Information Retrieval Evaluation

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

Ian Soboroff
ian.soboroff@nist.gov

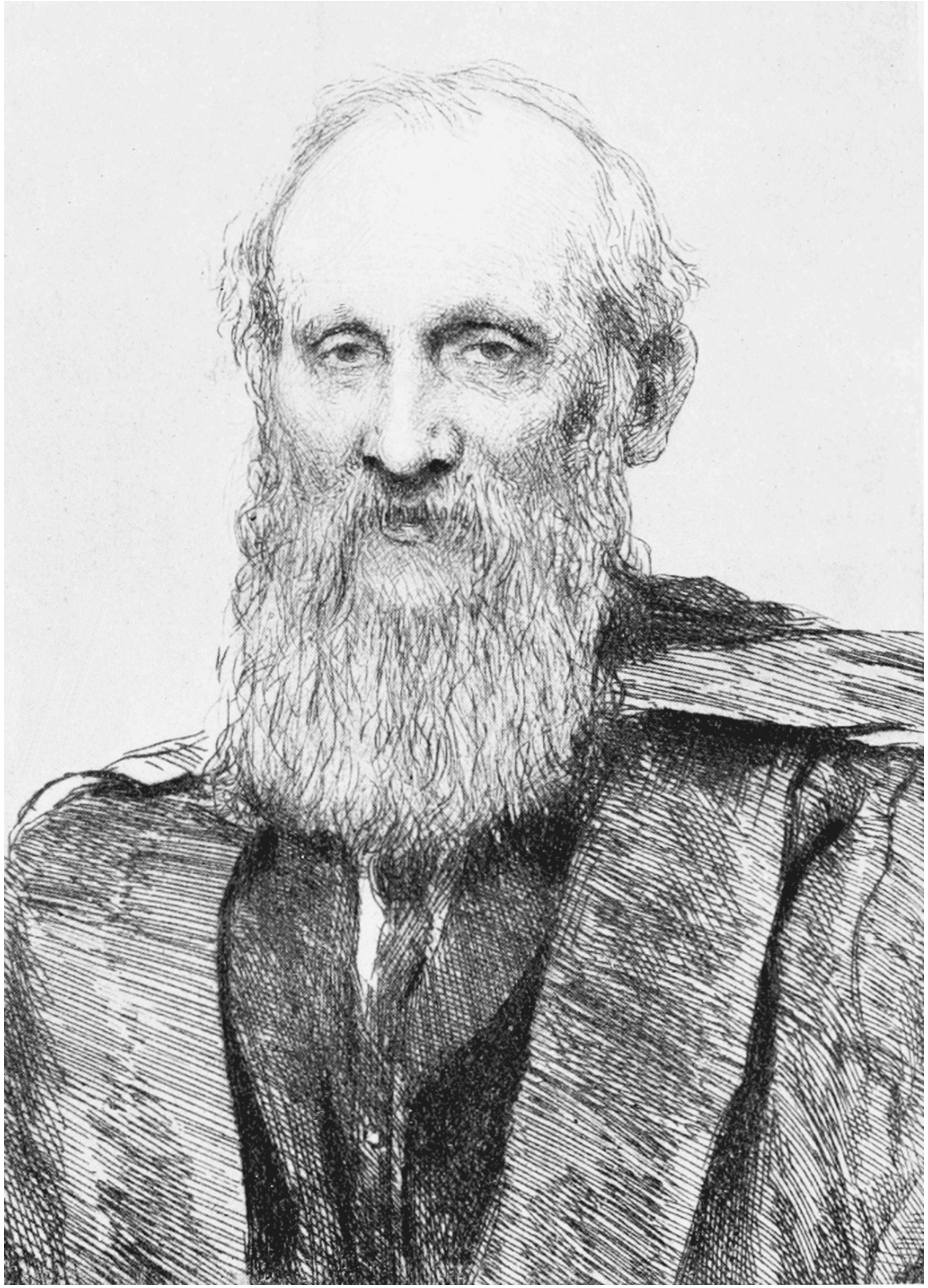
Tired? Want to ski instead?

Two books:

- ***TREC: Evaluation and Experiment***, Voorhees and Harman, ed., MIT Press.
- Diane Kelly, volume on user studies in the Morgan-Kaufman lecture series.

Agenda

- Why measure search effectiveness?
- A brief history of evaluation
- Using test collections
- Advanced test collections
- Designing test collections
- Interactive evaluation
- Research issues (sprinkled throughout)



“To measure is to know.”

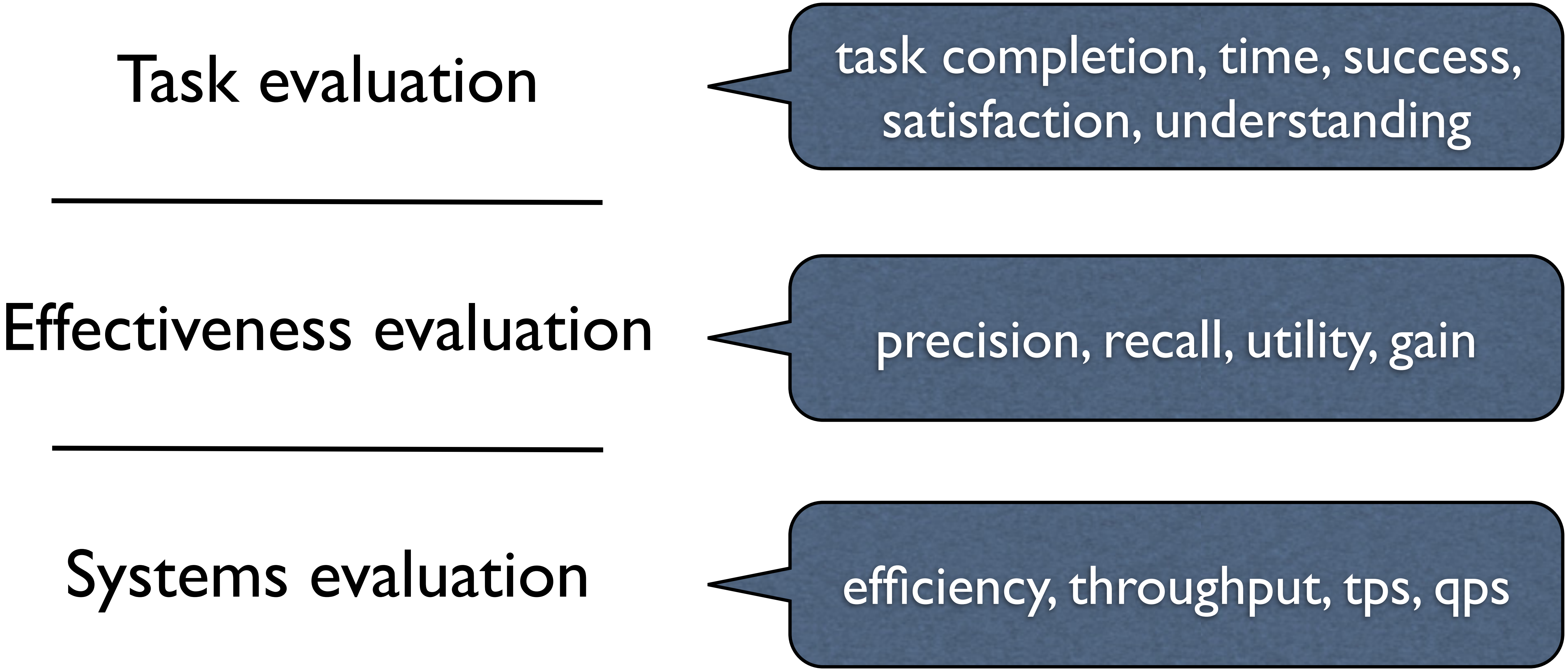
“If you cannot measure it,
you cannot improve it.”

“The true measure of
a man is what he would do
if he knew he would never
be caught."

-- Lord William Thompson,
first Baron Kelvin

NIST
National Institute of
Standards and Technology
U.S. Department of Commerce

Three levels for measurement



Task evaluation

task completion, time, success, satisfaction, understanding

Effectiveness evaluation

precision, recall, utility, gain

Systems evaluation

efficiency, throughput, tps, qps

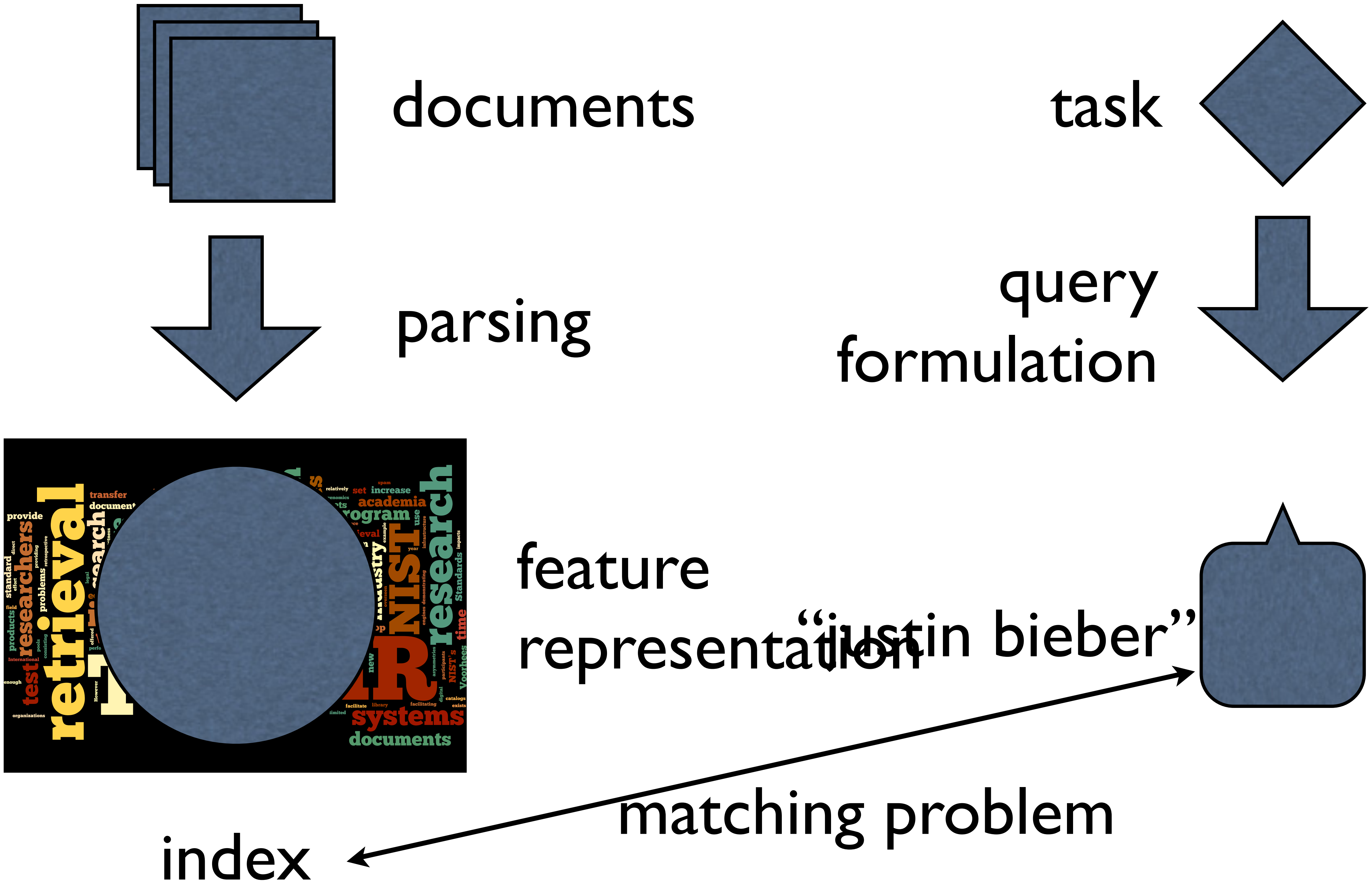
Two worlds

Databases:	Information retrieval:
• Items are structured, typed, and well-formed.	• Items are at best semi-structured.
• Semantics of values are well-known.	• Natural language has messy semantics.
• Queries are structured.	• Queries are short and unstructured.
• Query results are exact.	• Query results are fuzzy.

Fuzzy?

- Information is represented in human language.
  - Computers cannot fully understand human language.
  - Lots of human language is hard to understand for humans!
- Information needs are complicated.
  - People search within a rich context.
  - People search to accomplish a task or a goal.
- Different people have a different notion of "the right answer"... and they're all right!

(Mis)Representation



documents

task

parsing

query formulation

feature representation

"justin bieber"

matching problem

index

IR Evaluation

1. System level
   - benchmarking queries/sec, memory load, utilization, availability... essentially similar to database benchmarking.
2. Effectiveness level
   - Test collection methodology.
   - Put the search engine on a lab bench and poke at it in specific places in specific ways.
3. Task evaluation level
   - User studies.
   - Design a controlled experiment with users of a system, and try to measure contrasting effects.

Measuring effectiveness

Test collection methodology:

- ... also known as the “Cranfield” paradigm.
- An abstraction of a real user’s task.
- A set of documents.
- A set of queries.
- A mapping from the queries to a set of “right answers”.
- A set of measures that follow from the task.

Early information access

- Before the web (1992) and before information was electronically available, most information access was via the library with librarians using indexed versions of journals/ book lists (such as Index Medicus, Engineering Index, card catalogs, etc.)
- These indexes were manually produced, usually following (different) guidelines

Some manual indexing issues

- What terms to use to describe an article?
- How many terms to use?
- Should the terms be grouped into phrases rather than just single terms?
- Should the terms be selected from a controlled list?
- Should the terms be expanded using a thesaurus?
- Etc.

Cranfield experiments

- Designed and led by Cyril Cleverdon, head librarian at the College of Aeronautics, Cranfield, England in the 1960s
- Goal: To learn what makes a good set of indexing terms (descriptors)

Cranfield 2 indexing schemes

- Manual
  - four different types of indexing descriptors
  - three levels of exhaustivity (31, 25, and 13 descriptors)
- "automatic" indexing using the terms from abstracts and titles

What to measure

- How well the four descriptor types and three levels of exhaustivity (12 experiments) plus the "automatic" versions functioned when used as the descriptors in a search by a librarian
- To make the results statistically sound, he would have needed to do many searches involving a LOT of librarians
- So instead he simulated the task by creating a test collection

His user simulation

- User model: researcher wanting all documents relevant to their question
- Documents to be searched: 1400 abstracts from recent papers in aeronautical engineering
- Questions were gathered from authors of the papers, asking for the basic problem the paper addressed and also supplemental questions that could have been put to an information service

Getting the correct answers

- Graduate students spent a summer checking the ~225 questions against all 1400 abstracts to find "possible" answers
- This was then filtered by authors
  - Complete answer to a question
  - High degree of relevance, necessary for work
  - Useful as background
  - Minimal interest, historical interest only
  - No interest

Final Cranfield test collection

- 1400 abstracts
- 225/221?? questions
- A list of abstracts for each question that are the correct answers (relevant documents for that question), broken into the 5 levels of relevance/non-relevance; note that ALL of the abstracts had manual relevance judgments

Cranfield experiment

- Librarians manually searched the abstracts for each question, using each of the 33 indexing descriptor combinations
- Recall and precision used as the metrics
- Results: single terms were best but the "automatic" indexing worked astonishingly well; this result led to major IR research
- Since the test collection was NOT based on the specific indexing methods used, it was infinitely reusable

Cranfield Paradigm

- Faithfully model a real user application, in this case searching appropriate abstracts with “real” questions judged by questioner
- Establish relative effectiveness differences among experimental factors
- Have “enough” documents and queries to allow significance testing on results
- Build the collection BEFORE the experiments in order to prevent human bias and to enable infinite reusability
- Base the metrics on how a user would see the results, i.e., intuitive metrics

Continuation in TREC

- In 1990 DARPA asked NIST to build a new test collection for the TIPSTER project
- User model: intelligence analysts
  - –Large numbers of full text documents from newspapers, newswires, etc.
  - –“formatted” queries called topics in TREC
  - –High recall users meaning that “complete” relevance judgments were needed

TIPSTER Disk 1 and 2

Source	Size (MB)	documents	comments
Wall Street Journal, 1987-89	267	98,732	
1990-92	242	74,520	
Associated Press newswire, 1989	254	84,678	errors, repeats
1988	237	79,919	
Federal Register 1989	260	25,960	Very long texts
1988	209	19,860	
Computer Selects articles (Ziff-Davis)	242 175	75,180 56,920	Different domain
DOE abstracts	184	226,087	Diverse domain

Sample TREC-3 Topic

<top>

<num> Number: 396

<title> sick building syndrome

<desc> Description:
Identify documents that discuss sick building syndrome or building-related illnesses.

<narr> Narrative:
A relevant document would contain any data that refers to the sick building or building-related illnesses, including illnesses caused by asbestos, air conditioning, pollution controls. Work-related illnesses caused by the building, such as carpal tunnel syndrome, are not relevant.

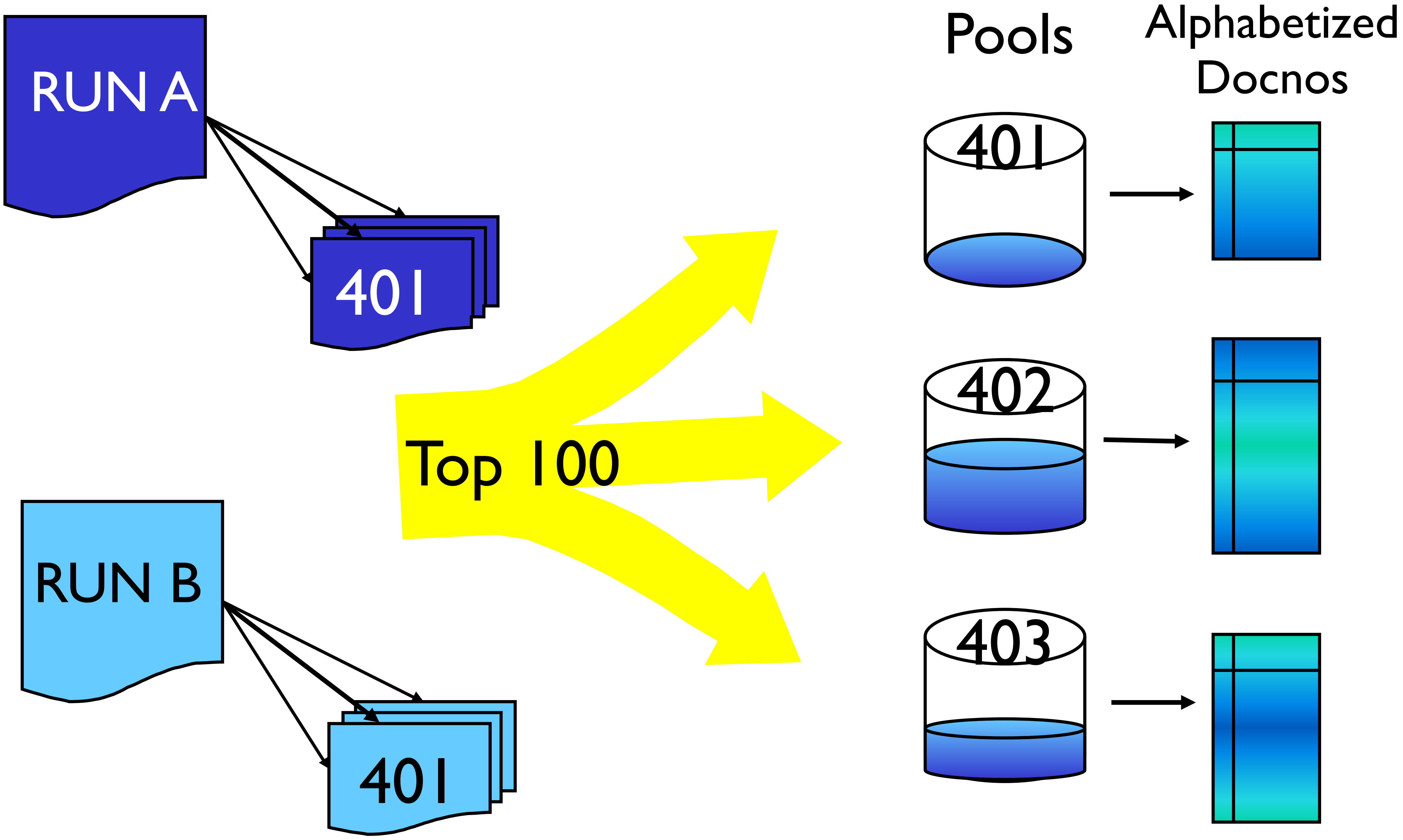
</top>

Relevance Judgments

Three possible methods for finding the relevant documents FOR EACH TOPIC:

- Full relevance judgments on all 2GB of documents
- Relevance judgments on a random sampling of the document collection
- Relevance judgments on the sample of documents selected by the various participating systems
  - This method is known as the pooling method, and had been used successfully in creating the NPL and INSPEC collections.

Pooling


RUN A
401
RUN B
401
Top 100
Pools
Alphabetized Docnos
401
402
403

What is relevant?

- Back to the user model (plus pragmatics)
- A document is relevant if you would use it in a report in some manner
- This means that even if only one sentence is useful, the document is relevant
- "Duplicates" also relevant as it would be very difficult to define and remove these

Relevancy FAQs

1. How do you know you have "all" the answers if not everything is judged??
    a. If documents that are not judged are automatically declared non relevant, isn't this biased against new systems, either not in the pool or "majorly" different in methodology?
2. These are manual judgments and there is known to be large variations of opinion; doesn't this make the results "unstable"?

How complete is relevant set?

- TREC-3 study: documents beyond rank 100 added to the pool for judgment
  - Some additional relevant documents found, however not enough to effect system ranking
  - topics with many relevant tend to have even more relevant documents

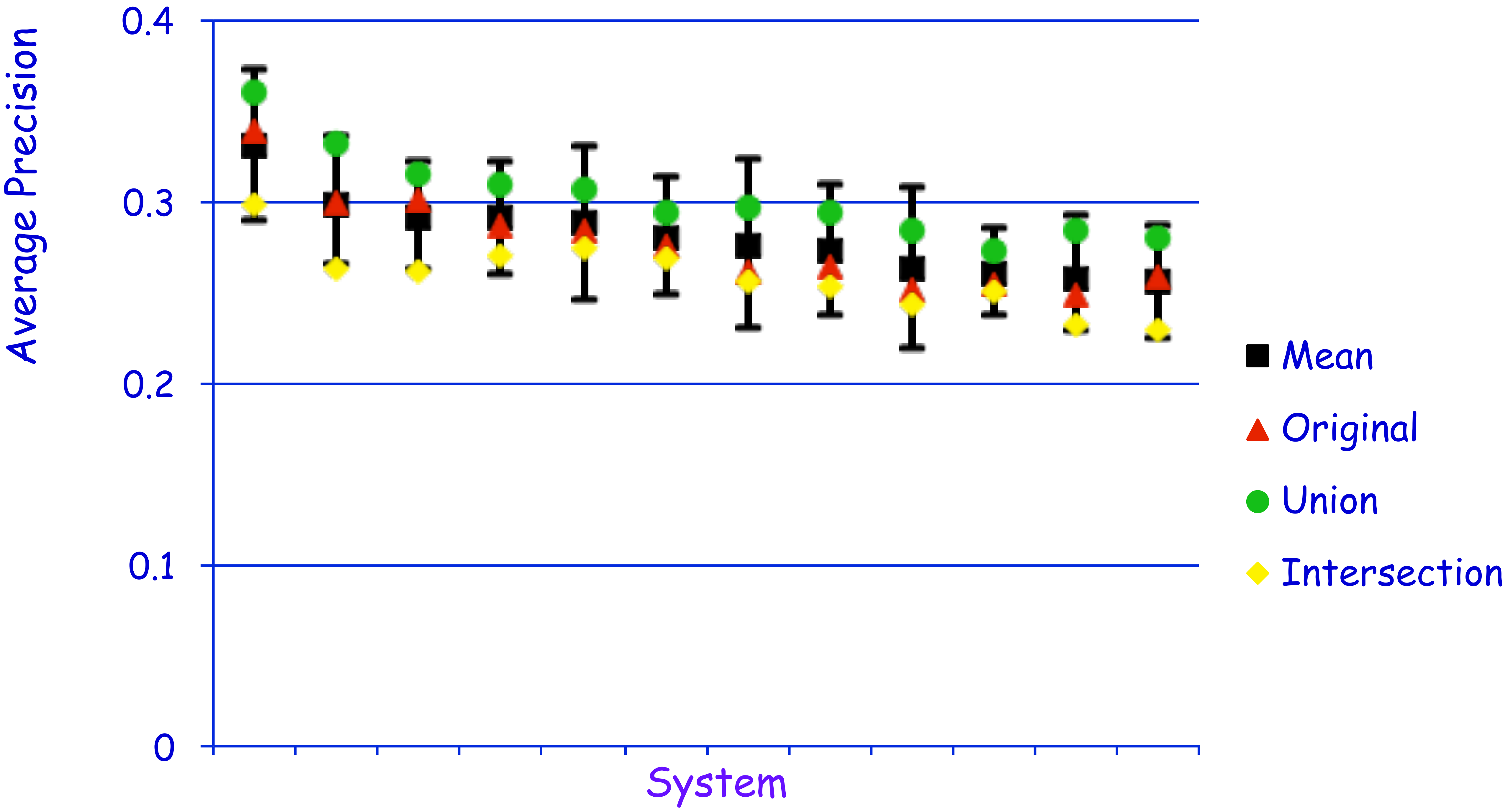
Robustness

- Study by Zobel [SIGIR-98] asks, are the TREC ad hoc collections biased against systems that do not contribute to the pools?
- Method:
  1. remove a group's runs from the pool.
  2. re-evaluate that group's runs using the residual pool, and measure the difference.
- His conclusion: the collections he examined were not biased.
- The relevance judgments were "sufficiently complete." (my phrase)

Variation of relevance

- Voorhees [SIGIR ’98] asks, what differences do we see measuring the same systems with different relevance judgments?
- Data: extra sets of judgments from TREC-4 and TREC-6.
- Method:
    - Measure systems using second judgment set.
    - Compare system scores as well as ranking of systems.
    - Construct new judgment sets by mixing and matching topics, and taking intersections and unions of judgments.
- Conclusion: scores do change, but the rank order of systems does not.

Stability of relevance judgments



Other Relevancy issues

- Relevancy is time and user dependent
  - Learning issues, novelty issues
  - User profile issues such as prior knowledge, reason for doing search, etc.
- TREC picked the broadest definition of relevancy for several reasons
  - It fit the user model well
  - It was well-defined and thus likely to be followed
  - Thousands of documents must be judged quickly
  - This creates a collection which can then be subset

Using existing test collections

- The advantage of using an existing test collection is not just the cost savings but the fact that there is training data, and results to compare with, and publications using the data
- Existing test collections from all of these evaluations are generally available: see the home site of these evaluations for info
- It is critical to read the full set of information about these test collections to understand their limitations

What are important issues here

- Does the user model on which the test collection was based "match" the user model of your experiment so the results are applicable?
- For cases where there are multiple test collections for a given user model (such as the TREC ad hoc task), are you using the best one?
- For example, the TREC ad hoc collections from TRECs 7 and 8 are generally considered the best ones to work with; similarly some of the earlier collections for given evaluations are less desirable than later ones.

What about other collections

- For non-English ad hoc collections, or ones for CLIR research, check out CLEF, NTCIR, FIRE and the 2002 TREC Arabic ones
- For other areas, such as patents (NTCIR), image retrieval (CLEF), video (TRECvid), structured data (INEX), look at those web sites
- In using any test collection, however, it is CRITICAL to read as much as you can find about this collection because often there are unexpected interactions between the collection and your experiment that need to be recognized

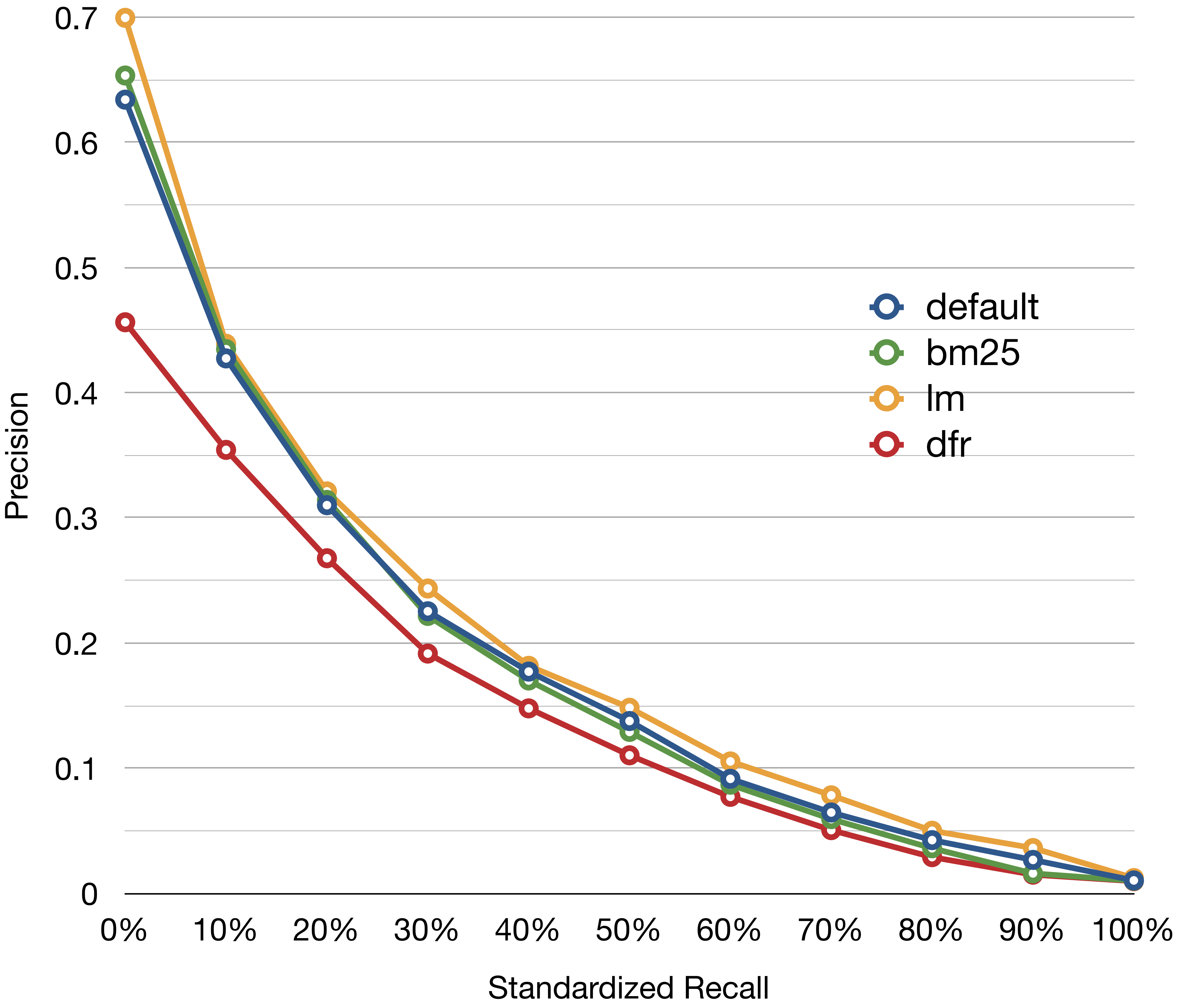
What about using the older collections such as TIME, CACM?

- This is generally a very bad idea!!
- All of them but TIME are abstracts rather than full text; we have moved beyond this
- As a learning exercise, it is OK to use the TIME collection, however any conclusions drawn from that collection need to be tested on the newer, larger collections
- In particular, it is unlikely that you will get a paper accepted using only the older collections; ideally it is best anyway to work with multiple collections to fully test ideas

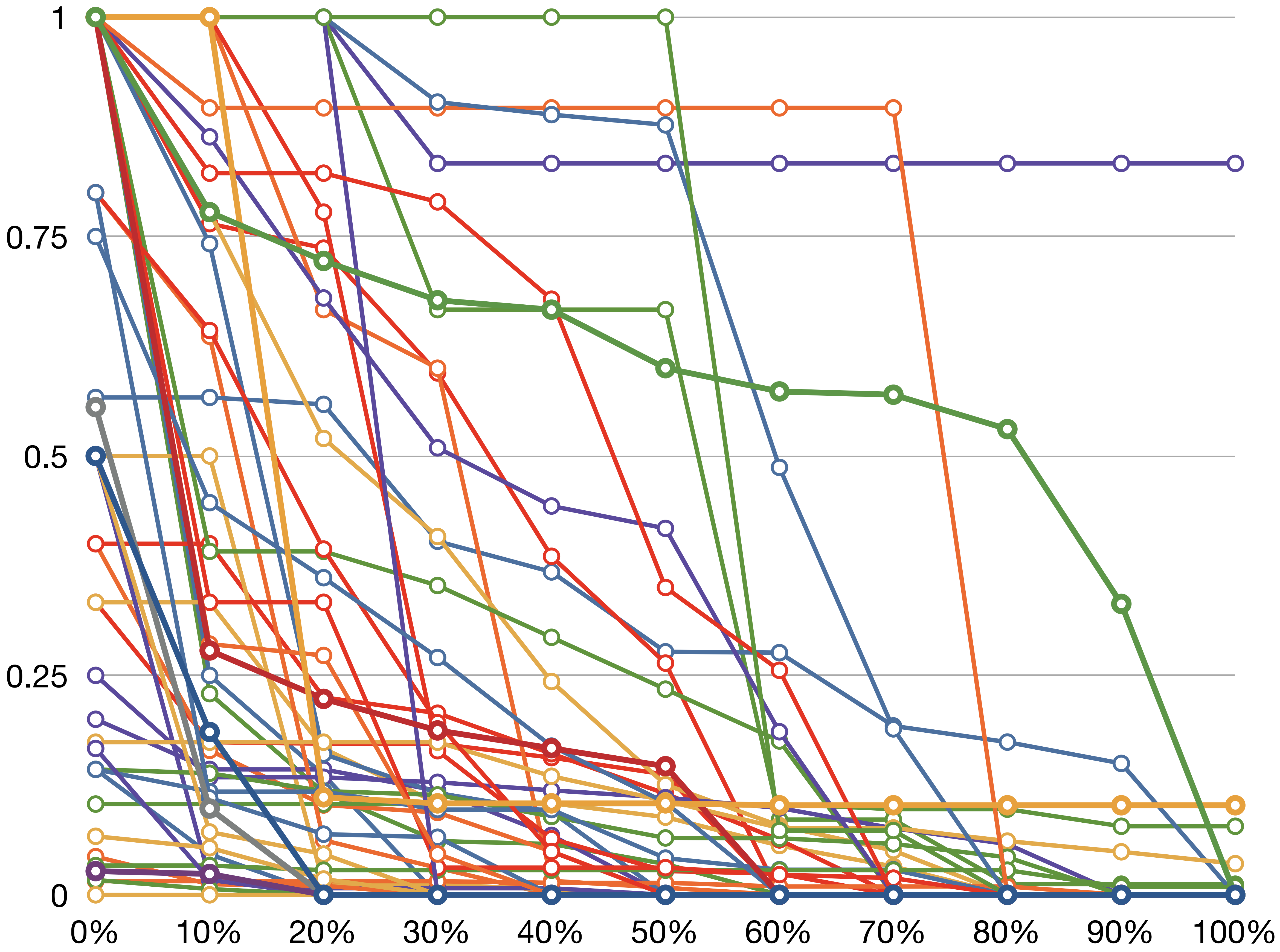
Beginner Evaluation Experiment

- Use TREC-6, 7, and 8 ad hoc test collections.
- Use an open-source search toolkit: Lucene
- Index the collection.
- Four search algorithms: default, lm, bm25, dfr
- Run each algorithm, taking the top 1000 hits for each topic.
- Use the trec_eval tool to measure the runs against the relevance judgments.

https://github.com/isoboroff/trec-demo


0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Precision
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
Standardized Recall
default
bm25
lm
dfr

"Default" run, topics 301-350



Evaluation experiment II

- Query expansion using pseudo-relevance feedback:
  - Perform initial query.
  - Extract high-value terms from top-ranked documents.
  - Expand initial query with those terms and do a second query.
- On average, this is a big win!
- But exhibits spectacular failures for some queries.
- See http://ir.nist.gov/ria/ for lots of data.

Cranfield in the Field

- The ad hoc collections model a scenario where the user formulates a single query, gets one results list, and wants all the answers.
- These collections may be useful in a deployed search setting for testing basic search capability and regression testing.
- But both for search-in-the-field, as well as more complicated user search tasks, we need to stretch the model a bit.

Warm-up Stretches

- The pooling approach used in TREC already stretches the Cranfield model, by not judging all the documents.
- Based on the ad hoc pooled collections, we have strong evidence that complete judgments are not necessary.
- How many do you need?
- How do you know you've got the right ones?

Number of topics needed

- Voorhees and Buckley [SIGIR ’02] ask, how many topics does a test collection need?
- Method (following Sakai [SIGIR ’06]):
    - draw two equally-sized samples of topics with replacement from the collection.
    - evaluate systems used to build that collection using each sample.
    - compare the rank order of systems based on each sample.
    - determine the minimum number of topics in the sample such that the probability of two systems exchanging places between the rankings is < 0.05.
- Conclusion: around 25 is ok for the ad hoc task, but you can’t tell without the larger sample to draw from.

How many relevance judgments?

- As of the mid-2000s, most test collections were between 500k and 2M documents, with 5-10% judged for some topic.
- Would the pooling assumption break if the collection gets very big?
- The odds get very good that your system won't retrieve a judged document.
- Buckley and Voorhees [SIGIR '04] proposed a measure, *bpref*, designed to be robust if relevance judgments are very incomplete.
- Assumes random incompleteness.

Adventures in Sampling

- Soboroff et al [SIGIR ’01] showed that randomly sampling the pool, with no relevance judgments, can predict large parts of the system ranking.
- Others followed this down the path of trying to build test collections with small relevance judgment sets.
    - Pavlu, Aslam, et al: voting strategies, statAP method
    - Yilmaz et al: inferred AP (infAP) method
    - Carterette: minimal test collections method
    - Buttcher et al: small samples + machine learning
- Bottom line: lots of good methods for reducing your judgment costs in test collection building.

Bias

- Whenever you’re drawing samples, you should be concerned about bias.
- Pooling itself is intentionally biased: given a reasonable set of decent search systems, the pool will be sufficiently complete.
- Experiments in subsampling generally assumed that sampling was random and unbiased.
- Buckley et al [IR 10;6, ’07] found an example where a system was very different from the pack, and the resulting test collection would have been biased against it by Zobel’s method.
- Hypothesis: as collections grow, bias becomes more likely.

Other kinds of tasks

- Cranfield is useful for measuring many aspects of the search “pipeline”.
- Also can be fit to other tasks besides ad hoc search:
  - Filtering: static query, streaming collection.
  - Known-item search: only one right answer.
  - Special kinds of relevance: key pages, homepages, highly relevant documents.
  - Diversity ranking: each query has many meanings.
  - Query sessions (without learning or context).
  - ...

Designing a test collection

- Understand the user’s task.
- Use a naturalistic collection of documents.
- Study queries and information needs from real users.
- Design a topic set based on the above.
- Define relevance from the task perspective.
- Get real users to judge documents for relevance.
- Analyze failures.
- Never trust an average.

Is there anything it can't do?

When can't you use Cranfield?

- When document relevance is not independent. (learning over queries, novelty detection, ...)
- When people can't consistently or reliably judge the task. (searches over time, complicated relevance concept, ...)
- If the systems are very poor, pooling can establish their rank order but not produce a reusable test collection.

Remember users?

- The Cranfield paradigm is incredibly powerful.
- However, like all good laboratories, it is only an abstraction of the real world.
- How do users start a search? How do they proceed through it?
- Do differences we see in test collection experiments translate into more successful users?
- What gain is needed in the lab to see a corresponding gain for the user?
- What level of effectiveness is "good enough" for the task?

Where’s the money?

- Hersh et al. [SIGIR ’00], Turpin and Hersh [’01], Turpin and Scholer [’06] ask: do improvements in a batch experiment yield improvements for users?
- Often, no!
- Differences among systems may be significant but small.
- Significant differences in user studies are hard to observe.
- Humans are really good at using tools beyond their capabilities.

User study example

Block	1 - pre-treatment				2 - treatment				3 - post-treatment				Group
Position #	1	2	3	4	5	6	7	8	9	10	11	12	
System	standard				standard				standard				Control
	standard				Inconsistently low rankings				standard				ILR
	standard				consistently low rankings				standard				CLR

- Smith and Kantor [SIGIR 08] ask, **what do searchers do** to maximize the performance of search systems?
- 12 topics, 36 test subjects, 3 search systems.
- Found that users of the poor system would issue more queries, and achieve equivalent results to the standard system.

User study details

- Three effects to be handled:
  - search topics vary in difficulty,
  - searchers have different skills, abilities, and knowledge,
  - searchers in the experiment will learn as they proceed.
- The experimental design keeps the system effect separate from these three confounding effects.
- Analysis of variance based on linear models of combinations of possible effects.

User study take-away

- Smith and Kantor could show that despite obvious differences in **system effectiveness**, users of those systems adapted their behavior to overcome them.
- Because Cranfield does not address query formulation or search process over a session, you cannot learn this from a test collection experiment.
- Challenge: can you design a search algorithm that makes users more effective than they can make themselves?

Consecutive searches

- Carterette and Kanoulas are running the TREC Session Track to try to bridge Cranfield out to session behavior.
- Models query transitions within a session.
- Goal: develop a test collection approach where system effectiveness can be reliably measured given the session environment.

Evaluating Interactive IR

- Pia Borlund and colleagues, 2000-
- User study manifesto!
- IIR evaluation requires that we:
  - examine interaction by actual users of the system.
  - handle individual and dynamic information needs.
  - compare simulated and real task scenarios.
  - use task-centered measures not based on binary relevance.

Simulated work tasks

- Similar to a TREC topic in spirit, the simulated work task describes:
  - the source of the information need,
  - the environment of the situation,
  - the problem to be solved.
- Goal: to engage the test subject in wanting to achieve the goal of the simulated work task.
- Developed with pilot testing before actual experiment.
- Further, with actual users of the system, actual work tasks can be interleaved with simulated ones.

Exploratory Search

- Nick Belkin's ASK study (1980)
  - Some searches have a well defined topic and come from a well-defined problem.
  - Even then, searches may be looking for further background information.
  - Search topic is specific, but the problem is not well-defined.
  - Problem is specific, but search topic is not well-defined.
  - Problems and topics not well-defined.
- Exploratory search tries to develop (and measure!) systems that assist the user in understanding the larger problem, developing search topics, knowing the unknown

Collaborative Search

- When was the last time you worked on something by yourself?
  (probably in school)
- Problems are shared among a larger group.
- Search topics probably overlap.
- Members of the group have a shared context, but differing levels of knowledge, experience, skill.
- How to improve the search success of the members so that the group achieves a goal.

Search given Context

- Search never happens in a vacuum.
- Context (following [Ruthven 'xx]):
  - task (what is my goal?)
  - social (who knows what I need to know?)
  - personal (what do I know? how do I feel?)
  - physical (where am I? when do I need to know?)
  - environmental (what is appropriate for the current situation?)
- A lot of discussion on how to model context.
- A little experimental work on how change in context affects changes in system effectiveness.

Search without queries

- If I have a good representation of context, perhaps there is no need for the user to formulate a query.
- Mobile scenario: it’s dinnertime, I’m in my car, two friends are with me, where shall we eat?
- Contextual guesses require diversity to maximize effectiveness.

Text features, relevance models, query matching, combining features, algorithms	User's task, and user's broader goals surrounding the tool
Information environment that supports creation and maintenance of content.	Interface capabilities that support the user's task

Conclusion

- "If you cannot measure it, you cannot improve it."
- Information retrieval is tricky to measure.
- All measurement requires good scientific method.
- There is a lot of good shoulders to stand on.