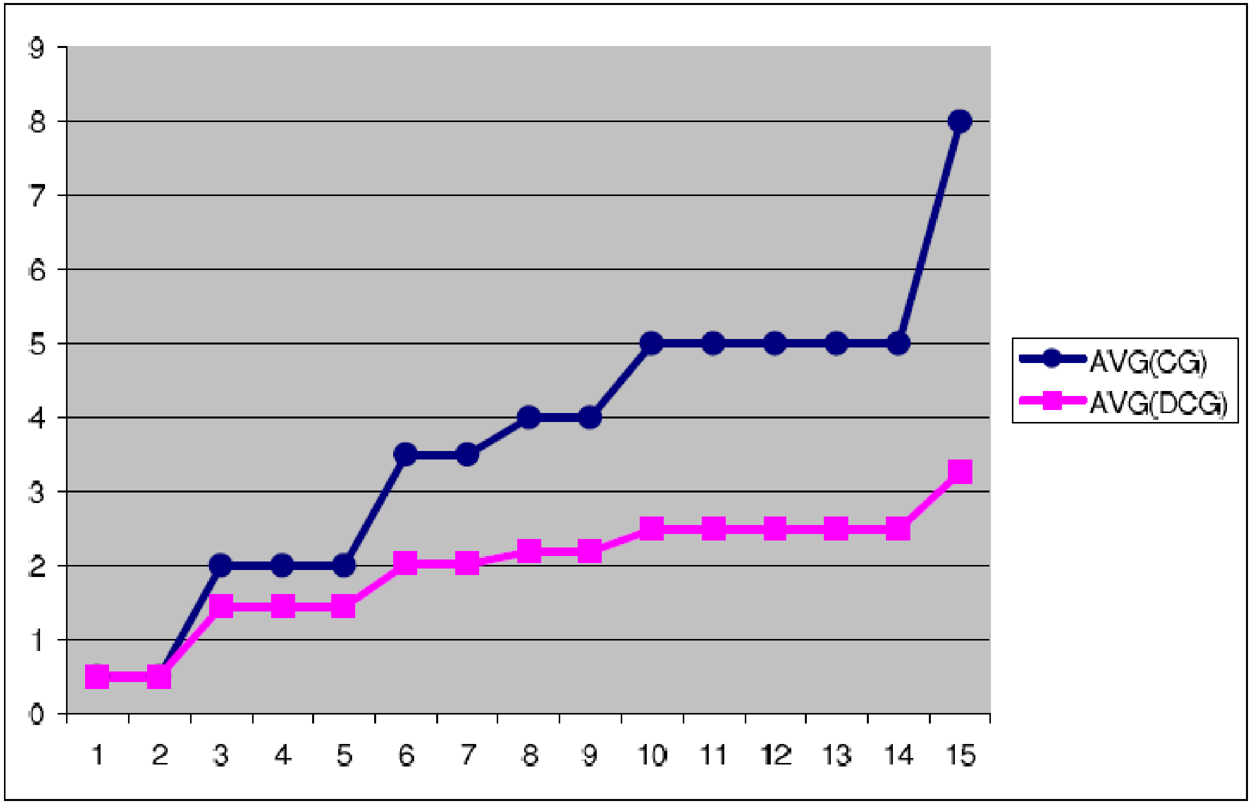
$$\overline{CG}[i] = \sum_{j=1}^{N_q} \frac{CG_j[i]}{N_q}; \qquad \overline{DCG}[i] = \sum_{j=1}^{N_q} \frac{DCG_j[i]}{N_q}$$

- For instance, for the example queries q1 and q2, these averages are given by

$$\begin{aligned}
\overline{CG} &= (0.5, 0.5, 2.0, 2.0, 2.0, 3.5, 3.5, 4.0, 4.0, 5.0, 5.0, 5.0, 5.0, 5.0, 8.0) \\
\overline{DCG} &= (0.5, 0.5, 1.5, 1.5, 1.5, 2.1, 2.1, 2.2, 2.2, 2.5, 2.5, 2.5, 2.5, 2.5, 3.3)
\end{aligned}$$

DCG Curves

- Then, average curves can be drawn by varying the rank positions from 1 to a pre-established threshold



Ideal CG and DCG Metrics

- Recall and precision figures are computed relatively to the set of relevant documents
- CG and DCG scores, as defined above, are not computed relatively to any baseline
- This implies that it might be confusing to use them directly to compare two distinct retrieval algorithms
- One solution to this problem is to define a baseline to be used for normalization
- This baseline are the ideal CG and DCG metrics

Ideal CG and DCG Metrics

- For a given test query q, assume that the relevance assessments made by the specialists produced:
  - n_3 documents evaluated with a relevance score of 3
  - n_2 documents evaluated with a relevance score of 2
  - n_1 documents evaluated with a score of 1
  - n_0 documents evaluated with a score of 0
- The ideal gain vector IG is created by sorting all relevance scores in decreasing order, as follows:

  $IG = (3, \ldots, 3, 2, \ldots, 2, 1, \ldots, 1, 0, \ldots, 0)$
- For instance, for the example queries q_1 and q_2:

  $IG_1 = (3, 3, 3, 2, 2, 2, 1, 1, 1, 1, 0, 0, 0, 0, 0)$

  $IG_2 = (3, 2, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$

Ideal CG and DCG Metrics

- Ideal CG and ideal DCG vectors can be computed analogously to the computations of CG and DCG
- For the example queries q_1 and q_2, we have

ICG_1 = (3, 6, 9, 11, 13, 15, 16, 17, 18, 19, 19, 19, 19, 19, 19)
ICG_2 = (3, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6)

- The ideal DCG vectors are given by

$IDCG_1$ = (3.0, 6.0, 7.9, 8.9, 9.8, 10.5, 10.9, 11.2, 11.5, 11.8, 11.8, 11.8, 11.8, 11.8, 11.8)
$IDCG_2$ = (3.0, 5.0, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6)

Ideal CG and DCG Metrics

- Further, average ICG and average IDCG scores can be computed as follows

$$\boxed{\overline{ICG}[i] = \sum_{j=1}^{N_q} \frac{ICG_j[i]}{N_q}; \qquad \overline{IDCG}[i] = \sum_{j=1}^{N_q} \frac{IDCG_j[i]}{N_q}}$$

- For instance, for the example queries q_1 and q_2, ICG and IDCG vectors are given by

$$\begin{aligned} \overline{ICG} &= (3.0, 5.5, 7.5, 8.5, 9.5, 10.5, 11.0, 11.5, 12.0, 12.5, 12.5, 12.5, 12.5, 12.5, 12.5) \\ \overline{IDCG} &= (3.0, 5.5, 6.8, 7.3, 7.7, 8.1, 8.3, 8.4, 8.6, 8.7, 8.7, 8.7, 8.7, 8.7, 8.7) \end{aligned}$$

- By comparing the average CG and DCG curves for an algorithm with the average ideal curves, we gain insight on how much room for improvement there is

Normalized DCG

- Precision and recall figures can be directly compared to the ideal curve of 100% precision at all recall levels
- DCG figures, however, are not built relative to any ideal curve, which makes it difficult to compare directly DCG curves for two distinct ranking algorithms
- This can be corrected by normalizing the DCG metric
- Given a set of N_q test queries, normalized CG and DCG metrics are given by

$$NCG[i] = \frac{\overline{CG}[i]}{\overline{ICG}[i]}; \qquad NDCG[i] = \frac{\overline{DCG}[i]}{\overline{IDCG}[i]}$$

Normalized DCG

- For instance, for the example queries q_1 and q_2, NCG and NDCG vectors are given by

$$
\begin{aligned}
NCG &= (0.17, 0.09, 0.27, 0.24, 0.21, 0.33, 0.32,\\
&\quad 0.35, 0.33, 0.40, 0.40, 0.40, 0.40, 0.40, 0.64)\\
NDCG &= (0.17, 0.09, 0.21, 0.20, 0.19, 0.25, 0.25,\\
&\quad 0.26, 0.26, 0.29, 0.29, 0.29, 0.29, 0.29, 0.38)
\end{aligned}
$$

- The area under the NCG and NDCG curves represent the quality of the ranking algorithm
- The higher the area, the better the results
- Thus, normalized figures can be used to compare two distinct ranking algorithms

Discussion on DCG Metrics

- CG and DCG metrics aim at taking into account multiple level relevance assessments
- This has the **advantage** of distinguishing highly relevant documents from mildly relevant ones
- The inherent **disadvantages** are that multiple level relevance assessments are harder and more time consuming to generate

Discussion on DCG Metrics

- Despite these inherent difficulties, the CG and DCG metrics present benefits:
  - They allow systematically combining document ranks and relevance scores
  - Cumulated gain provides a single metric of retrieval performance at any position in the ranking
  - It also stresses the gain produced by relevant docs up to a position in the ranking, which makes the metrics more immune to outliers
  - Further, discounted cumulated gain allows down weighting the impact of relevant documents found late in the ranking

Rank Correlation Metrics

- Precision and recall allow comparing the relevance of the results produced by two ranking functions
- However, there are situations in which
  - we cannot directly measure relevance
  - we are more interested in determining how differently a ranking function varies from a second one that we know well
- In these cases, we are interested in comparing the relative ordering produced by the two rankings
- This can be accomplished by using statistical functions called **rank correlation metrics**

Rank Correlation Metrics

- Let rankings R_1 and R_2
- A rank correlation metric yields a correlation coefficient $C(R_1,R_2)$ with the following properties:
  - $-1 <= C(R_1,R_2) <= 1$
  - if $C(R_1,R_2) = 1$, the agreement between the two rankings is perfect i.e., they are the same.
  - if $C(R_1,R_2) = -1$, the disagreement between the two rankings is perfect i.e., they are the reverse of each other.
  - if $C(R_1,R_2) = 0$, the two rankings are completely independent.
  - increasing values of $C(R_1,R_2)$ imply increasing agreement between the two rankings.

The Spearman coefficient

- The Spearman coefficient is likely the mostly used rank correlation metric
- It is based on the differences between the positions of a same document in two rankings
- Let
  - $s_{1,j}$ be the position of a document d_j in ranking R_1 and
  - $s_{2,j}$ be the position of d_j in ranking R_2

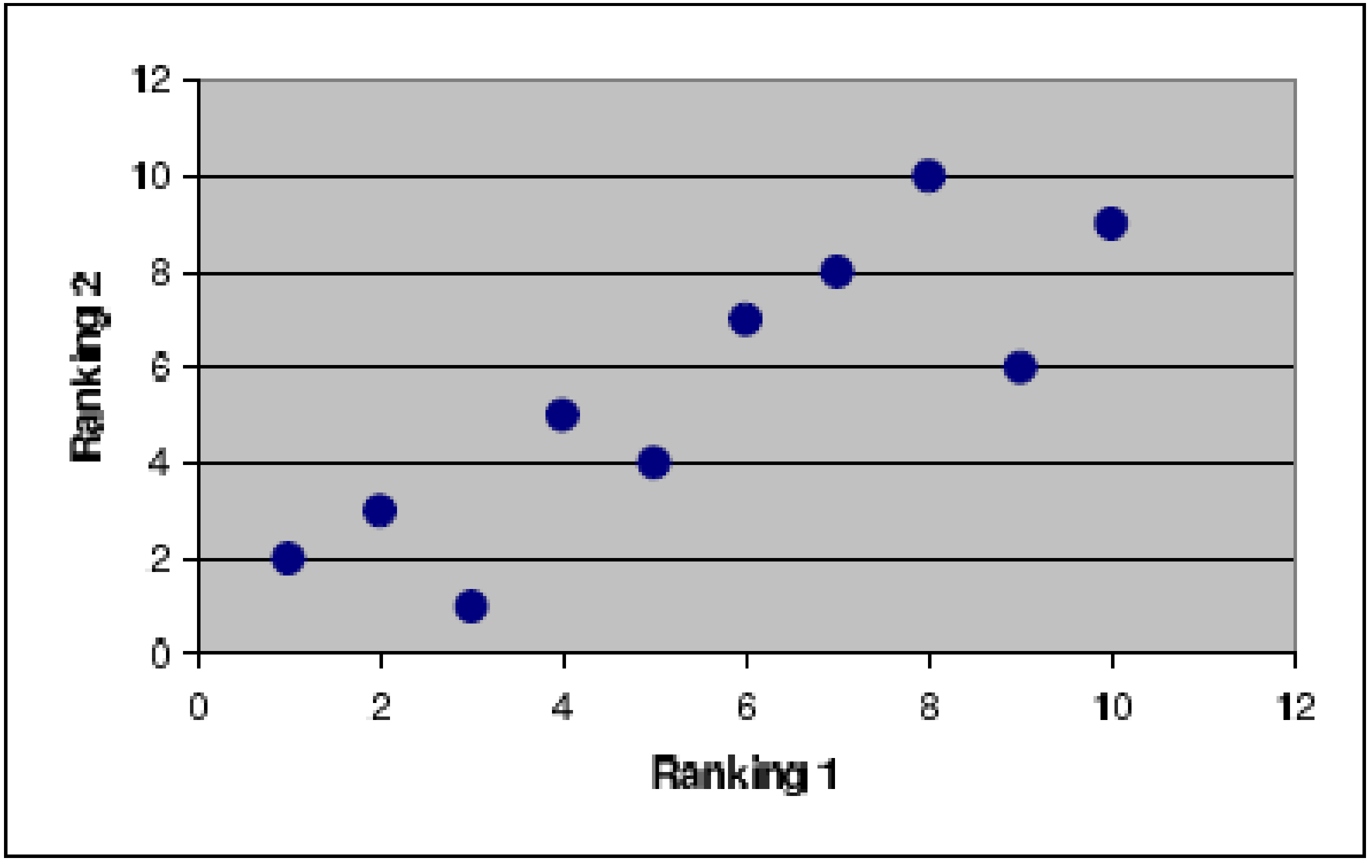
The Spearman coefficient

- Consider 10 example documents retrieved by two distinct rankings R_1 and R_2. Let $s_{1,j}$ and $s_{2,j}$ be the document position in these two rankings, as follows:

documents	$s_{1,j}$	$s_{2,j}$	$s_{i,j} - s_{2,j}$	$(s_{1,j} - s_{2,j})^2$
d_{123}	1	2	-1	1
d_{84}	2	3	-1	1
d_{56}	3	1	+2	4
d_6	4	5	-1	1
d_8	5	4	+1	1
d_9	6	7	-1	1
d_{511}	7	8	-1	1
d_{129}	8	10	-2	4
d_{187}	9	6	+3	9
d_{25}	10	9	+1	1
Sum of Square Distances				24

The Spearman coefficient

- By plotting the rank positions for R_1 and R_2 in a 2-dimensional coordinate system, we observe that there is a strong correlation between the two rankings:



The Spearman coefficient

- To produce a quantitative assessment of this correlation, we sum the squares of the differences for each pair of rankings
- If there are *K* documents ranked, the maximum value for the sum of squares of ranking differences is given by

$$\frac{K \times (K^2 - 1)}{3}$$

- Let K = 10
  - ✓ If the two rankings were in perfect disagreement, then this value is $(10 \times (10^2 - 1))/3$, or 330
  - ✓ On the other hand, if we have a complete agreement the sum is 0

The Spearman coefficient

- Let us consider the fraction

$$\frac{\sum_{j=1}^{K}(s_{1,j}-s_{2,j})^2}{\frac{K\times(K^2-1)}{3}}$$

- Its value is
  - 0 when the two rankings are in perfect agreement
  - +1 when they are in perfect disagreement
- If we multiply the fraction by 2, its value shifts to the range [0,+2]
- If we now subtract the result from 1, the resultant value shifts to the range [−1,+1]

The Spearman coefficient

- This reasoning suggests defining the correlation between the two rankings as follows
- Let $s_{1,j}$ and $s_{2,j}$ be the positions of a document d_j in two rankings R_1 and R_2, respectively
- Define

$$\boxed{S(\mathcal{R}_1, \mathcal{R}_2) = 1 - \frac{6 \times \sum_{j=1}^{K} (s_{1,j} - s_{2,j})^2}{K \times (K^2 - 1)}}$$

where

- ✓ $S(R_1,R_2)$ is the Spearman rank correlation coefficient
- ✓ K indicates the size of the ranked sets

The Spearman coefficient

- For the rankings in Figure below, we have

$$S(\mathcal{R}_1, \mathcal{R}_2) = 1 - \frac{6 \times 24}{10 \times (10^2 - 1)} = 1 - \frac{144}{990} = 0.854$$

documents	$s_{1,j}$	$s_{2,j}$	$s_{i,j} - s_{2,j}$	$(s_{1,j} - s_{2,j})^2$
d_{123}	1	2	-1	1
d_{84}	2	3	-1	1
d_{56}	3	1	+2	4
d_6	4	5	-1	1
d_8	5	4	+1	1
d_9	6	7	-1	1
d_{511}	7	8	-1	1
d_{129}	8	10	-2	4
d_{187}	9	6	+3	9
d_{25}	10	9	+1	1
Sum of Square Distances				24

The Kendall Tau coefficient

- It is difficult to assign an operational interpretation to Spearman coefficient
- One alternative is to use a coefficient that has a natural and intuitive interpretation, as the Kendall Tau coefficient

The Kendall Tau coefficient

- When we think of rank correlations, we think of how two rankings tend to vary in similar ways
- To illustrate, consider two documents d_j and d_k and their positions in the rankings R_1 and R_2
- Further, consider the differences in rank positions for these two documents in each ranking, i.e.

$$s_{1,k} - s_{1,j}$$

$$s_{2,k} - s_{2,j}$$

- If these differences have the same sign, we say that the document pair $[d_k, d_j]$ is **concordant** in both rankings
- If they have different signs, we say that the document pair is **discordant** in the two rankings

The Kendall Tau coefficient

- Consider the top 5 docs in rankings R_1 and R_2

documents	$s_{1,j}$	$s_{2,j}$	$s_{i,j} - s_{2,j}$
d_{123}	1	2	-1
d_{84}	2	3	-1
d_{56}	3	1	+2
d_6	4	5	-1
d_8	5	4	+1

- The ordered document pairs in ranking R_1 are

$$[d_{123}, d_{84}],\quad [d_{123}, d_{56}],\quad [d_{123}, d_6],\quad [d_{123}, d_8],$$
$$[d_{84}, d_{56}],\quad [d_{84}, d_6],\quad [d_{84}, d_8],$$
$$[d_{56}, d_6],\quad [d_{56}, d_8],$$
$$[d_6, d_8]$$

for a total of ½ × 5 × 4, or 10 ordered pairs

The Kendall Tau coefficient

- Repeating the same exercise for the top 5 documents in ranking R_2, we obtain

$$[d_{56}, d_{123}],\quad [d_{56}, d_{84}],\quad [d_{56}, d_8],\quad [d_{56}, d_6],$$
$$[d_{123}, d_{84}],\quad [d_{123}, d_8],\quad [d_{123}, d_6],$$
$$[d_{84}, d_8],\quad [d_{84}, d_6],$$
$$[d_8, d_6]$$

- We compare these two sets of ordered pairs looking for concordant and discordant pairs

The Kendall Tau coefficient

- Let us mark with a C the concordant pairs and with a D the discordant pairs
- For ranking R_1, we have

$$C, D, C, C,$$
$$D, C, C,$$
$$C, C,$$
$$D$$

- For ranking R_2, we have

$$D, D, C, C,$$
$$C, C, C,$$
$$C, C,$$
$$D$$

The Kendall Tau coefficient

- That is, a total of 20, i.e., K(K − 1), ordered pairs are produced jointly by the two rankings
- Among these, 14 pairs are concordant and 6 pairs are discordant
- The Kendall Tau coefficient is defined as

$$\tau(\mathcal{R}_1, \mathcal{R}_2) = P(\mathcal{R}_1 = \mathcal{R}_2) - P(\mathcal{R}_1 \neq \mathcal{R}_2)$$

- In our example

$$\begin{aligned} \tau(\mathcal{R}_1, \mathcal{R}_2) &= \frac{14}{20} - \frac{6}{20} \\ &= 0.4 \end{aligned}$$

The Kendall Tau coefficient

- Let,
  - $\Delta(\mathcal{R}_1, \mathcal{R}_2)$: number of discordant document pairs in $\mathcal{R}_1$ and $\mathcal{R}_2$
  - $K(K-1) - \Delta(\mathcal{R}_1, \mathcal{R}_2)$: number of concordant document pairs in $\mathcal{R}_1$ and $\mathcal{R}_2$
- Then,

$$
\begin{aligned}
P(\mathcal{R}_1 = \mathcal{R}_2) &= \frac{K(K-1) - \Delta(\mathcal{R}_1, \mathcal{R}_2)}{K(K-1)} \\
P(\mathcal{R}_1 \neq \mathcal{R}_2) &= \frac{\Delta(\mathcal{R}_1, \mathcal{R}_2)}{K(K-1)}
\end{aligned}
$$

which yields $\boxed{\tau(\mathcal{R}_1, \mathcal{R}_2) = 1 - \frac{2 \times \Delta(\mathcal{R}_1, \mathcal{R}_2)}{K(K-1)}}$

The Kendall Tau coefficient

- For the case of our previous example, we have
  - $\Delta(\mathcal{R}_1, \mathcal{R}_2) = 6$
  - $K = 5$
  - Thus,

$$\tau(\mathcal{R}_1, \mathcal{R}_2) = 1 - \frac{2 \times 6}{5(5-1)} = 0.4$$

    as before
- The Kendall Tau coefficient is defined only for rankings over a same set of elements
- Most important, it has a simpler algebraic structure than the Spearman coefficient

References

Baeza-Yates, R.A., Ribeiro-Neto, B.A., "Modern Information Retrieval", ACM Press /Addison-Wesley, 1999.

Barker, F., Wyatt, B., Veal, D., Service, U. K. C. I., 1974. Retrieval Experiments Based on Chemical Abstracts Condensates. United Kingdom Chemical Information Service.

Croft, W.B., Metzler, D., Strohman, T., "Search Engines: Information Retrieval in Practice", Pearson Education, 2010.

Cleverdon, C. W., 1960. Report on the first stage of an investigation onto the comparative efficiency of indexing systems. Tech. rep., The College of Aeronautics, Cranfield, England.

Cleverdon, C. W., 1970. The effect of variations in relevance assessments in comparative experimental tests of index languages. Tech. rep., The College of Aeronautics, Cranfield, England.

Cleverdon, C. W., 1991. The significance of the Cranfield tests on index languages. In Proceed. of the 14th annual int'l ACM SIGIR conference on Research and development in information retrieval. SIGIR '91.

Cleverdon, C. W., Mills, J., Keen, M., 1966. Factors determining the performance of indexing systems. Aslib Cranfield Research Project Cranfield England.

Järvelin, K., Kekäläinen, J., 2000. IR evaluation methods for retrieving highly relevant documents. In Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 41–48.

Järvelin, K., Kekäläinen, J., 2002. Cumulated gain-based evaluation of IR techniques. ACM Trans. Inf. Syst.

Kantor, P. B., Voorhees, E. M., 2000. The TREC-5 confusion track: Comparing retrieval methods for scanned text. Information Retrieval, 2(2/3): 165-176.

Keen, E. M., Digger, J. A., 1972. Report of an information Science Index Languages Test. Aberystwyth, Department of Information Retrieval Studies, College of Librarianship Wales.

Kent, A., Berry, M. M., Luehrs, Perry, J. W., 1955. Machine literature searching VIII. Operational criteria for designing information retrieval systems. American Documentation.

Lancaster, F., 1968. Evaluation of the MEDLARS demand search service: by F. W. Lancaster. U.S. Dept. of Health, Education, and Welfare, Public Health Service.

Salton, G., & McGill, M. J. "Introduction to Modern Information Retrieval", McGraw-Hill, 1983.

vanRijsbergen, C.J., "Information Retrieval", Butterworth & Co., Boston, MA, 1979.