CS172 INFORMATION RETRIEVAL

Evaluation Topics Overview (Chapter 8)

- Will cover this topic in more depth later

Evaluating Ranking

- We examined various methods for ranking document, but how do we evaluate the ranking methods?

- Evaluation:

  - **Precision:** Fraction of returned documents that are relevant
  - **Recall:** Fraction of relevant documents that are returned
  - Efficiency

TREC

- The Text **REtrieval** Conference (**TREC**) is an ongoing series of workshops focusing on a list of different **information retrieval** (**IR**) research areas, or tracks.

- Publish datasets (documents and queries) with labeled ranking for each document-query pair

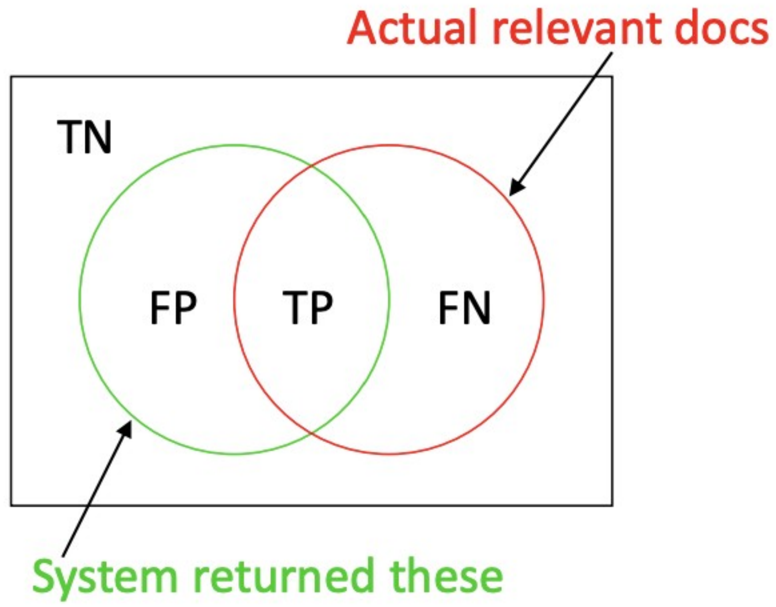
- Host competitions in Information Retrieval
  - Thats how we got BM25 algorithm

Measuring Performance

- **Precision**
  - Proportion of retrieved set that are in fact relevant
  - Institution: How much junk are you giving to the user?
  - Computed as $\frac{TP}{TP+FP}$

- **Recall**
  - Proportion of target items that are selected
  - Institution: How much of the good stuff did we miss?
  - Computed as $\frac{TP}{TP+FN}$

- **TN / True Negative:** case was negative and predicted negative
- **TP / True Positive:** case was positive and predicted positive
- **FN / False Negative:** case was positive but predicted negative
- **FP /False Positive:** case was negative but predicted positive



Retrieved?

Relevant?

	YES	NO
YES	TP	FN
NO	FP	TN

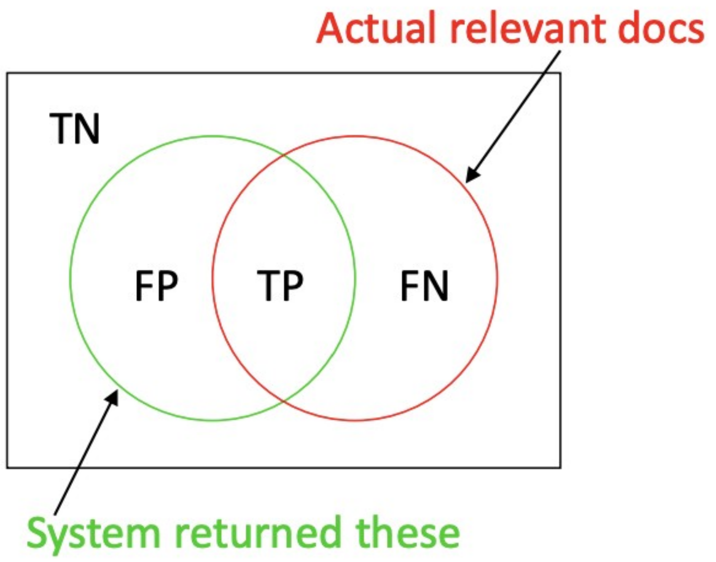
Contingency table

Why not accuracy?

- We can think of retrieval as 'classification' task
  - Hence consider accuracy as a measure

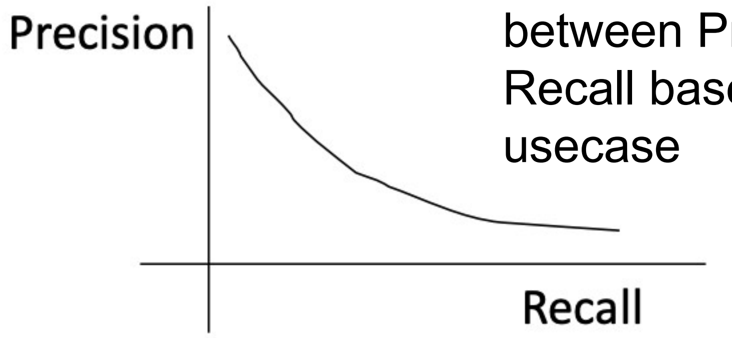
- Accuracy
  - Computed as $\frac{TP+TN}{N}$

- But in this case, Accuracy is meaningless
  - In IR, accuracy is 99.99% for any search algorithm
    - For any query, almost all documents are non-relevant
    - Often the best strategy is to retrieve nothing



Measuring Performance

- Trade-off
  - If you recall everything, then you are generate result that are not accurate, hence lowering precision.
  - If precision is high, obviously recall will be low.



Choose best balance between Precision and Recall based on your usecase

<u>What if we maximize Recall?</u>
 - unlikely user will keep browsing through each and every product … they will jump to a different search engine

<u>What if Precision is high?</u>
- Too few results

Example Exercise

	Predicted Negative	Predicted Positive
Negative Cases	TN: 976	FP: 14
Positive Cases	FN: 4	TP: 6

- What is the accuracy?
  - (976+6)/1000 = 98.2%
- What is precision?
  - 6/20 = 30%
- What is recall?
  - 6/10 = 60%

Evaluation: TREC

- How do you evaluate information retrieval algorithms?
- Need prior relevance judgements
- TREC: Text Retrieval Competition
  - Given:
    - Documents
    - A set of queries: For each query, prior relevance judgements

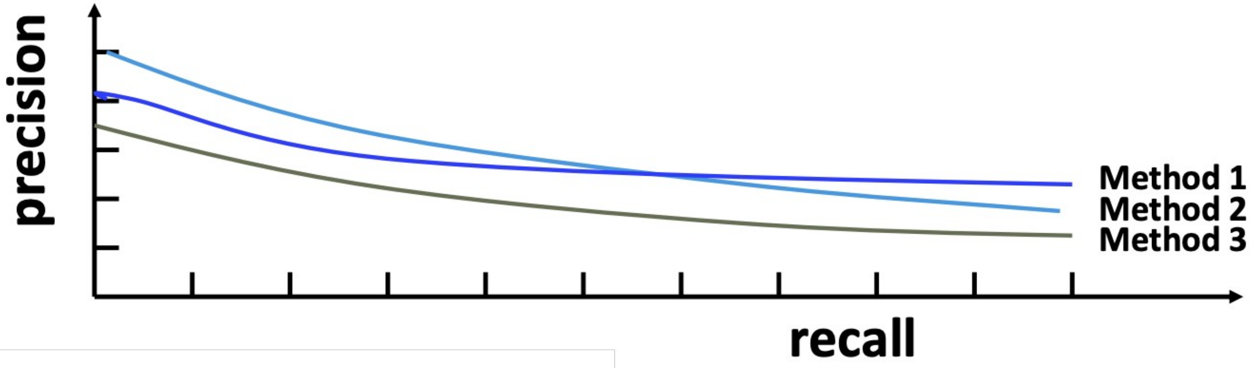
  - Judgement:
    - For each query:
      - Documents are judged in isolation from other possibly relevant documents that have been shown
        - Mostly because the potential subsets of documents already shown can be exponential; too many relevance judgements
      - Rank the systems based on their precision recall on the corpus of queries

    - In practice, search engine maintains logs to record click-through-rate
      - Will discuss in chapter 8

Precision-Recall Curves

- Assuming there the 3 methods and we are evaluating their retrieval effectiveness

- A large number of queries are used and their average precision-recall curve is plotted below



- Methods 1 and 2 are better than method 3
- Method 1 is better than method 2 for higher recalls

Combining Precision and Recall

- We consider a weighted summation of precision and recall into a single quantity
  - F-measure summarizes effectiveness in a single number

- What is the best way to combine?
  - Arithmetic mean
    - Will be affected more by values that are unusually large (outliers).
    - Ex. If recall is 1.0 and precision is 0, the arithmetic mean is 0.5
  - Harmonic mean
    - harmonic mean emphasizes the importance of small values
    - EX. If recall is 1.0 and precision is 0, the harmonic mean is close to 0

$$f_\beta = \frac{(\beta^2 + 1)pr}{\beta^2 p + r}$$

β – relative importance of recall and precision

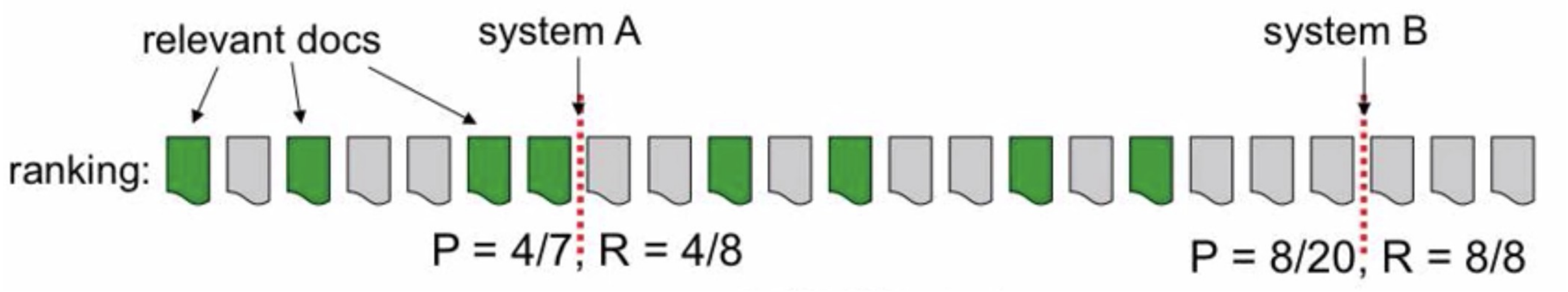
β = 1, gives f $= \frac{2pr}{p+r}$

Heavily penalizes small values of P and R

Comparing Recall/Precision

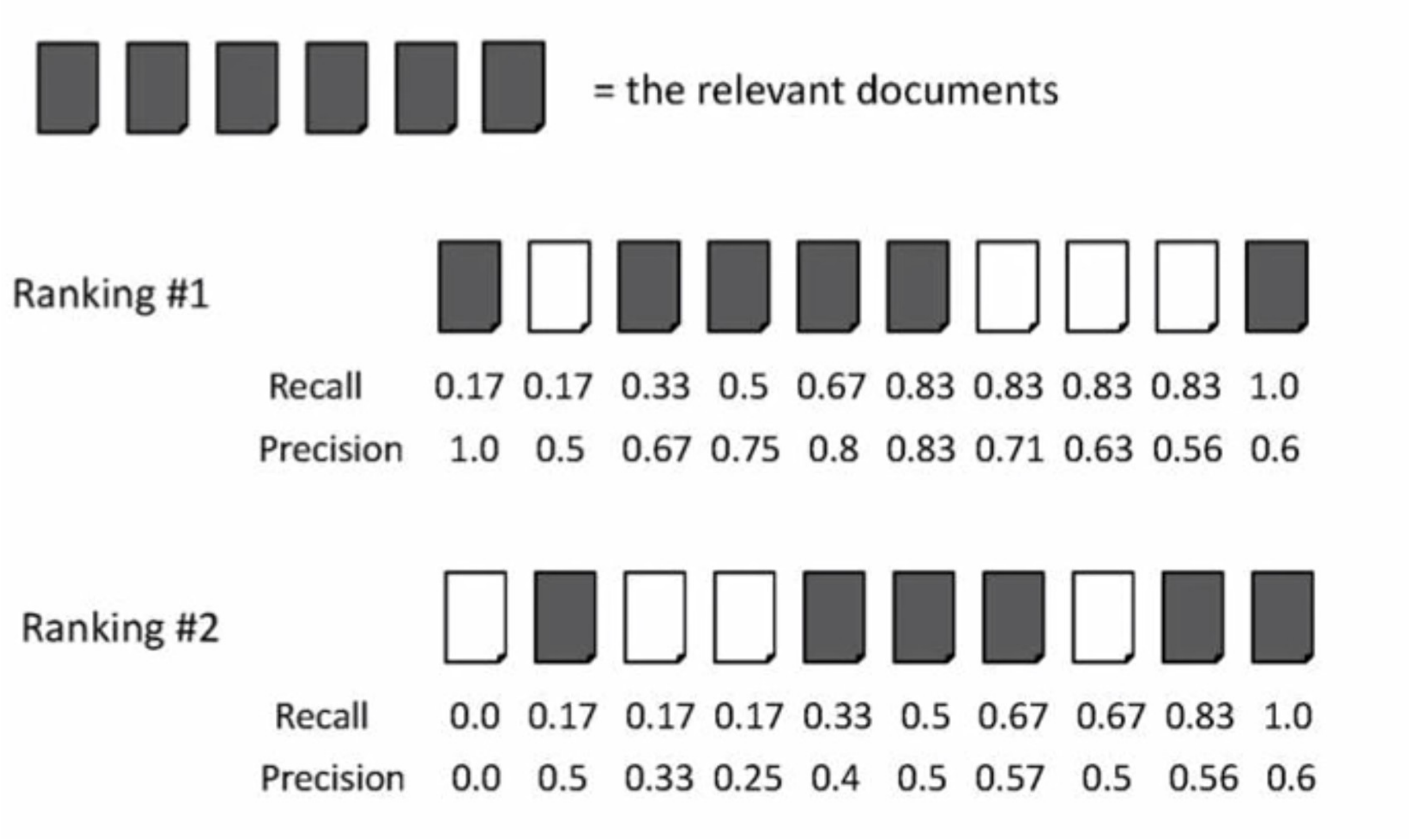
- Which of the following is a better system?
  - System A: Recall = 50%, Prevision 57%, F1=53%
  - System B: Recall = 100%, Precision=40%, F1=57%

- Could be the same exact system!!!
  - Using different threshold settings
  - R/P, F1 comparisons can be meaningless
  - More informative to compare ranking against ranking

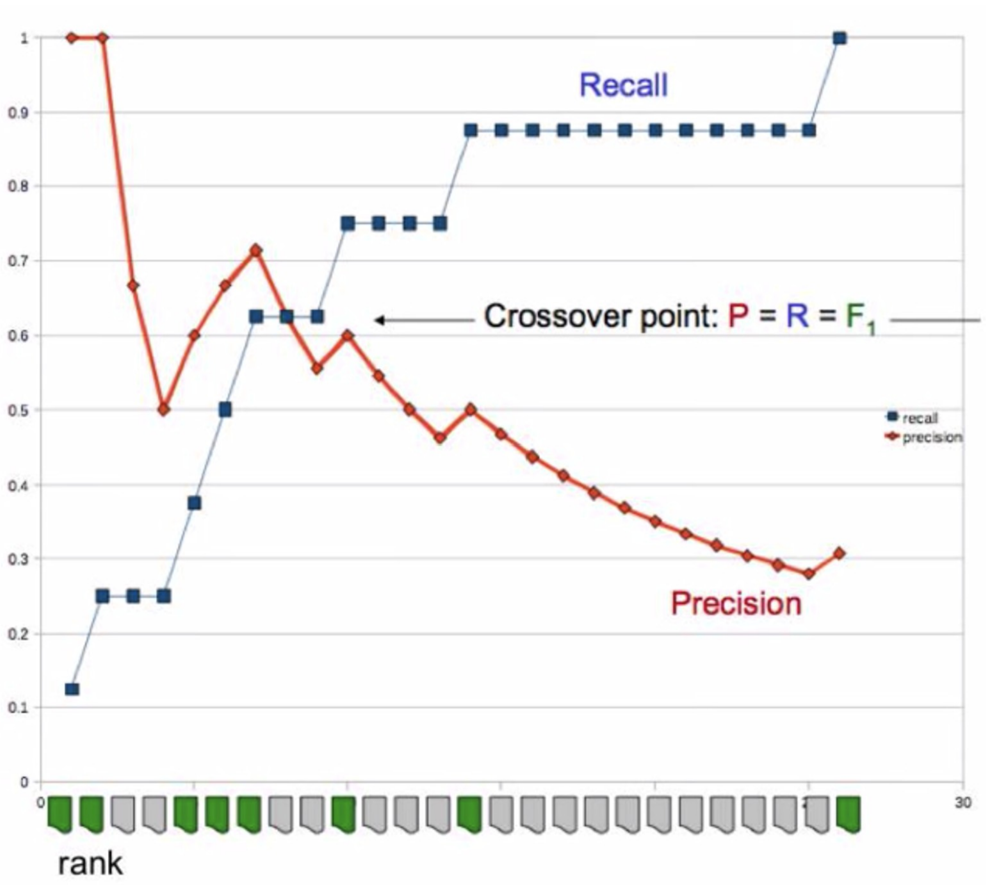


Recall / Precision and ranking

- Search engine produces a ranking, not a set
  - Can compute recall, precision at every rank

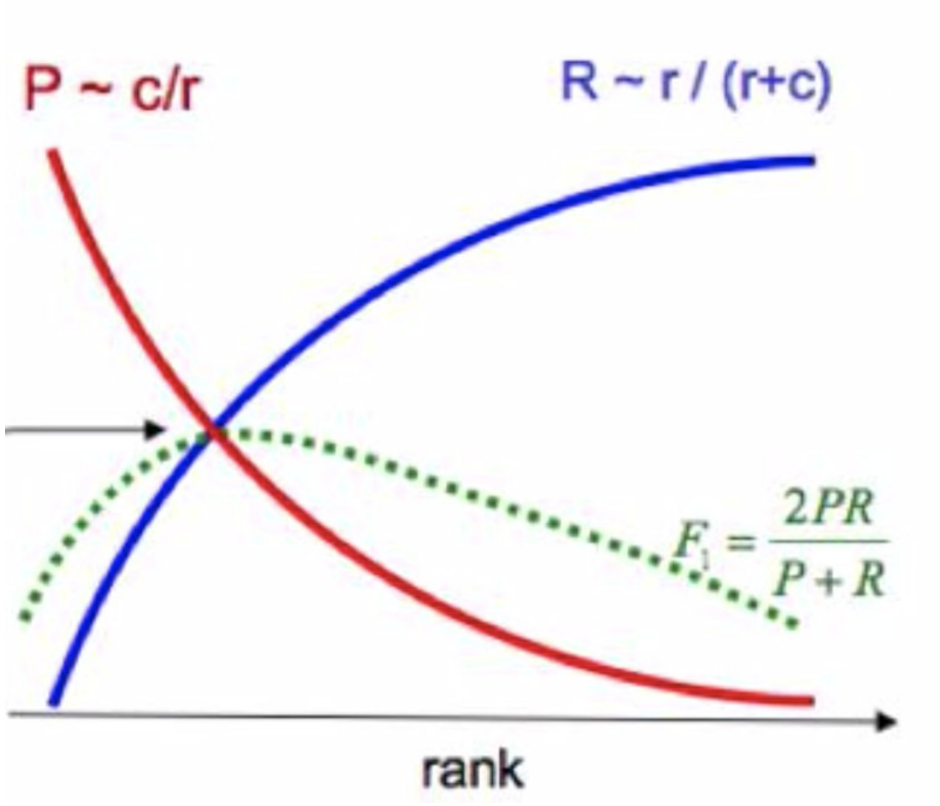


Recall / Precision and ranking (Cont.)



F1

Max F_1



Mean Average Precision

- Sometimes need a single number metric
  - Comparing many systems, tuning parameters

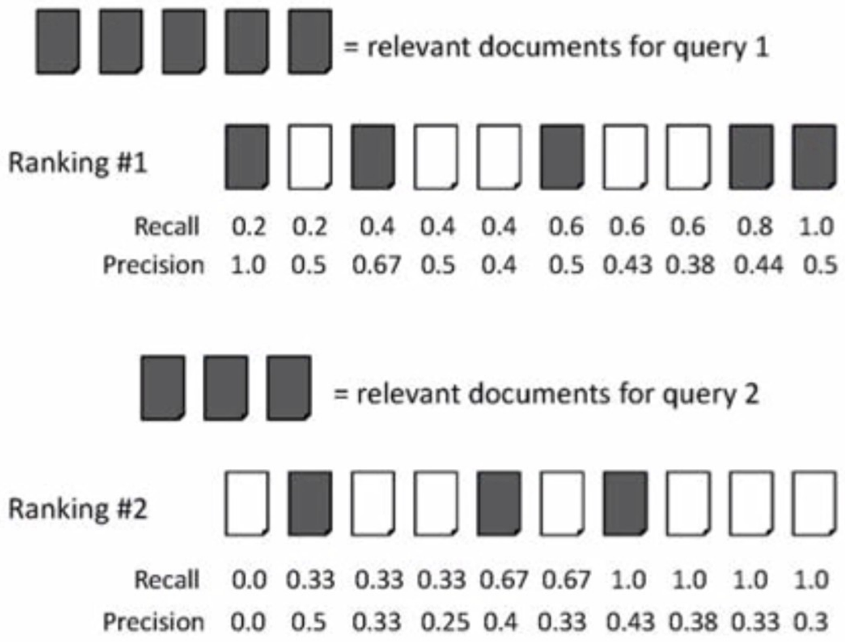
- Mean Average Precision (MAP)
  - Most frequently used measure in research papers
  - Average precision values at ranks of relevant docs
    - Assumes user wants to find many relevant docs
    - Biased toward top of the ranking (rank1=2*rank2)

  - Take the mean of AVE. P values across queries

  - GMAP: geometric average go combine Ave. P.
    - Heavily penalized if any query has low performance

<u>Takeaways</u>
- Looks at the entire ranking (not just a fraction)
- Assigns higher weight for documents ranked higher (or first)

Mean Average Precision: Example



$$average\ precision\ query\ 1 = (1.0 + 0.67 + 0.5 + 0.44 + 0.5)/5 = 0.62$$
$$average\ precision\ query\ 2 = (0.5 + 0.4 + 0.43)/3 = 0.44$$

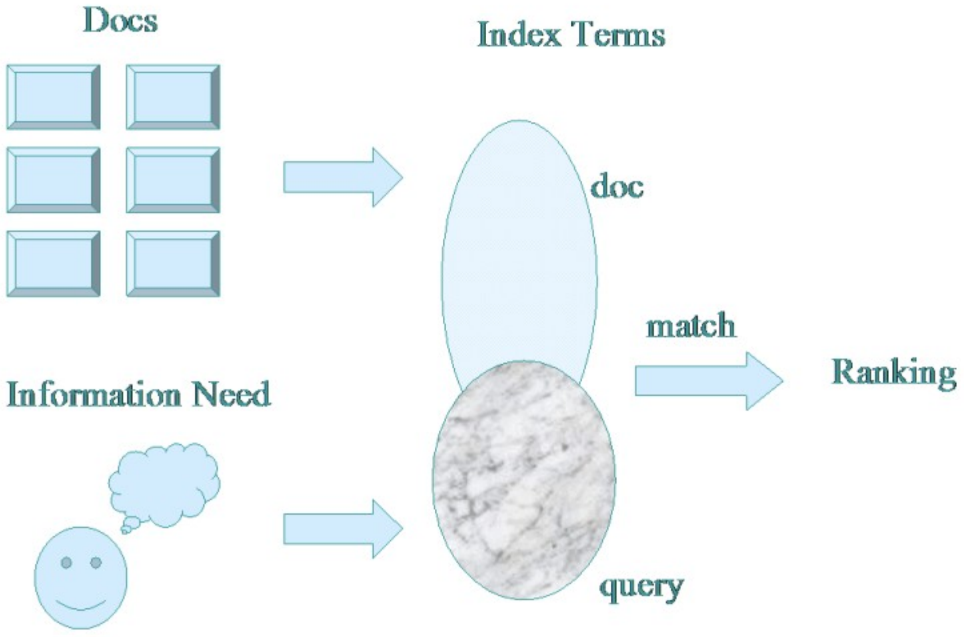
$$mean\ average\ precision = (0.62 + 0.44)/2 = 0.53$$

Relevance: The most overloaded word in IR

- We want to rank and return documents that are relevant to the user's query
  - Easy if each document has a relevance number R

- What does relevance depend on?
  - The document ***d***
  - The query ***q***
  - The user ***u***
  - The other documents already shown $\{d_1, d_2, \ldots, d_k\}$

$$R (d \mid Q, U, \{d_1 d_2 \ldots d_k\})$$



How to compute relevance?

- Specify up front
  - Too hard—one for each query, user and shown results combination

- Learn
  - Active (utility elicitation)
  - Passive (learn from what the user does)

- Make up the users’ mind
  - What you are “really” looking for is.. (used car sales people)

- Combination of the above

- Assume (impose) a relevance model
  - Based on “default” models of d and U.