Australian National University

COMP6490
Document Analysis
Session 2 - 2021

Evaluation of IR Systems

Research School of Computer Science, ANU

So far..

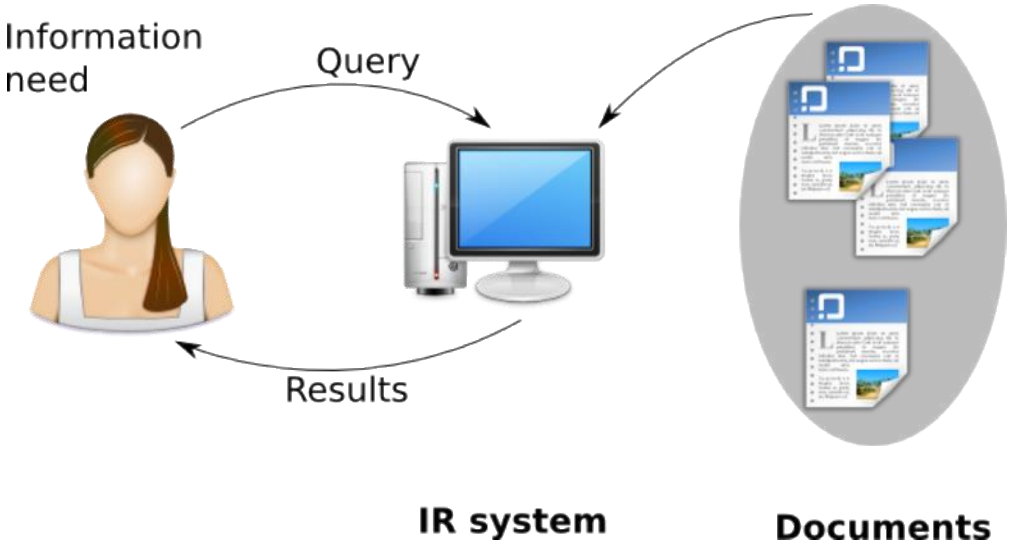
1. Boolean Retrieval
2. Ranked Retrieval

Table of Contents

- Evaluation of IR systems
  - Purpose of evaluation
  - Test collection
  - Evaluation of unranked retrieval sets
  - Evaluation of ranked retrieval sets

Why do we need evaluation?

- To build IR system satisfying user's information need



- Given multiple candidate systems, what would be the best?

What do we want to evaluate?

System Efficiency

- Speed
- Storage
- Memory
- Cost

System Effectiveness

- Quality of search result
- Does it find what I'm looking for
- Does it return lots of junk?

We will focus on evaluating system **effectiveness**.

To improve system effectiveness

- IR system design:
  - Which tokenizer? which stemmer?
  - Boolean or ranked?
  - Which scoring method?
  - *tf-idf* or *wf-idf*?
  - Length normalization or not?
- What will be the best choice?

Table of Contents

For example

- **Test collection** is a collection of relevance judgment on (query, document) pairs.
- Query 1
  - Doc 1: **relevant**
  - Doc 2: irrelevant
  - Doc 3: irrelevant
  - Doc 4: **relevant**
  - Doc 5: irrelevant
- Query 2
  - Doc 1: irrelevant
  - Doc 2: irrelevant
  - Doc 3: **relevant**
  - Doc 4: irrelevant
  - Doc 5: **relevant**
- Relevancy information next to the document is called **ground truth**. Judged by some expert assessors.

Three Components of Test Collections

1. A collection of documents
2. A test suite of information needs, expressible as queries
3. A set of relevance judgment; a binary assessment of either relevant or irrelevant for each query-document pair

Relevance Judgment

- **Relevance** is assessed relative to an information need, *not a query*.
  - For example, if our information need is:
    - *Information on whether drinking red wine is more effective at reducing your risk of heart attacks than white wine*
    - Candidate query: wine red white heart attack effective
- A document is relevant if it addresses the information need.
- It is not important that the document contains all query terms or not.

Standard Test Collections

Collection name	date	docs	size
Cranfield II	1963	1400	1.6MB
MEDLARS	1973	450	
Time	1973	425	1.5MB
.GOV2	2004	25M	426GB
Clueweb09	2009	1B	25TB

Table: Examples of test collections

And now also blogs, Twitter, legal, patents, chemical, genomic, ...

Build Large Test Collection

- Recent collections do not have relevance judgments on all possible pairs of (query, document).
  - It is just impossible!
  - Given a query, try multiple IR systems and obtain a set of candidate documents
  - Multiple judges assess the relevancy of a candidate doc given information need (Chapter 8.5)

Table of Contents

- Evaluation of IR systems
  - Purpose of evaluation
  - Test collection
  - Evaluation of unranked retrieval sets
  - Evaluation of ranked retrieval sets

Evaluation Retrieval Result

- Two possible ways of evaluating test collections:
  - Evaluation of unranked retrieval sets (Boolean retrieval)
    - Rank of document is not important
    - Retrieved (returned) documents vs. Not retrieved documents
  - Evaluation of ranked retrieval sets
    - Consider the rank of retrieved documents
    - Relevant documents on high ranks

Evaluation of unranked retrieval sets

- Scenario:
  - Say we have 10 documents in our system.
  - Given query q, the system returns 4 documents:
    - The system judges these 4 documents are relevant given query.

Retrieved (Returned) docs:

- Doc 2: **relevant** (ground truth)
- Doc 4: irrelevant
- Doc 5: irrelevant
- Doc 7: **relevant**

Not retrieved docs:

- Doc 1: irrelevant
- Doc 3: irrelevant
- Doc 6: irrelevant
- Doc 8: **relevant**
- Doc 9: irrelevant
- Doc 10: irrelevant

How can we evaluate the performance of this system?

Contingency Table

- Contingency table: a summary table of retrieval result

Table: Contingency table

	Relevant	Not relevant
Retrieved	true positive (tp)	false positive (fp)
Not retrieved	false negative (fn)	true negative (tn)

- tp: Number of relevant documents returned by system
- fp: Number of irrelevant documents returned by system
- fn: Number of relevant documents not returned by system
- tn: Number of irrelevant documents not returned by system

Precision, Recall, and Accuracy

- Precision: fraction of retrieved documents that are relevant

$$Precision = \frac{\#\text{of relevant docs retrieved}}{\#\text{of retrieved docs}} = \frac{tp}{tp + fp}$$

- Recall: fraction of relevant documents that are retrieved

$$Recall = \frac{\#\text{of relevant docs retrieved}}{\#\text{of relevant docs}} = \frac{tp}{tp + fn}$$

- Accuracy: fraction of relevant documents that are correct

$$Accuracy = \frac{\#\text{of correctly classified docs}}{\#\text{of total docs}} = \frac{tp + tn}{tp + tn + fp + fn}$$

Precision and Recall

- From the previous example:

Table: 10 document example

	Rel.	Not rel.
Retrieved	2	2
Not retrieved	1	5

- Precision $= \frac{2}{2+2} = 0.50$
- Recall $= \frac{2}{2+1} = 0.66$
- System with high precision and recall is always preferable.

Accuracy is not for IR

- Assume we have 100 documents, and only 1 document is relevant given a certain query *q*.

Table: System 1

	Rel.	Not rel.
Retrieved	0	0
Not retrieved	1	99

Table: System 2

	Rel.	Not rel.
Retrieved	1	4
Not retrieved	0	95

- Accuracy of System 1: 0.99
- Accuracy of System 2: 0.96
- System 1 performs better in terms of accuracy, but..

F-Measure

- *F-measure*: a single measure that trades off precision versus recall

$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}}, \qquad \alpha \in [0, 1]$$

which is weighted harmonic mean of precision(P) and recall(R) .

  - α > 0.5: emphasise precision, e.g., (α = 1)) Precision
  - α < 0.5: emphasise recall, e.g., (α = 0)) Recall
- *F1-measure*: a harmonic mean of precision and recall (α = 0.5)

$$F_1 = \frac{2PR}{P+R}$$

Table of Contents

Evaluation of ranked retrieval sets

- Scenario: With the same pair of documents (10 docs) and query, an IR system generates ranked results as follows:

Rank of System 1:

1. Doc 2: **relevant** (ground truth)
2. Doc 4: irrelevant
3. Doc 7: **relevant**
4. Doc 5: irrelevant
5. Doc 1: irrelevant
6. Doc 8: **relevant**
7. Doc 3: irrelevant
8. Doc 9: irrelevant
9. Doc 10: irrelevant
10. Doc 6: irrelevant

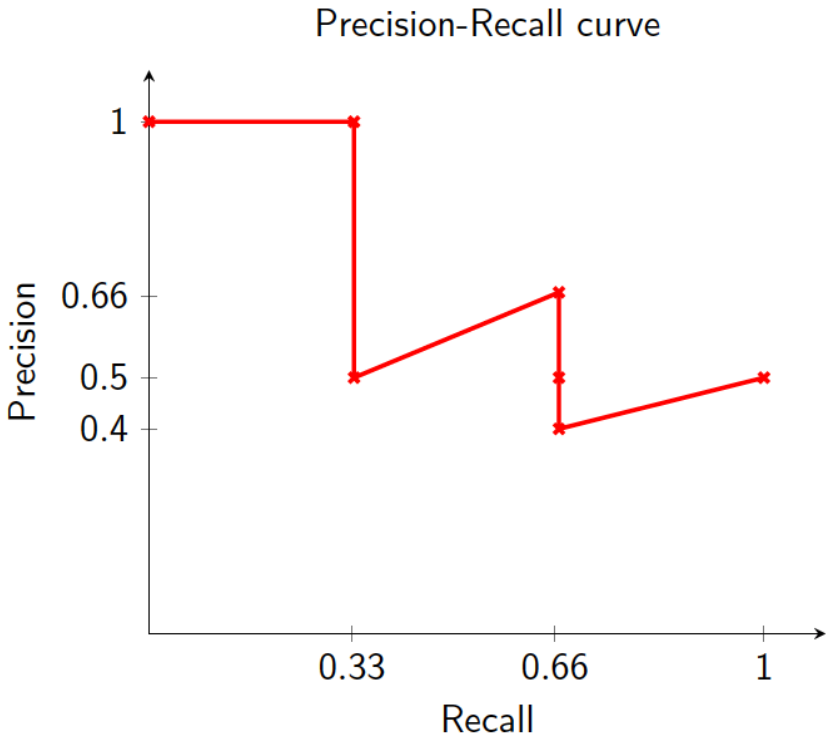
- Unlike the previous example, the system retrieved all documents in our collection.
- Precision & Recall cannot be directly applied in this case.
- Need a metric to measure the performance of ranked list!

How can we quantify the performance of this result?

Precision-Recall Curve

- What are the precision and recall when *top k* docs retrieved?

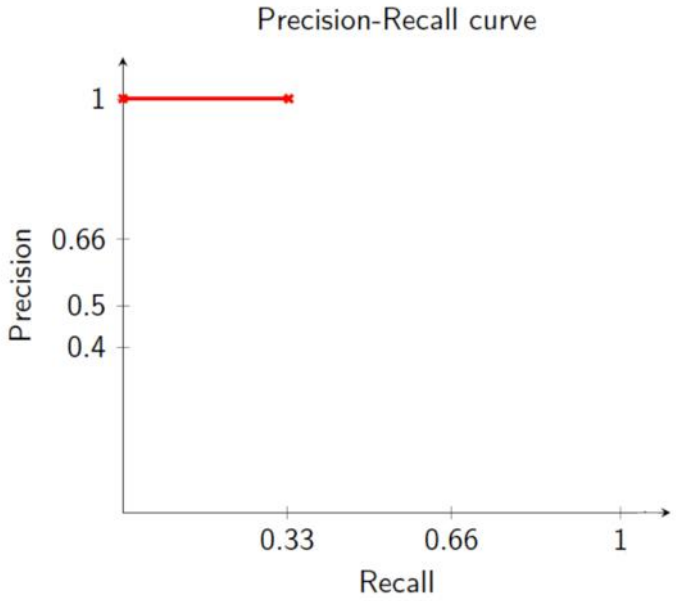
1. Doc 2: **relevant**
2. Doc 4: irrelevant
3. Doc 7: **relevant**
4. Doc 5: irrelevant
5. Doc 1: irrelevant
6. Doc 8: **relevant**
7. Doc 3: irrelevant
8. Doc 9: irrelevant
9. Doc 10: irrelevant
10. Doc 6: irrelevant



Precision-Recall Curve

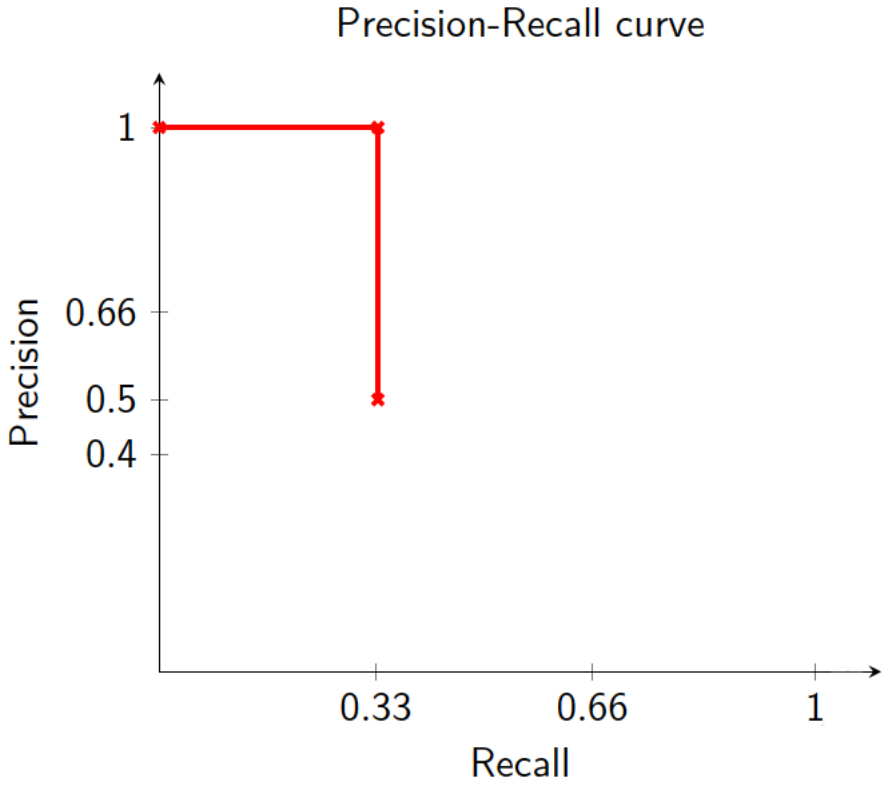
- Compute recall and precision at each *rank k* w.r.t. *top k* docs
- Plot (recall, precision) points until recall reaches 1

1. Doc 2: **relevant** (1/3, 1)



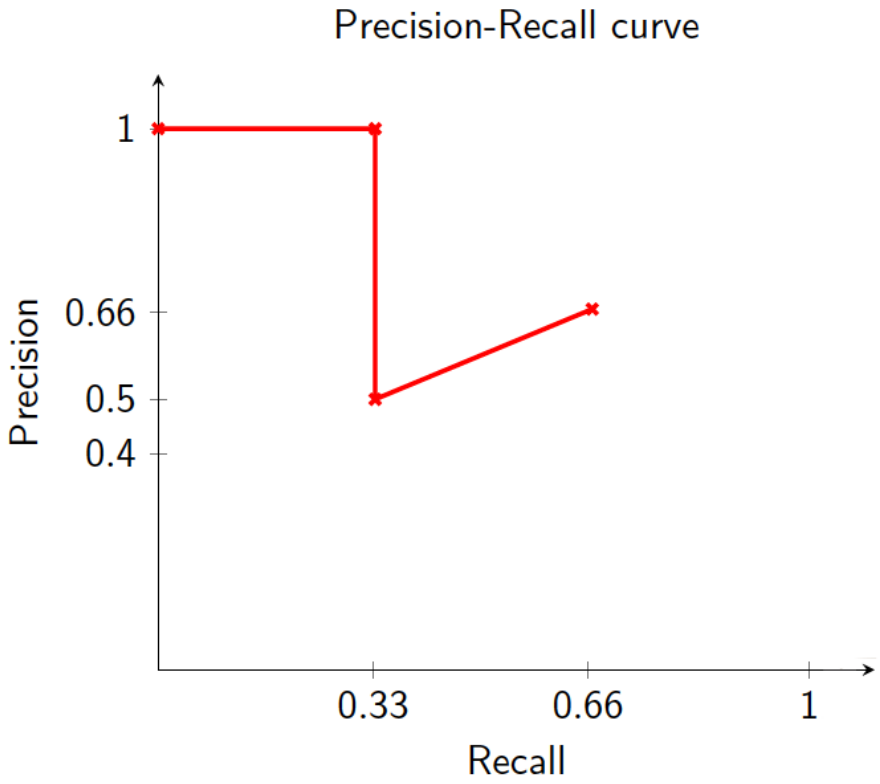
Precision-Recall Curve

1. Doc 2: **relevant** (1/3, 1)
2. Doc 4: irrelevant (1/3, 1/2)



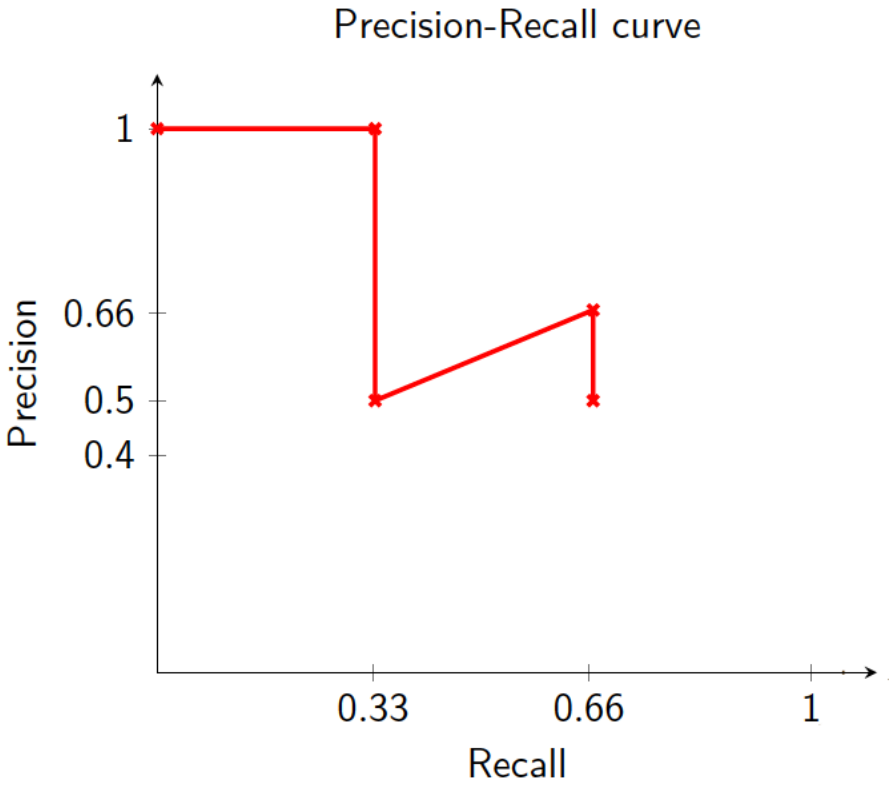
Precision-Recall Curve

1. Doc 2: **relevant** (1/3, 1)
2. Doc 4: irrelevant (1/3, 1/2)
3. Doc 7: **relevant** (2/3, 2/3)



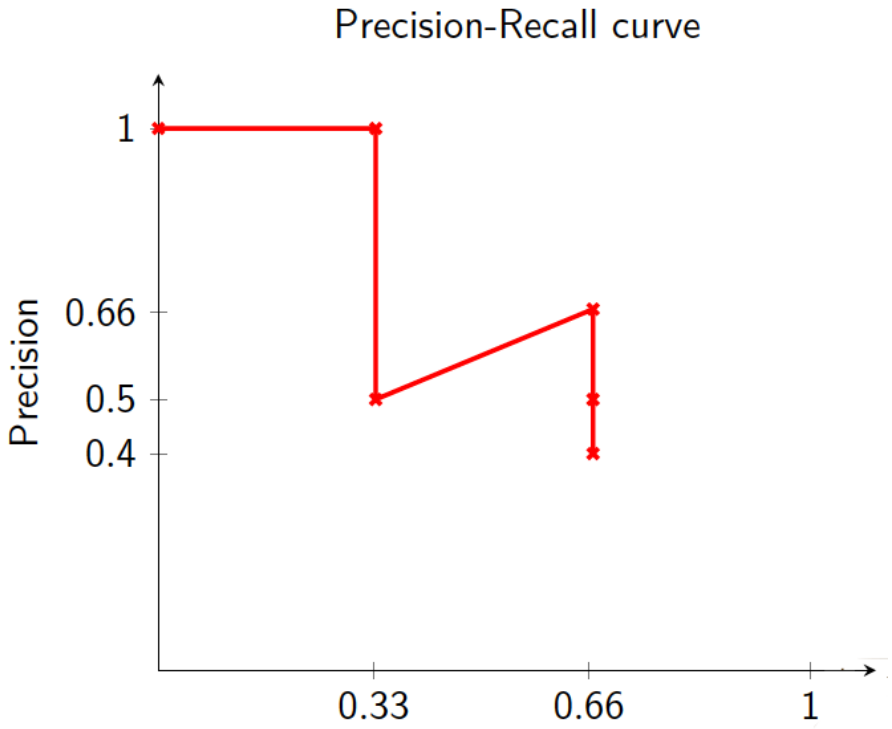
Precision-Recall Curve

1. Doc 2: **relevant** (1/3, 1)
2. Doc 4: irrelevant (1/3, 1/2)
3. Doc 7: **relevant** (2/3, 2/3)
4. Doc 5: irrelevant (2/3, 2/4)



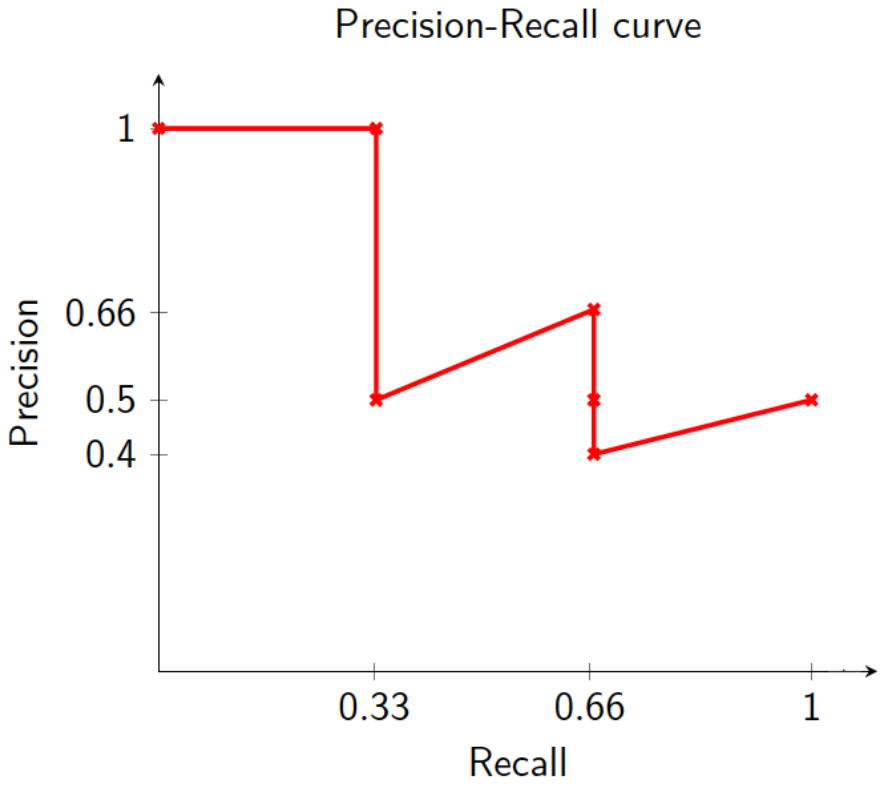
Precision-Recall Curve

1. Doc 2: **relevant** (1/3, 1)
2. Doc 4: irrelevant (1/3, 1/2)
3. Doc 7: **relevant** (2/3, 2/3)
4. Doc 5: irrelevant (2/3, 2/4)
5. Doc 1: irrelevant (2/3, 2/5)



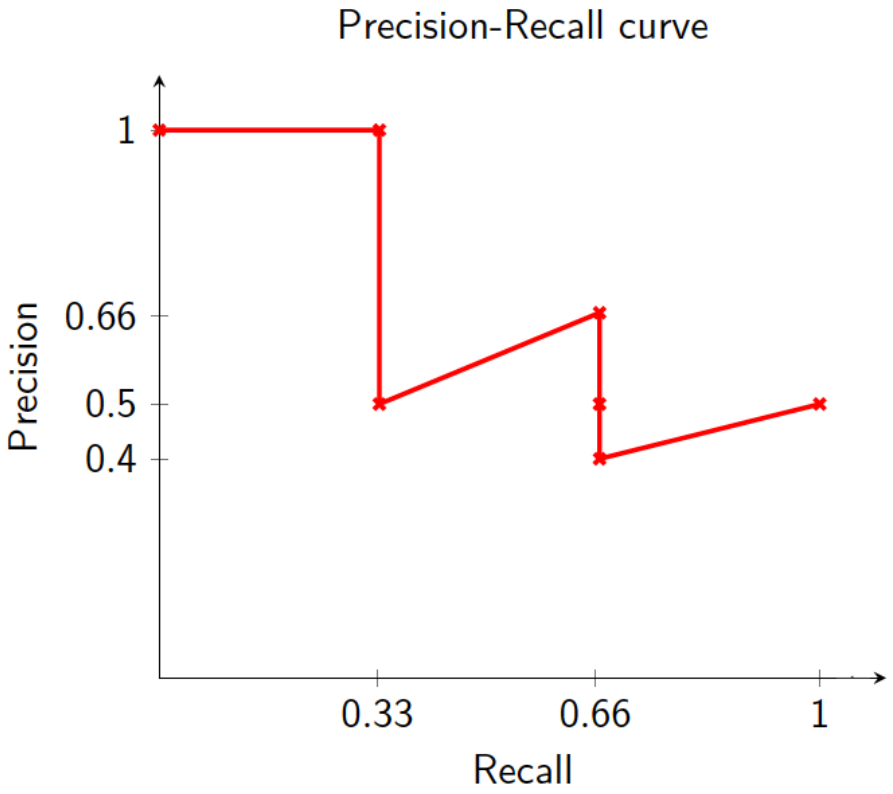
Precision-Recall Curve

1. Doc 2: **relevant** (1/3, 1)
2. Doc 4: irrelevant (1/3, 1/2)
3. Doc 7: **relevant** (2/3, 2/3)
4. Doc 5: irrelevant (2/3, 2/4)
5. Doc 1: irrelevant (2/3, 2/5)
6. Doc 8: **relevant** (1, 3/6)



Precision-Recall Curve

1. Doc 2: **relevant** (1/3, 1)
2. Doc 4: irrelevant (1/3, 1/2)
3. Doc 7: **relevant** (2/3, 2/3)
4. Doc 5: irrelevant (2/3, 2/4)
5. Doc 1: irrelevant (2/3, 2/5)
6. Doc 8: **relevant** (1, 3/6)
7. Doc 3: irrelevant
8. Doc 9: irrelevant
9. Doc 10: irrelevant
10. Doc 6: irrelevant

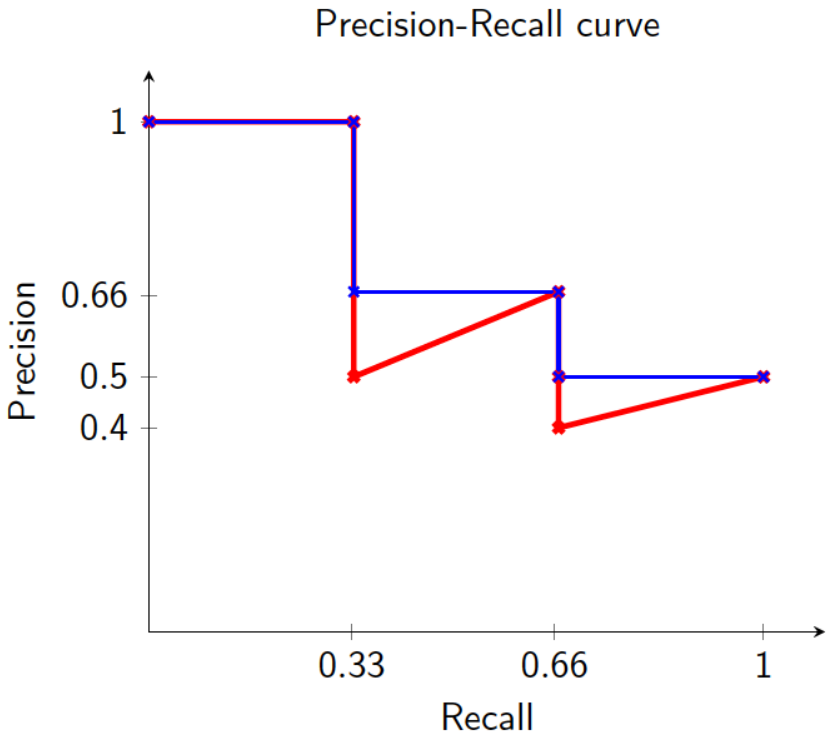


Interpolated Precision-Recall Curve

- Zigzag line is bit hard to interpret.
- Interpolate precision: fill furrows

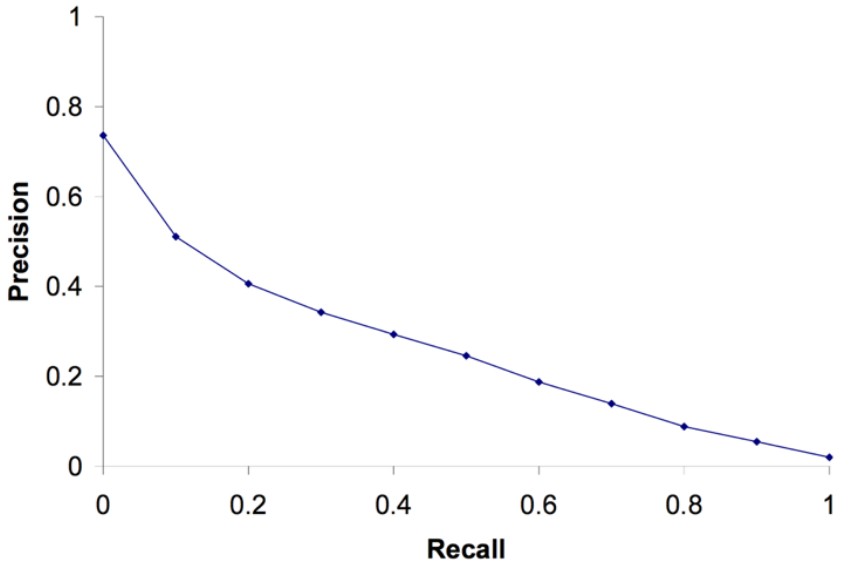
Recall	Interp. Prec.
0.0	1.00
0.1	1.00
0.2	1.00
0.3	1.00
0.4	0.66
0.5	0.66
0.6	0.66
0.7	0.50
0.8	0.50
0.9	0.50
1.0	0.50

Table: 11-point interpolated precision.



Interpolated Average Precision

- For system evaluation
  - 11-point interpolated precision is averaged across all queries in test collection
  - A perfect system will have a straight line from (0,1) to (1,1)



Mean Reciprocal Rank (MRR)

- MRR = Averaged inverse rank of the first correct answer.

Query	Ranked by System	Relevant docs	Rank	Reciprocal rank
q1	doc3, doc2, **doc1**	doc1	3	1/3
q2	doc2, **doc3**, **doc1**	doc3, doc1	2	1/2
q3	**doc1**, doc3, doc2	doc1	1	1

Table: MRR example (from Wikipedia)

$$\text{Mean Reciprocal Rank} = \frac{1}{3}(\frac{1}{3} + \frac{1}{2} + 1) = 0.61$$

Other ranking measures

- Mean Average Precision
  - roughly the average area under the precision-recall curve
- Precision at K
  - Average precision at top k documents
- Recall at K
  - Average recall at top k documents
- Receiver Operating Characteristics (ROC) curve
- Normalized Discounted Cumulative Gain (NDCG)

Summary

- Evaluation of IR systems
  - Purpose of evaluation
  - Test collection
  - Evaluation of unranked retrieval sets
  - Evaluation of ranked retrieval sets

References

- Some lecture slides are from:
  Pandu Nayak and Prabhakar Raghavan, CS276
  Information Retrieval and Web Search, Stanford University