Information Retrieval and Organisation

Dell Zhang

Birkbeck, University of London

2015/16

IR Chapter 08

Evaluation in Information Retrieval

Measures for an IR System

- How fast does it index
  - Number of documents/bytes per hour
- How fast does it search
  - Latency as a function of index size or queries per second
- What is the cost per query?
  - Given certain requirements, e.g., a 20-billion-page index

Measures for an IR System

- All of the preceding criteria are *measurable*
  - We can quantify speed / size / money
- However, the key measure for a search engine is *user happiness*
  - What is user happiness and how do we measure it?
  - Factors include:
    - Speed of response
    - Size of index
    - Uncluttered UI
    - Most important: relevance
  - Note that none of these is sufficient: blindingly fast, but useless answers won't make a user happy.

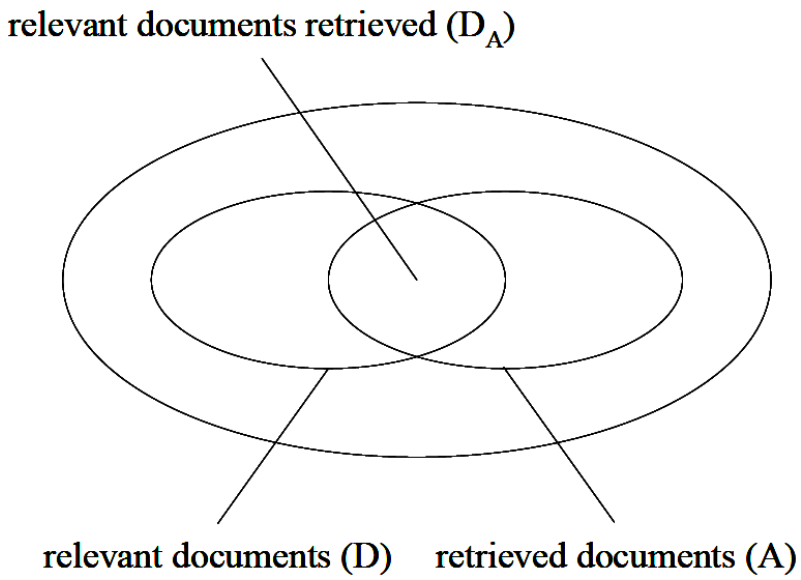
Measuring User Happiness

- Most common definition of user happiness: relevance of returned documents
- How do we measure the quality of what is returned by an IR system?
- There are two basic measures:
  - Precision P: the fraction of retrieved documents that are relevant
  - Recall R: the fraction of relevant documents that are retrieved

Precision and Recall

- Let us give a more formal definition
- Let A be the set of retrieved documents, D be the set of relevant documents and D_A the set of relevant documents retrieved, then

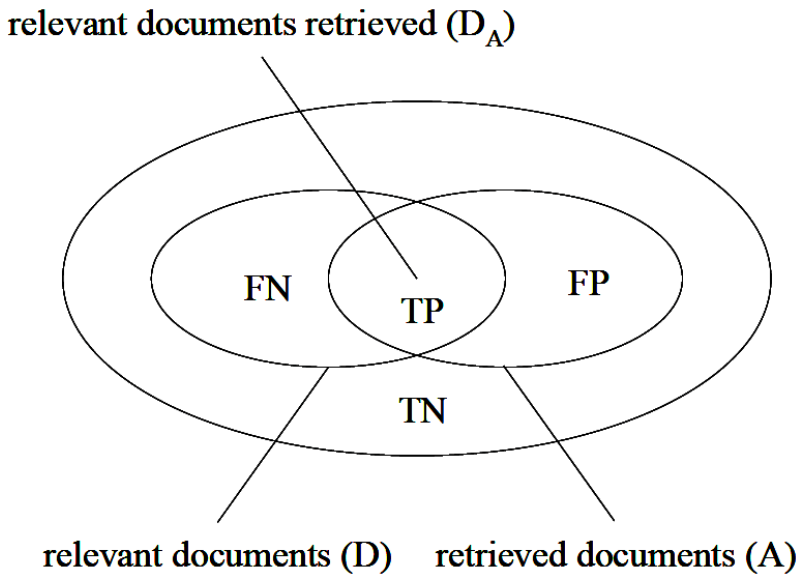
$$P = \frac{|D_A|}{|A|} \text{ and } R = \frac{|D_A|}{|D|}$$



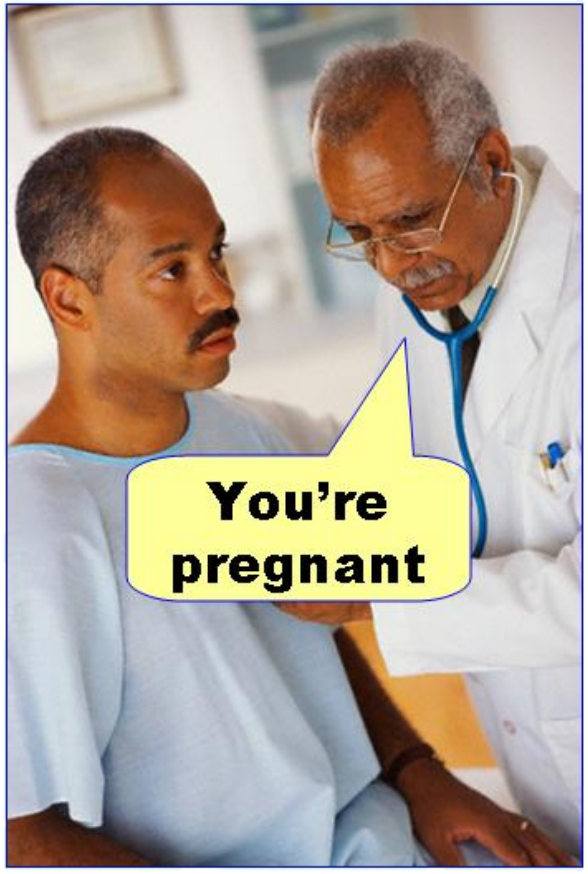
Alternative Definition

	Relevant	Non-relevant
Retrieved	true positives (TP)	false positives (FP)
Not-retrieved	false negatives (FN)	true negatives (TN)

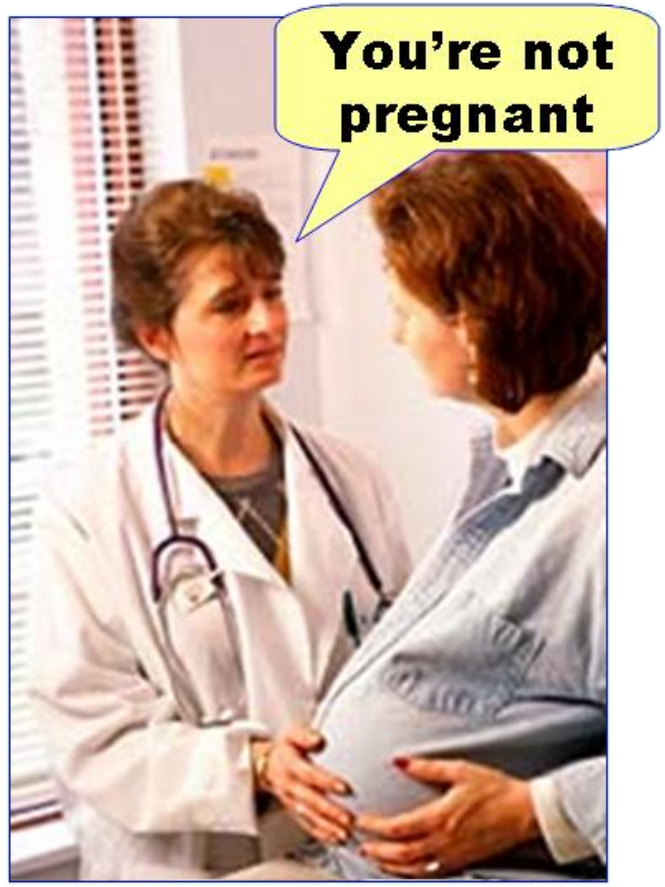
$$P = TP/(TP + FP) \text{ and } R = TP/(TP + FN)$$



(false positive)



(false negative)



Accuracy

- Why do we use complex measures like precision and recall?
- Why not something simple like accuracy?
  - Accuracy is the fraction of decisions (relevant/nonrelevant) that are correct.
  - In terms of the contingency table above, accuracy $= (TP + TN)/(TP + FP + FN + TN)$.
- There's a problem with that ...

Accuracy

- Simple trick to maximize accuracy in IR:
  - always say 'no' and return nothing, then
  - you get 99.99% accuracy on most queries (there is a huge number of true negatives you get right)
- Searchers on the web (and in IR in general) *want to find something* and have a certain tolerance for junk.

Precision/Recall Trade-off

- You can increase recall by returning more docs
  - Recall is a non-decreasing function of the number of docs retrieved
  - A system that returns all docs has 100% recall!
- The converse is (usually) also true:
  You can increase precision by returning fewer docs
  - A system that only returns documents that have a very high score (usually) has a high precision

Precision/Recall Trade-off

- Depending on the application one or the other may be more important:
  - Typical web surfers
    - would like every result on the first page to be relevant (high precision)
    - are not interested in looking at every document that might be relevant (there might be millions)
  - Various professional searchers such as paralegals and intelligence analysts
    - are usually very concerned with trying to get as high recall as possible
    - will tolerate fairly low precision to avoid missing relevant results

A Single Measure

- For comparison reasons it's easier to have a single number
- We can use a combined measure

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

  - $\beta^2 = (1-\alpha)/\alpha$
  - β is a parameters with which we express how important precision and recall are to us —
    $\beta < 1$: precision is more important;
    $\beta > 1$: recall is more important.

F_1 Measure

- The most frequently used F measure is the *balanced* one with $\beta = 1$ (or $\alpha = \frac{1}{2}$), commonly written as F_1
- F_1 is the *harmonic mean* of P and R:

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

- Why use the harmonic mean?
  - The simple (arithmetic) mean is 50% for "return-everything" IR system, which is too high
  - Desideratum: punish really bad performance on either precision or recall

Precision-Recall Curve

- Precision/recall are measures for *unranked sets*
- We can easily turn set measures into measures of *ranked lists*.
- Just compute the set measure for each "prefix": the top 1, top 2, top 3, top 4, ... results
- Doing this for precision and recall gives you a *precision-recall curve*.

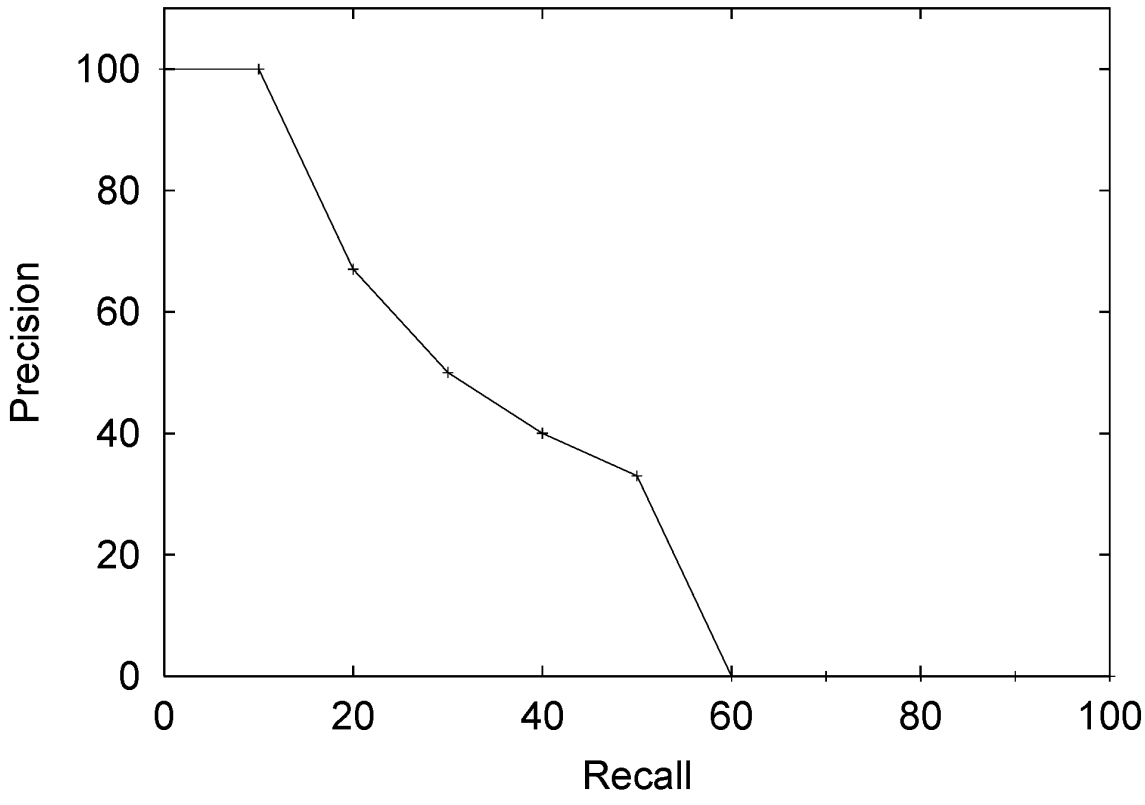
Precision-Recall Curve

- Example
  - Assume following documents are relevant for the query q: $\{d_3, d_5, d_9, d_{25}, d_{39}, d_{44}, d_{56}, d_{71}, d_{89}, d_{123}\}$
  - IR system gives back this ranked list:

Ranking	Recall	Precision
1. d_{123} ←	10%	100%
2. d_{84}	10%	50%
3. d_{56} ←	20%	67%
4. d_6	20%	50%
5. d_8	20%	40%
6. d_9 ←	30%	50%
7. d_{511}	30%	43%
8. d_{129}	30%	38%
9. d_{187}	30%	33%
10. d_{25} ←	40%	40%
11. d_{38}	40%	36%
12. d_{48}	40%	33%
13. d_{250}	40%	31%
14. d_{113}	40%	29%
15. d_3 ←	50%	33%

Precision-Recall Curve

- As the recall is increasing monotonically, we can plot the precision in relation to the recall:

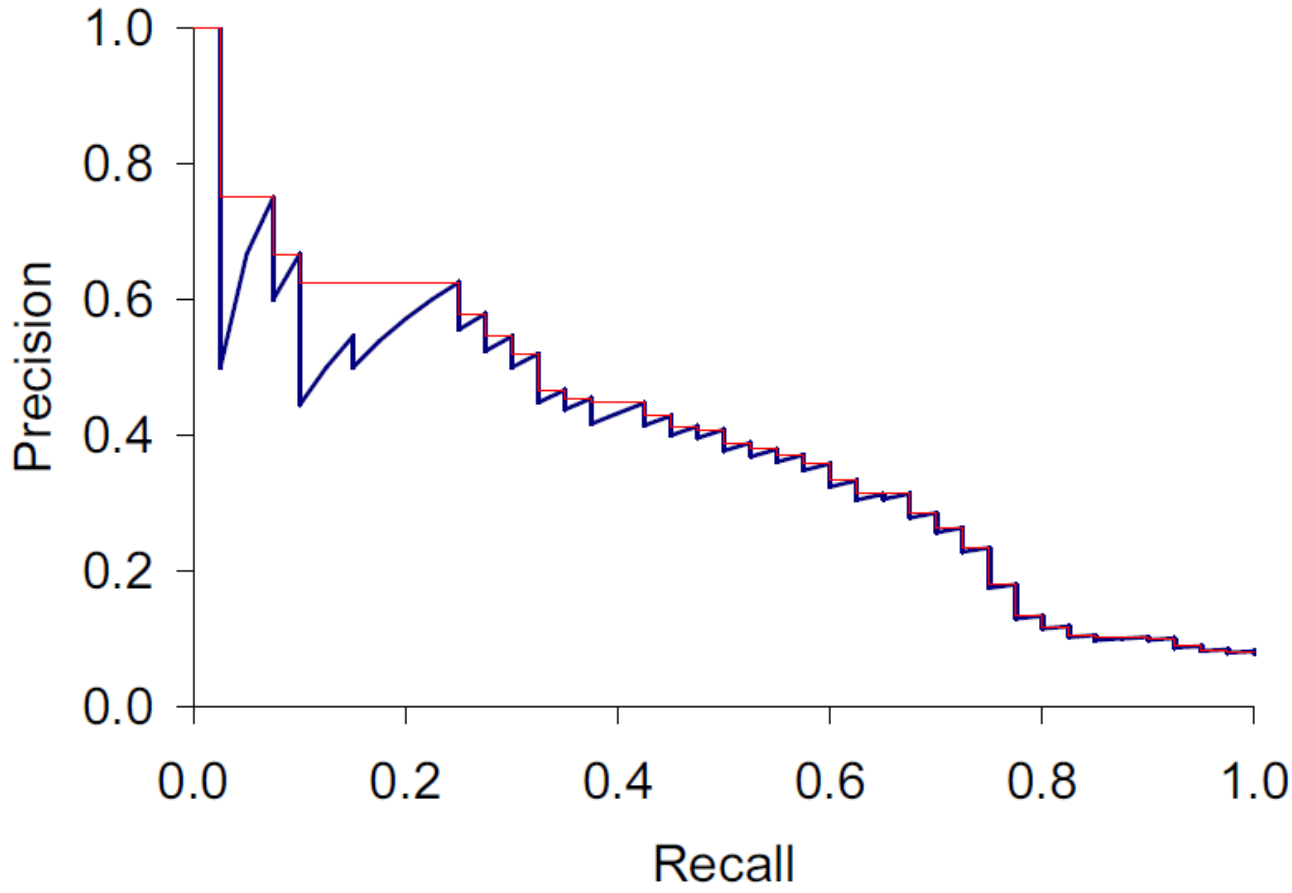


11-Point Interpolated Average Precision

- Examining the entire precision-recall curve can be very informative, but often we only want an overview
- The traditional way of doing this is the 11-point interpolated average precision
  - For each test query, the *interpolated* precision is measured at the 11 recall levels of $0.0, 0.1, 0.2, \ldots, 1.0$
  - We then average the precision at each level

11-Point Interpolated Average Precision

- Interpolation (in red): take the maximum of all future points
  - Rationale: the user is willing to look at more stuff if both precision and recall get better



MAP

- If the set of relevant documents for a query $q_j \in Q$ is $\{d_1, \ldots, d_{m_j}\}$ and R_{jk} is the set of ranked retrieval results from the top result until you get to document d_k, then the Mean Average Precision (MAP) is:

$$MAP(Q) = \underbrace{\frac{1}{|Q|}\sum_{j=1}^{|Q|}}_{\text{Mean}} \underbrace{\frac{1}{m_j}\sum_{k=1}^{m_j}}_{\text{Average}} Precision(R_{jk})$$

  - No use of fixed recall levels. No interpolation.
  - When no relevant doc is retrieved, the average precision is taken to be 0.

Precision/Recall at k

- Prec@k: Precision on the top k retrieved docs.
  - Appropriate for Web search engines: most users scan only the first few (e.g., 10) links that are presented.
- Rec@k: Recall on the top k retrieved docs.
  - Appropriate for archival retrieval systems: what fraction of total number of relevant docs did a user find after scanning the first few (say 100) docs?

R-Precision

- Precision at *Rel*
  - *Rel* is the size of a set of known-to-be relevant documents (though perhaps incomplete).
  - A perfect IR system could score 1 on this metric for each query.

PRBEP

- Given a precision-recall curve, the Precision/Recall Break-Even Point (PRBEP) is the value at which the precision is equal to the recall.
  - It is obvious from the definition of precision/recall, the equality is achieved for contingency tables with $TP + FP = TP + FN$, i.e., when the number of retrieved documents is the same as the number of relevant documents.
  - It is equivalent to R-Precision when there are indeed Rel relevant documents in total.

Queries vs Information Needs

- Where do we get the queries with which to test the system?
  - We'll talk about this in just a minute . . .
- We still haven't defined when a document is relevant.
  - Who decides when a document is relevant and relevant to what?
  - "Relevance to a query" is very problematic.
  - A user starts out with an *information need*, not a query

Queries vs Information Needs

- Let's look at an example:
  - Information need i: You are looking for information on whether drinking red wine is more effective at reducing the risk of heart attacks than white wine.
    - This is an information need, not a query.
  - (Possible) query q: `wine AND red AND white AND heart AND attack`
  - Consider document d': "He then launched into the heart of his speech and attacked the wine industry lobby for downplaying the role of red and white wine in drunk driving."
    - d' is relevant to the query q ...
    - d' is not relevant to the information need i.

Queries vs Information Needs

- User happiness can only be measured by *relevance to an information need*, not by *relevance to queries*.
- We've been a bit sloppy with our terminology:
  - We talk about query/document relevance judgements

  even though we mean
  - information-need/document relevance judgements

Benchmarks

- What we need is a *benchmark*
- A benchmark for IR systems consists of
  - A collection of documents
    - Documents must be representative of the documents we expect to see in reality.
  - A collection of information needs (which we will often incorrectly refer to as queries)
    - Information needs must be representative of the information needs we expect to see in reality.
  - And last but not least: human relevance assessments
    - We need to hire/pay "judges" or assessors to do this (expensive, time-consuming).
    - Judges must be representative of the users we expect to see in reality.

Standard Relevance Benchmarks

- Cranfield
  - Pioneering: first testbed allowing precise quantitative measures of information retrieval effectiveness
  - Late 1950s, UK
  - 1398 abstracts of aerodynamics journal articles, a set of 225 queries, exhaustive relevance judgements of all query-document-pairs
  - Too small, too untypical for serious IR evaluation today

Standard Relevance Benchmarks

- TREC
  - TREC = Text Retrieval Conference (TREC)
  - Organized by the U.S. National Institute of Standards and Technology (NIST)
  - TREC is actually a set of several different relevance benchmarks.
  - Best known: TREC Ad Hoc, used for first 8 TREC evaluations between 1992 and 1999
  - 1.89 million documents, mainly newswire articles, 450 information needs

TREC: Example Collection

Disk	*Contents*	*Size Mb*	*Number Docs*	*Words/Doc. (median)*	*Words/Doc. (mean)*
1	WSJ, 1987-1989	267	98,732	245	434.0
	AP, 1989	254	84,678	446	473.9
	ZIFF	242	75,180	200	473.0
	FR, 1989	260	25,960	391	1315.9
	DOE	184	226,087	111	120.4
2	WSJ, 1990-1992	242	74,520	301	508.4
	AP, 1988	237	79,919	438	468.7
	ZIFF	175	56,920	182	451.9
	FR, 1988	209	19,860	396	1378.1
3	SJMN, 1991	287	90,257	379	453.0
	AP, 1990	237	78,321	451	478.4
	ZIFF	345	161,021	122	295.4
	PAT, 1993	243	6,711	4,445	5391.0
4	FT, 1991-1994	564	210,158	316	412.7
	FR, 1994	395	55,630	588	644.7
	CR, 1993	235	27,922	288	1373.5
5	FBIS	470	130,471	322	543.6
	LAT	475	131,896	351	526.5
6	FBIS	490	120,653	348	581.3

TREC: Example Collection

- Data Sources:
  - WSJ = Wall Street Journal
  - AP = Associated Press
  - ZIFF = Computer Selects, Ziff-Davis
  - FR = Federal Register
  - DOE = US DOE Publications
  - SJMN = San Jose Mercury News
  - PAT = US Patents
  - FT = Financial Times
  - CR = Congressional Record
  - FBIS = Foreign Broadcast Information Service
  - LAT = LA Times

TREC: Example Document

```
<doc>

<docno> WSJ880406-0090 </docno>
<hl> AT&T Unveils Services to Upgrade Phone Networks Under
Global Plan </hl>
<author> Janet Guyon (WSJ Staff) </author>
<dateline> New York </dateline>

<text>
American Telephone & Telegraph Co. introduced the first of a new
generation of phone services with broad ...
</text>

</doc>
```

- Documents contain SGML markup tags
- Important fields like document number (<docno>) and text (<text>) can be found in all documents

TREC: Example Information Need

<top>

<num> Number: 168
<title> Topic: Financing AMTRAK

<desc> Description:
A document will address the role of the Federal Government in financing the operation of the National Railroad Transportation Corporation (AMTRAK).

<narr> Narrative: A relevant document must provide information on the government's responsibility to make AMTRAK an economically viable entity. It could also discuss the privatization of AMTRAK as an alternative to continuing government subsidies. Documents comparing government subsidies given to air and bus transportation with those provided to AMTRAK would also be relevant.

</top>

- Information needs (topics) are defined in natural language
- These have to be translated into a query and then processed

TREC: Relevance

- No exhaustive relevance judgements: that would be too expensive
- NIST assessors' relevance judgements are available only for the documents that were among the top-K
- This means the top-K of systems entered in the TREC evaluation for which the information need was developed

Standard Relevance Benchmarks

- GOV2
  - Another TREC/NIST collection
  - 25 million web pages
  - Largest collection that is easily available
  - But still 3 orders of magnitude smaller than what Google/Yahoo/MSN index
- NTCIR
  - East Asian language and cross-language information retrieval
- Cross Language Evaluation Forum (CLEF)
  - This evaluation series has concentrated on European languages and cross-language information retrieval

Evaluation of Large IR Systems

- How do you measure recall on the web?
  - Search engines often use precision at top-K, e.g., $K = 10$ ...
  - ... or measures that reward you more for getting rank 1 right than for getting rank 10 right
- Search engines also use non-relevance-based measures.
  - Example 1: clickthrough on first result
    - Not very reliable if you look at a single clickthrough
    - ... but pretty reliable in the aggregate.
  - Example 2: Ongoing studies of user behaviour in the lab

A/B Testing

- Purpose: Test a single innovation
  - Have most users use old system (pre-requisite: you have a large search engine up and running)
  - Divert a small proportion of traffic (e.g., 1%) to the new system that includes the innovation
  - Evaluate with an "automatic" measure like clickthrough on first result
  - Now we can directly see if the innovation does improve user happiness.
- Probably the evaluation methodology that large search engines trust most
  - Variant: Give users the option to switch to new algorithm/interface

Result Summaries

- How do we present results to the user?
  - Most often: as a list – aka "10 blue links"
- How should each document in the list be described?
  - This description is crucial.
  - User can identify good hits (= relevant hits) based on description.
  - No need to "click" on all documents sequentially

Result Summaries

- Doc description in result List
  - Most commonly: doc title, url, some metadata . . . , and a *summary*
- How do we "compute" the summary? Two basic kinds: (i) static (ii) dynamic.
  - A *static summary* of a document is always the same, regardless of the query that hit the document
  - *Dynamic summaries* are *query-dependent*. They attempt to explain why the document was retrieved for the query at hand.

Result Summaries

- Static Summaries
  - Simplest form of summary takes e.g. the first two sentences or 50 words of a document
  - May also extract information from a particular zone of the document or from metadata, e.g. title and author
  - Typically extracted and cached at indexing time, so that it can be retrieved and presented quickly
  - There are more sophisticated approaches using natural language processing (NLP).
    - Many of these are still subject to research and not within the scope of this module.

Result Summaries

- Dynamic Summaries
  - Dynamic summaries display one or more "windows" on the document
  - Usually windows contain query terms, and so are often referred to as *keyword-in-context* or *KWIC* snippets
    - If the query is found as a phrase, occurrences of the phrase in the document will be shown as the summary
    - If not, windows within the document that contain multiple query terms will be selected
    - These windows may just stretch some number of words to the left and right of the query terms
    - NLP can also be employed usefully: users prefer snippets that read well because they contain complete phrases

Result Summaries

- Dynamic Summaries
  - They are liked by users: you can scan them to decide if you want to click (e.g. Google provides them). However, not easy to implement as they cannot be precomputed.
  - Reconstructing the context with only a positional index is also difficult and time-consuming, but generating snippets must be fast since many snippets are typically generated for each query.
  - Caching the whole documents is not feasible: it is common to cache a fixed-size prefix
    - For short documents, the whole document is cached
    - For longer documents we assume that prefix will contain some summary

Summary

- How to evaluate the retrieval quality of an IR system:
  - Discussing different measures for doing so
  - Explaining how relevance of documents is determined
- How to present summaries of the document answer set to a user