

Web Information Retrieval

Lecture 8

Evaluation in information retrieval

Recap of the last lecture

- Vector space scoring
- Efficiency considerations
 - Nearest neighbors and approximations

This lecture

- Results summaries
- Evaluating a search engine
 - Benchmarks
 - Precision and recall

Results summaries

Summaries

- Having ranked the documents matching a query, we wish to present a results list
- Typically, the document title plus a short summary
- Title – typically automatically extracted
- What about the summaries?

Summaries

- Two basic kinds:
 - Static and
 - Query-dependent (Dynamic)
- A static summary of a document is always the same, regardless of the query that hit the doc
- Dynamic summaries attempt to explain why the document was retrieved for the query at hand

Static summaries

- In typical systems, the static summary is a subset of the document
- Simplest heuristic: the first 50 (or so – this can be varied) words of the document
 - Summary cached at indexing time
- More sophisticated: extract from each document a set of “key” sentences
 - Simple NLP heuristics to score each sentence
 - Summary is made up of top-scoring sentences.
- Most sophisticated, seldom used for search results: NLP used to synthesize a summary

Dynamic summaries

- Present one or more “windows” within the document that contain several of the query terms
- Generated in conjunction with scoring
 - If query found as a phrase, the occurrences of the phrase in the doc
 - If not, windows within the doc that contain multiple query terms
- The summary itself gives the entire content of the window – all terms, not only the query terms – how?

Generating dynamic summaries

- If we have only a positional index, cannot (easily) reconstruct context surrounding hits
- If we cache the documents at index time, can run the window through it, cueing to hits found in the positional index
 - E.g., positional index says “the query is a phrase in position 4378” so we go to this position in the cached document and stream out the content
- Most often, cache a fixed-size prefix of the doc
 - Cached copy can be outdated

Evaluating search engines

Measures for a search engine

- How fast does it index
 - Number of documents/hour
 - (Average document size)
- How fast does it search
 - Latency as a function of index size
- Expressiveness of query language
 - Speed on complex queries

Measures for a search engine

- All of the preceding criteria are *measurable*:
 - we can quantify speed/size
 - we can make expressiveness precise
- The key measure: user happiness
 - What is this?
 - Speed of response/size of index are factors
 - But blindingly fast, useless answers won't make a user happy
- Need a way of quantifying user happiness

Measuring user happiness

- Issue: who is the user we are trying to make happy?
 - Depends on the setting
- **Web engine:** user finds what they want and return to the engine
 - Can measure rate of return users
- **eCommerce site:** user finds what they want and make a purchase
 - Is it the end-user, or the eCommerce site, whose happiness we measure?
 - Measure time to purchase, or fraction of searchers who become buyers?

Measuring user happiness

- **Enterprise** (company/govt/academic): Care about “user productivity”
 - How much time do my users save when looking for information?
 - Many other criteria having to do with breadth of access, secure access... more later

Happiness: elusive to measure

- Commonest proxy: **relevance** of search results
- But how do you measure relevance?
- Will detail a methodology here, then examine its issues
- Requires 3 elements:
 1. A benchmark document collection
 2. A benchmark suite of queries
 3. A binary assessment of either **Relevant** or **Irrelevant** for each query-doc pair

Evaluating an IR system

- Note: **information need** is translated into a **query**
- Relevance is assessed relative to the **information need** *not* the **query**
- E.g., Information need: *I'm looking for information on whether drinking red wine is more effective at reducing your risk of heart attacks than white wine.*
- Query: **wine red white heart attack effective**
- Evaluate whether the doc addresses the information need, not whether it has these words

Standard relevance benchmarks

- TREC - National Institute of Standards and Testing (NIST) has run large IR test bed for many years
- 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 **Irrelevant**
 - or at least for subset of docs that some system returned for that query

Unranked results

- We next assume that the search engine returns a set of documents as potentially relevant
- Does not perform any ranking
- We want to assess the quality of these results

Precision and Recall

- **Precision:** fraction of retrieved docs that are relevant = $P(\text{relevant}|\text{retrieved})$
- **Recall:** fraction of relevant docs that are retrieved = $P(\text{retrieved}|\text{relevant})$

	Relevant	Irrelevant
Retrieved	tp	fp
Not Retrieved	fn	tn

- Precision $P = \text{tp}/(\text{tp} + \text{fp})$
- Recall $R = \text{tp}/(\text{tp} + \text{fn})$

Accuracy

- Given a query an engine classifies each doc as “Relevant” or “Irrelevant”.
- Accuracy of an engine: the fraction of these classifications that is correct.
- The **accuracy** of an engine: the fraction of these classifications that are correct
 - $(tp + tn) / (tp + fp + fn + tn)$
- **Accuracy** is a commonly used evaluation measure in machine learning classification work
- Why is this not a very useful evaluation measure in IR?

Why not just use accuracy?

- How to build a 99.9999% accurate search engine on a low budget....



snoogle.com

Search for:

0 matching results found.

- People doing information retrieval want to find *something* and have a certain tolerance for junk.

Precision/Recall

- 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
 - Precision usually decreases (in a good system)

Difficulties in using precision/recall

- Should average over large corpus/query ensembles
- Need human relevance assessments
 - People aren't reliable assessors
- Assessments have to be binary
 - Nuanced assessments?
- Heavily skewed by corpus/authorship
 - Results may not translate from one domain to another

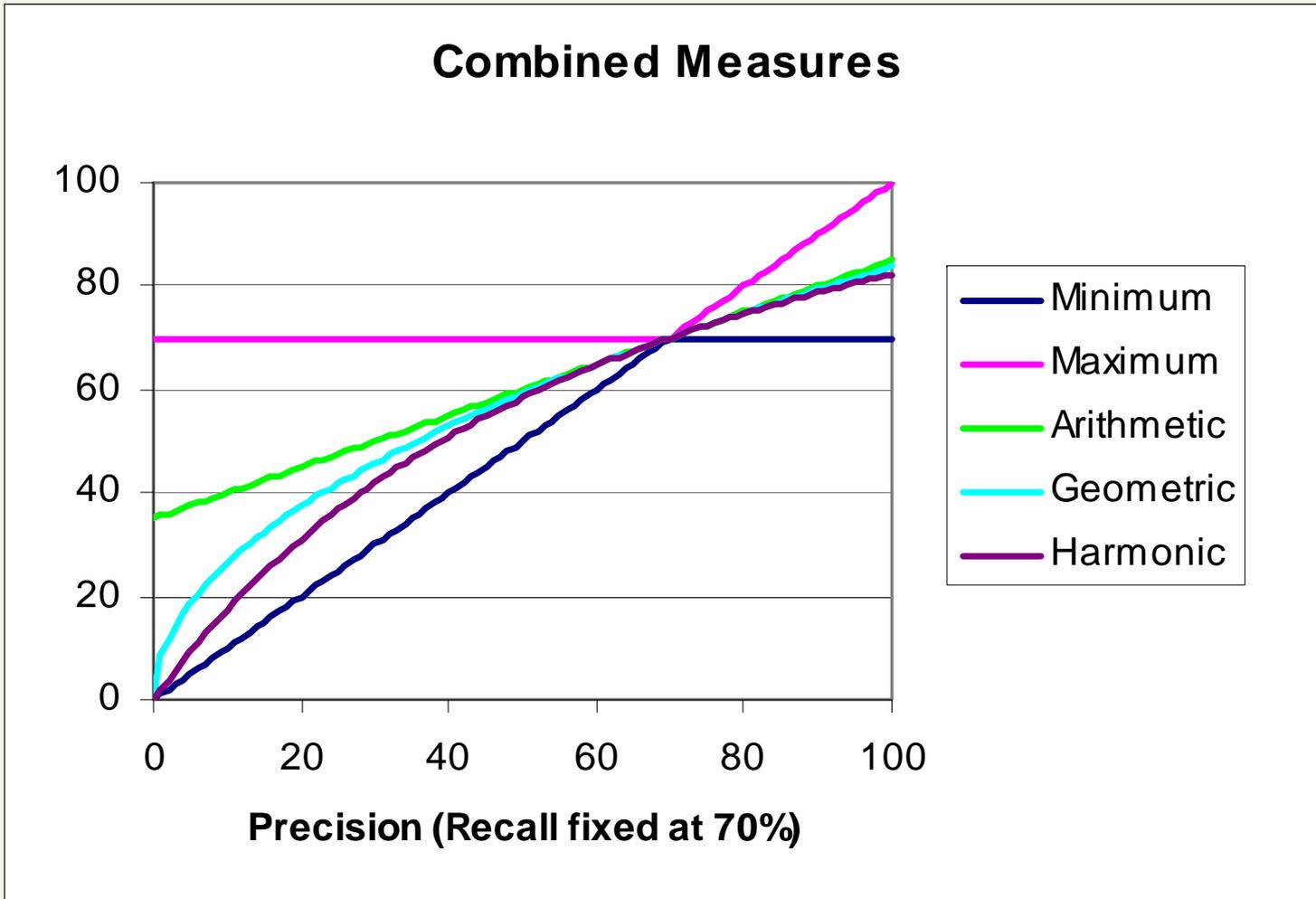
A combined measure: F

- Combined measure that assesses this tradeoff is F measure (weighted harmonic mean):

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

- People usually use balanced F_1 measure
 - i.e., with $\beta = 1$ or $\alpha = \frac{1}{2}$
- Harmonic mean is conservative average
 - See CJ van Rijsbergen, *Information Retrieval*

F_1 and other averages



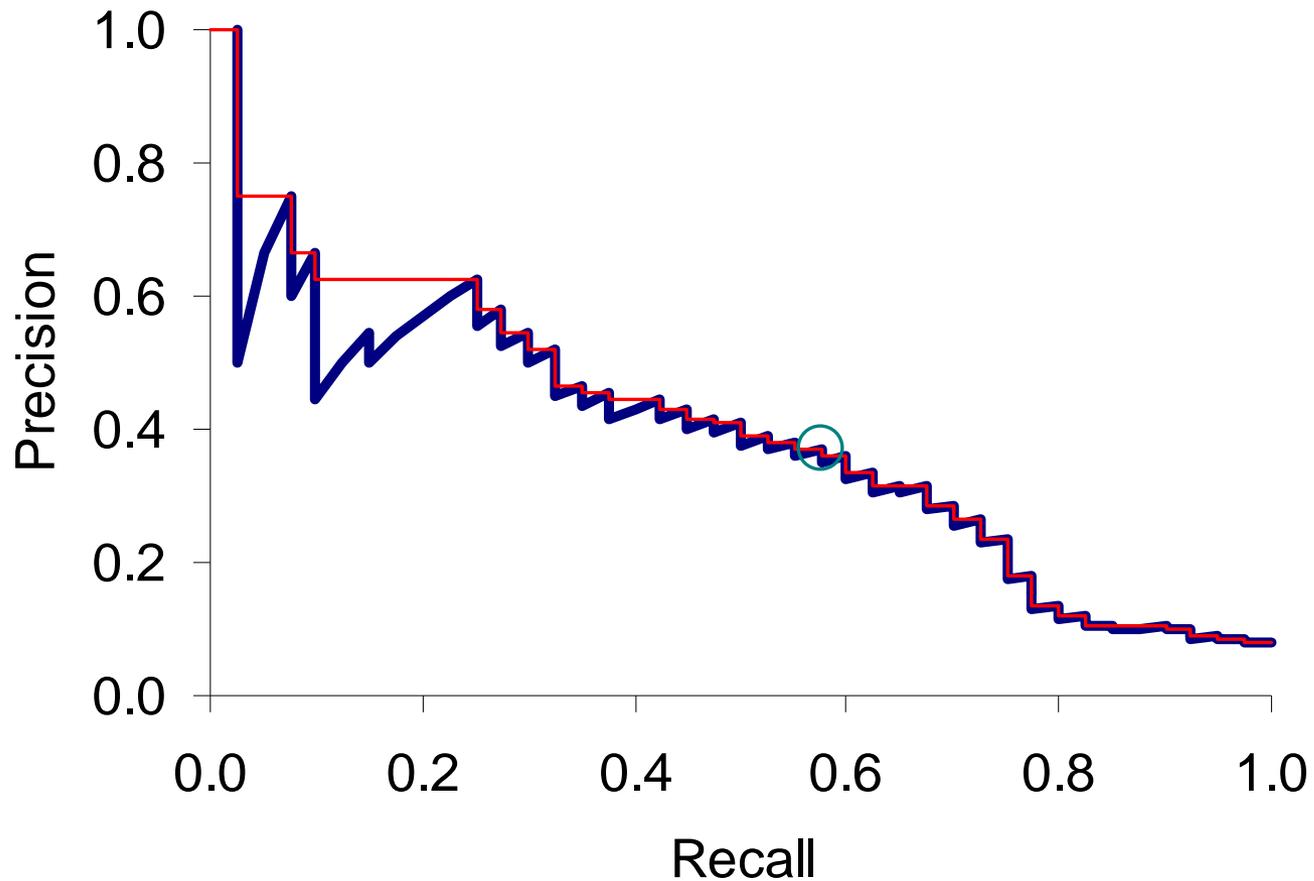
Ranked results

- Now we assume a search engine that returns a set of results ranked according to relevance
- We want to also assess the ranking
- Evaluation of ranked results:
 - You can return any number of results
 - By taking various numbers of returned documents (levels of recall), you can produce a *precision-recall curve*

Precision at k

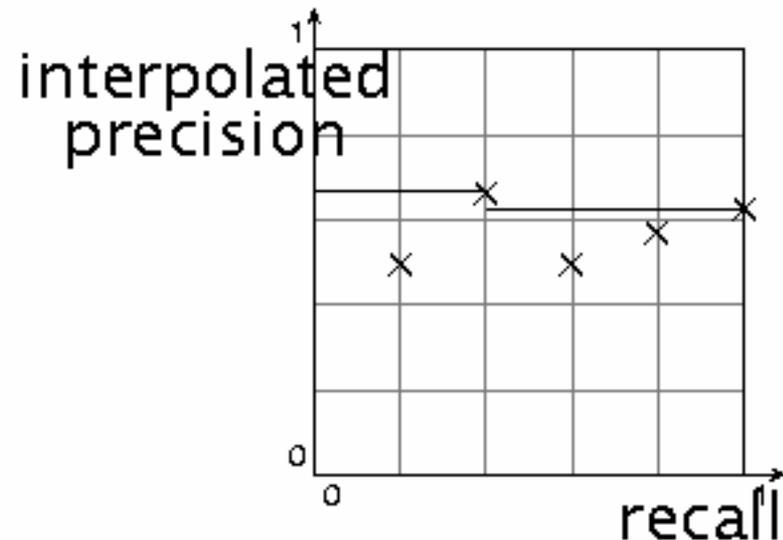
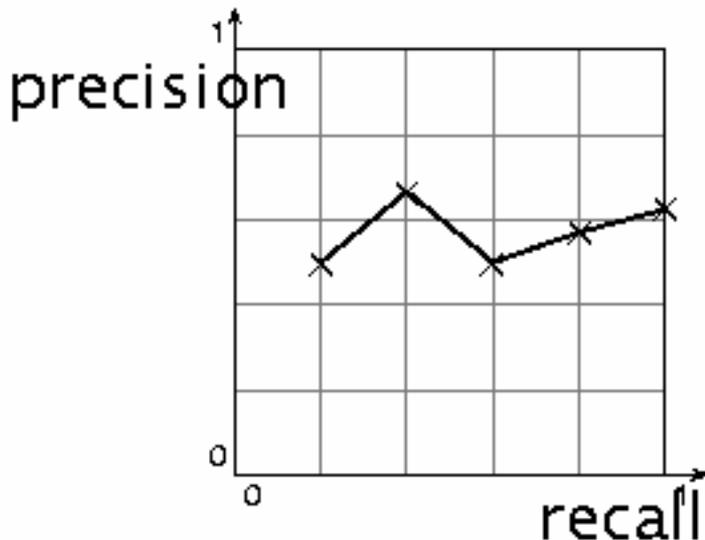
- We look only at the first k docs and we consider it as a set
- We compute the precision as before
- $K = 1, 10, 100, \dots$

A precision-recall curve



Interpolated precision

- If you can increase precision by increasing recall, then you should get to count that...
- So you take the max of precisions to right of value

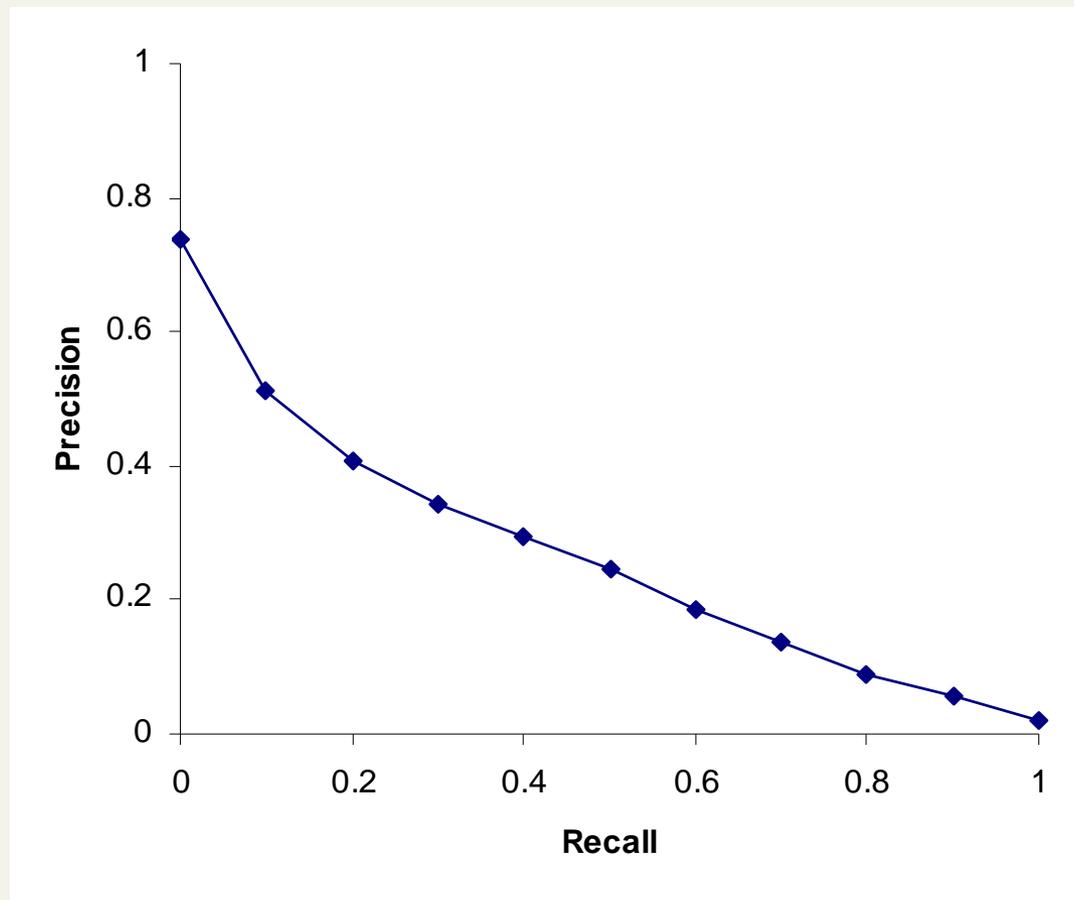


Evaluation

- 11-point interpolated average precision
 - The standard measure in the TREC competitions: you take the precision at 11 levels of recall varying from 0 to 1 by tenths of the documents, using interpolation (the value for 0 is always interpolated!), and average them

Typical (good) 11 point precisions

- SabIR/Cornell 8A1 11pt precision from TREC 8 (1999)



Yet more evaluation measures...

- Mean average precision (MAP)
 - Average of the precision value obtained for the top k documents, each time a relevant doc is retrieved
 - Avoids interpolation, use of fixed recall levels
- R-precision
 - If we have a known (though perhaps incomplete) set of relevant documents of size Rel , then calculate precision of the top Rel docs returned
- Check IIR Chapter 8.4 for more details
- There is not a best measure. Each measure gives different type of information. Which is more appropriate depends on the application

Resources

- IIR Chapters 8 – 8.4, 8.7