Information Retrieval

Evaluating IR Systems

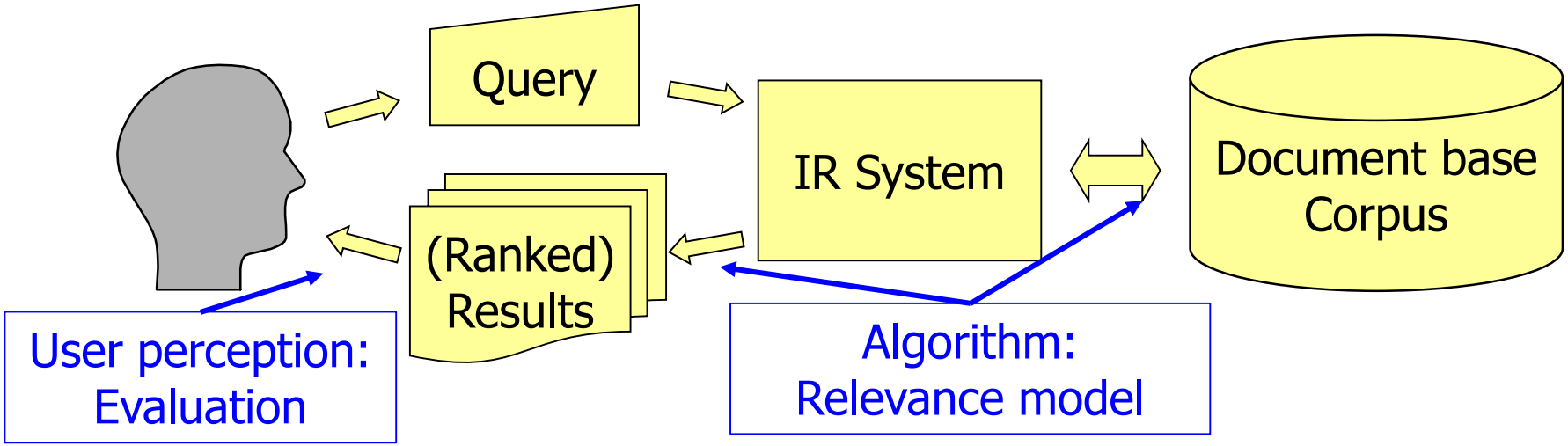
Ulf Leser

Content of this Lecture

- Evaluating IR Systems
- Real-Life Example: VIST

The Informal Problem

- IR problem: Help user in quickly finding the requested information within a given set of documents
- Central : How helpful is a given set for a given query?
  - We need an evaluation method
  - Important to compare different IRS / algorithms
  - Strong subjective component: "Information need"



First Approach: Binary Evaluation Model

- We assume a fixed corpus D as given
- We assume that for a query q and any $d \in D$, somebody (the truth) determines whether d is relevant for q or not
  - An expert? An average user?
  - Binary decisions: No ranking (for now)
  - Think of the decision what to display on the first result page
  - We call this set T(q)
  - This is a gold standard
    - Costly to obtain, probably subjective – we'll meet the topic again
- The IR system (IRS) returns a set X(q) of docs it considers relevant for q
- How to compare T(q) and X(q)?

Classifying Documents

- More formally
  - Let T be the set of all truly relevant docs for q
  - Let X the set of all IRS-computed docs for q

	Truth: relevant	**Truth: not relevant**
IRS: relevant	True positives	False positives
IRS: not relevant	False negatives	True negatives

- We can partition
  - $T = TP \cup FN$
  - $X = TP \cup FP$

Precision and Recall

I'm lazy – should be |TP| / (|TP|+|FP|)

- Precision = TP/(TP+FP)
  - Fraction of relevant/correct answers in X
- Recall = TP/(TP+FN)
  - Fraction of correct answers from T actually returned?
- The perfect world

	Truth: Relevant	Truth: Not relevant
IRS: Relevant	A	0
IRS: Not relevant	0	B

Example

- Let |D| = 10.000, |X|=15, |T|=20, |X∩T|=9

	Truth: Positive	Truth: Negative
IRS: Positive	TP = 9	FP = 6
IRS: Negative	FN = 11	TN= 9.974

  - Precision = TP/(TP+FP) = 9/15 = 60%
  - Recall = TP/(TP+FN) = 9/20 = 45%

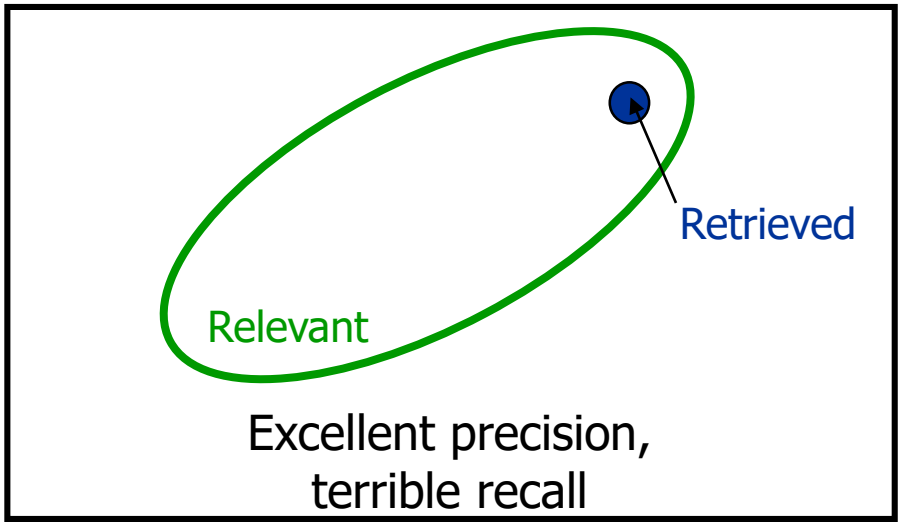
- Assume another result: |X|=10, |X∩T|=7

	Truth: Positive	Truth: Negative
IRS: Positive	TP = 7	FP = 3
IRS: Negative	FN = 13	

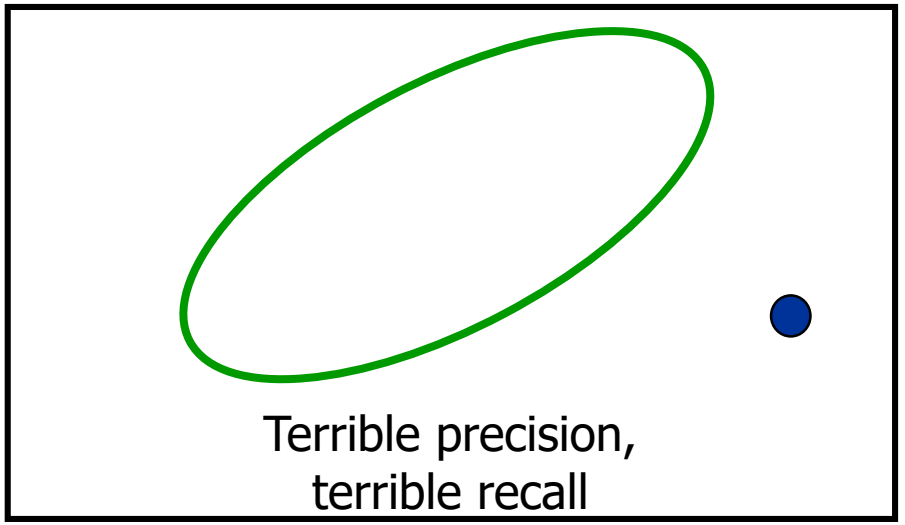
  - Precision: 70%, recall = 35%

A Different View

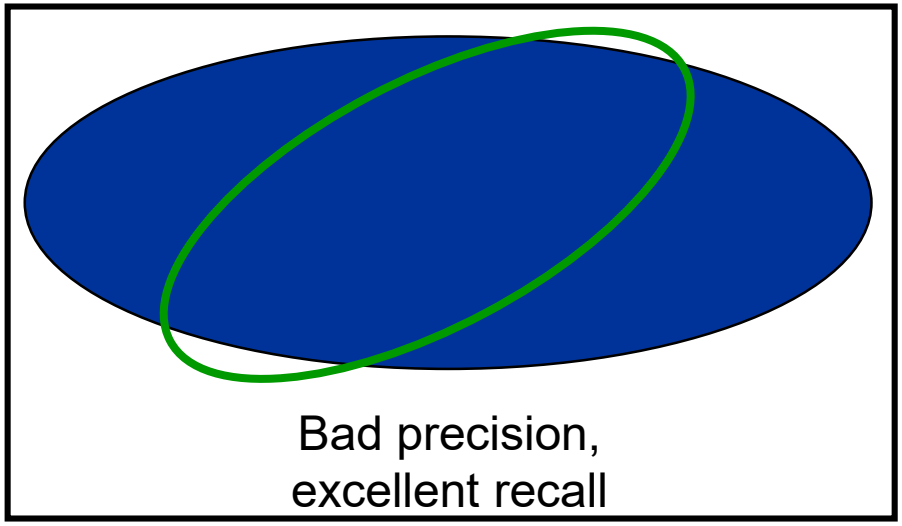
Quelle: A. Nürnberger, VL IR



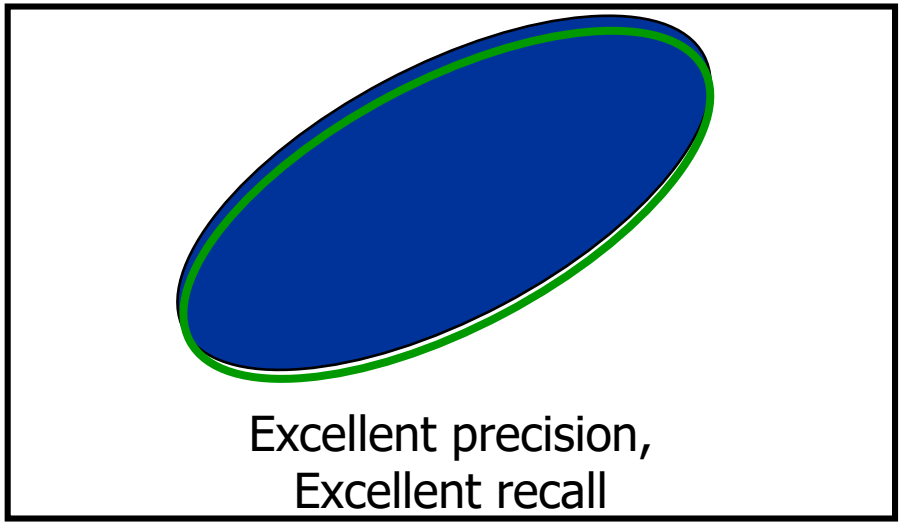
Excellent precision,
terrible recall



Terrible precision,
terrible recall



Bad precision,
excellent recall



Excellent precision,
Excellent recall

Trade-off

- Trade-off between precision and recall
- Most methods compute a similarity score between docs and q
  - Assume a reasonable score: High sim-score implies high probability of being relevant
  - Methods use a threshold t to enforce a binary decision

Relevant?	Ranked result	Class
T	d	TP
T	d	TP
F	d	FP
T	d	TP
F	d	FP
T	d	TP
T	d	TP
F	d	FP
F	d	TN
T	d	FN
…	…	…
F	d	TN
F	d	TN
T	d	FN
F	d	TN
F	d	TN
F	d	TN
F	d	TN

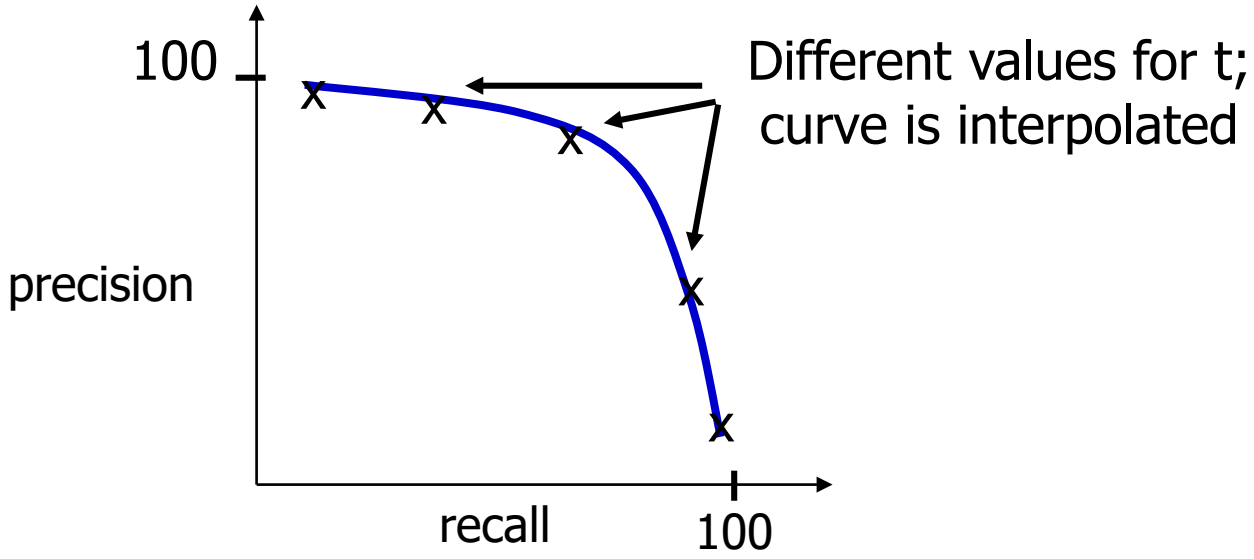
Relevant?	Ranked result	Class
T	d	TP
T	d	TP
F	d	FP
T	d	TP
F	d	**TN**
T	d	**FN**
T	d	**FN**
F	d	**TN**
F	d	TN
T	d	FN
…	…	…
F	d	TN
F	d	TN
T	d	FN
F	d	TN
F	d	TN
F	d	TN
F	d	TN

Trade-off

- Trade-off between precision and recall
- Most methods compute a similarity score between docs and q
  - Assume a reasonable score: High sim-score implies high probability of being relevant and vice-versa
  - Methods use a threshold t to enforce a binary decision
  - Increase t: Less results, most of them very likely relevant
    Precision increases, recall drops
    Set t=1: P ~ 100%, R ~ 1/|T|
  - Decrease t: More results, some might be wrong
    Precision drops, recall increases
    Set t=0: P = |T|/|D|, R = 100%

Precision / Recall Curve

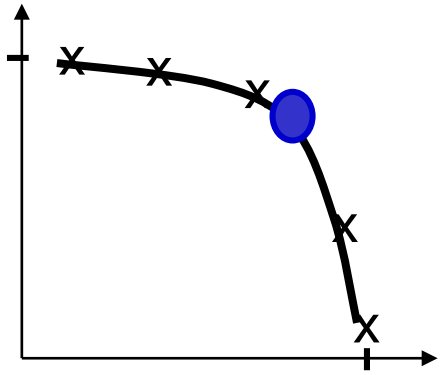
- Sliding the threshold t gives a precision/recall - curve

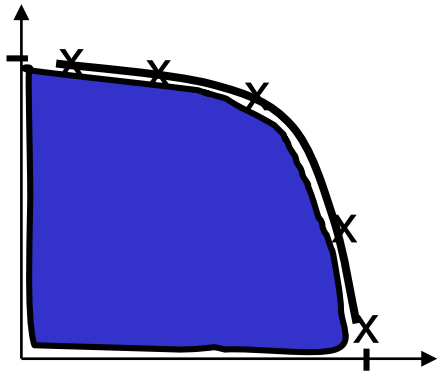


- Typical goal of IRSs: Best point within curve
- But what is "best"?

F-Measure

- Defining one measure instead of two
  - E.g. to rank different IR-systems
- Classical: F1-Measure = 2*P*R / (P+R)
  - F-Measure is harmonic mean between precision and recall
  - Favors balanced P/R values
  - Fx-Measure: $(1+x^2)*P*R / (x^2*P+R)$
    - Recall x-times as important as precision
- Alternative: Area-under-the-curve, (AUC)
  - Independent of concrete threshold t
  - But real IRS need a t …





Accuracy

	Truth: relevant	Truth: not relevant
IR: relevant	TP	FP
IR: not relevant	FN	TN

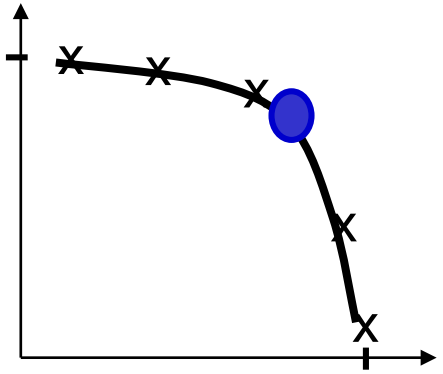
- Accuracy= (TP+TN) / (TP+FP+FN+TN)
  - Which percentage of the system's decision were correct?
  - Makes only sense with small corpora and large result set
  - Typically in IR, TN >>> TP+FP+FN
  - Thus, accuracy is always excellent (~0,99999…5)
- Used in problems with balanced sets of TN / TP
  - E.g. typical classification evaluations

Where are we?

- For some q, produce gold standard T + compute answer X

	Truth: Relevant	Truth: Not relevant
IRS: Relevant	TP	FP
IRS: Not relevant	FN	TN

- Popular measures
  - Precision = TP/(TP+FP)
  - Recall = TP/(TP+FN)
  - F1-Measure = 2*P*R / (P+R)



- But: Which query? Which expert? Which gold standard?

From user/query to users/queries

- We need to look at a range of different queries
  - Compute average P/R values over all queries
  - Of course, stddev is also important
- We need to look at different users
  - Different users may have different thoughts about what is relevant
  - This leads to different gold standards
  - Compute inter-annotator agreement as upper bound
- Who can judge millions of docs?
  - Evaluate on small gold standard corpus
    - But: Extrapolation difficult: Are the properties of application/corpus really equal to properties of GS?
  - Use implicit feedback, e.g. click-through rates in top-K results

Micro- versus Macro Averages

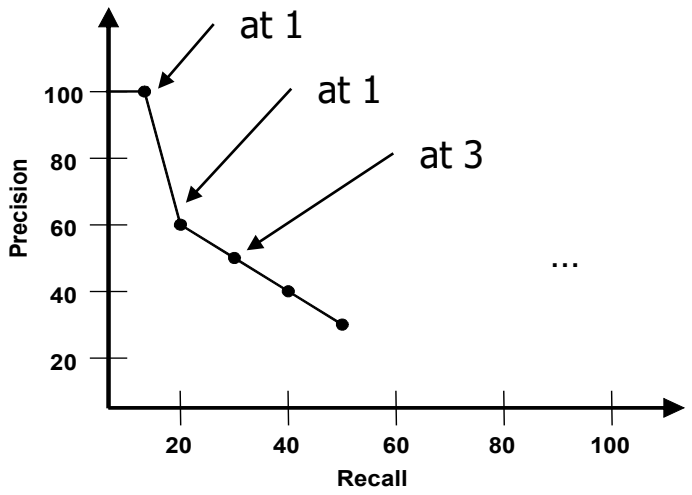
- Evaluating different queries: Beware different sizes of T
  - Larger T → larger TP/FP/FN → stronger impact on the average
- Two ways of computing an average over m queries
  - Macro-Average: Average P and R over P_1, R_1, … values of queries
  - Micro-Average: Compute P and R over all TP_1, FP_1, … values

$$\frac{\sum_{i=1..m} P_i}{m} \neq \frac{\sum_{i=1..m} TP_i}{\sum_{i=1..m} TP_i + \sum_{i=1..m} FP_i}$$

- Comparison
  - Micro-Average implicitly weights queries with result size
  - Macro-Average is less affected by outliers (with large result sizes)
  - Be cautious when results different largely
    - Heterogeneous query set

Evaluating Rankings

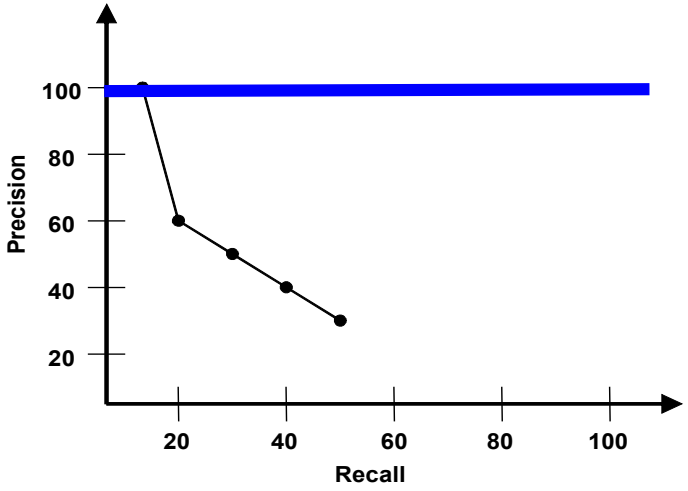
- Recall: Real IRS compute ranked answers (sim-score)
- Assume we still have a binary gold standard
- Typical approach: "P/R/F at k"
  - Move a pointer down the sorted list
  - Consider docs above the pointer as set X
  - Gives one P/R value per list position k

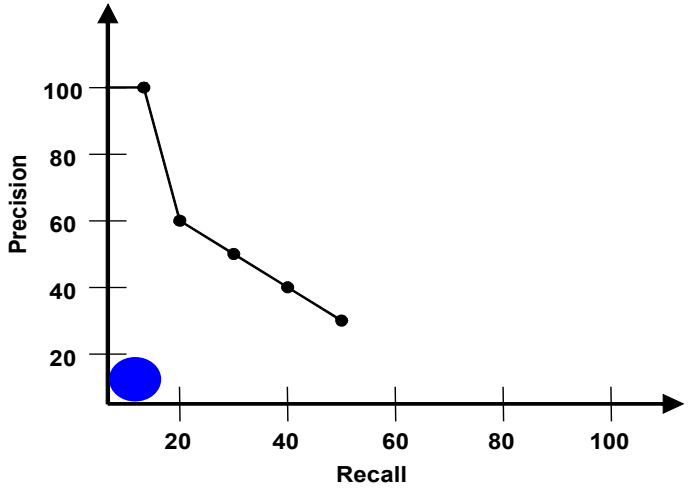


- Assume there are 10 truly relevant docs and result = {**5**,9,**7**,67,9,**4**,17,3,90,**21**,…}
- At 1st position, IR scores P=100 and R=10 (1 out of 10)
- At 2nd position, P=50, R=10
- Pos 3: 66/20
- Pos 6: 50/30
- …

Evaluating Rankings

- Recall: Real IRS compute ranked answers (sim-score)
- Assume we still have a binary gold standard
- Typical approach: "P/R/F at k"
  - Move a pointer down the sorted list
  - Consider docs above the pointer as set X
  - Gives one P/R value per list position k





Advanced: Evaluate Rankings with Rankings

- Assume users also have several grades for „relevance“
  - Lickert-scale: Very relevant, quite relevant, neutral …
- Compare a user ranking with a IR-ranking
  - We need a distance function for rankings
    - E.g. Kendall-Tau: Percentage of pairs-wise disagreements
- Users with different rankings: What is the GS-ranking?
  - Median ranking: ranking with least total distance to user rankings
- Things get difficult when rankings may have ties, different rankings rank different sets of objects, or rank-distance should be included
  - Median-ranking becomes NP-hard
  - See: Brancotte et al. (2015). "Rank aggregation with ties“, VLDB

Critics

- Precision and recall are not independent from each other
- F1 gives equal weight to precision and recall – why?
- Both assume a static process – no user feedback, no second chance
  - Does not evaluate the process-view of IR
- Both ignore or average over many important aspects
  - Documents might be relevant yet boring (e.g. duplicates)
  - Different users find different results interesting (personalization)
- Both rely on gold standards
  - Which often don't exist / are very expensive to create
  - Which might have been defined with a different conception than that of an average user

Universal and often very Difficult Issue

- Assume a medical test for some disease producing a score
  - E.g. PCR tests for mRNA produce a "cT" value (crossing threshold)
  - High value: Low concentration of mRNA; low value: High concentr.
  - Test result: cT above a predefined threshold
- In mass tests (screenings) – how to set the threshold?
  - Low threshold – Higher precision, lower recall
    - Fewer false alarms, more missed diseases
  - High threshold – Lower precision, improved recall
    - More false alarms (unnecessary surgery?), fewer missed diseases
- Very difficult ethical question
  - All mass screenings require an ethically difficult decision
- Ask your doctor about sensitivity / specificity of a test

Why „F"-measure [Dave Lewis]

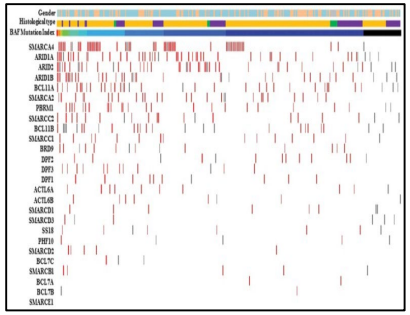
- http://metaoptimize.com/qa/questions/1088/f1-score-name-origin
  - Why is the F1 score called F1?
  - Yes, it was a bizarre lucky break! I was on the MUC program committee, and there was pressure for a single measure of how effective a system was. I knew of the E-measure from Van Rijsbergen's textbook on Information Retrieval, so thought of that.
  - However, *lower* values of E are better, and that just wouldn't do for a government-funded evaluation. I took a quick look in the book, and mistakenly interpreted another equation as being a definition of F as 1-E. I said great, we'll call 1-E the "F-measure". Later I discovered my mistake, but it was too late. Still later, I was reading Van Rijsbergen's dissertation, and saw that he had used E and F in the same relationship, but that hadn't made it into his textbook. Whew.
  - It's a somewhat unfortunate name, since there's an F-test and F-distribution in statistics that has nothing to do with the F-measure. But I guess that's inevitable with only 26 letters. :-)

Content of this Lecture

- Evaluating IR Systems
- Real-life Example: VIST – Variant Information Search Tool

Molecular Tumor Boards

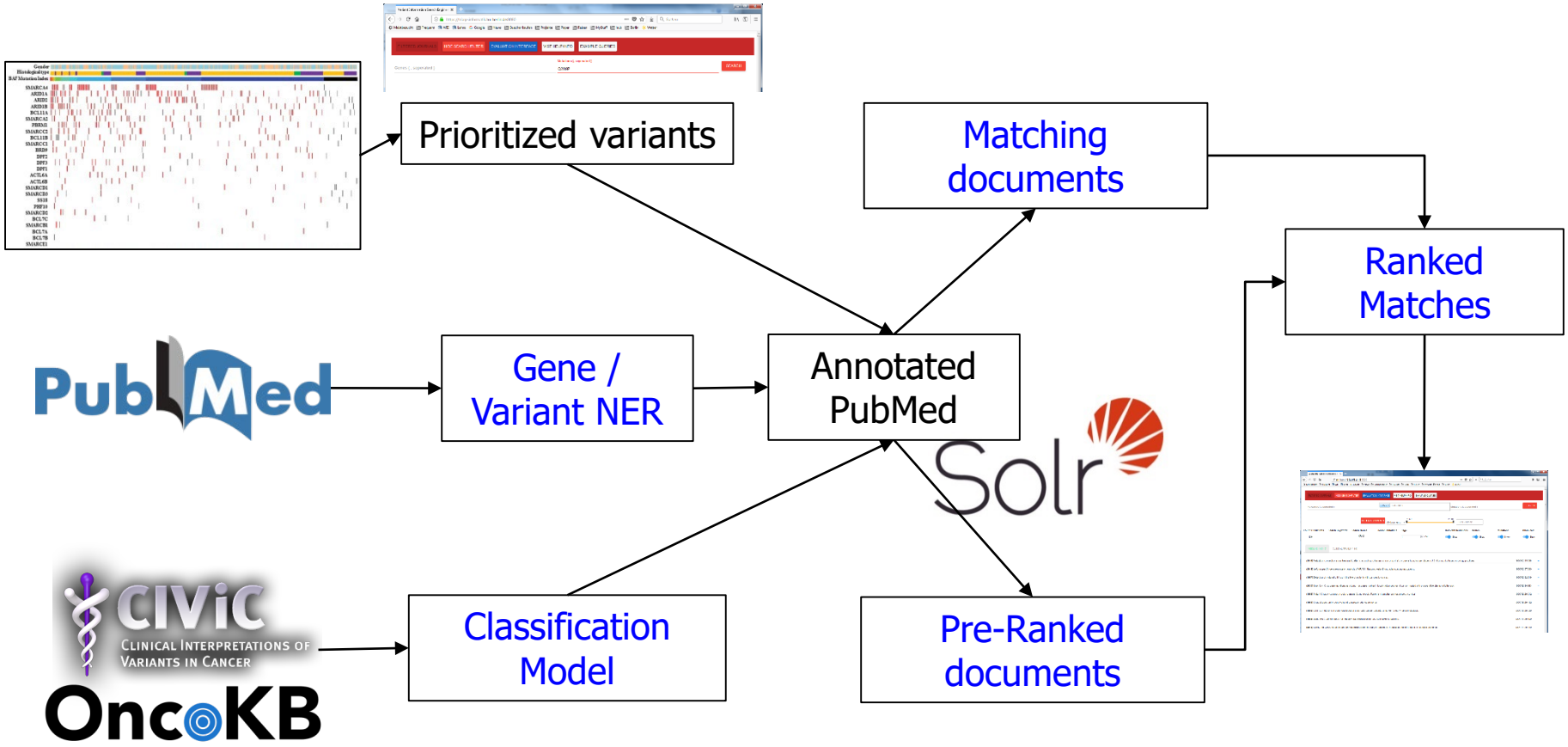
- Interdisciplinary team discussing individual patients
- Decisions based on molecular data, esp. genomic variants
  - Genome/exome/panel, transcriptome, proteome, epigenome, …
- Given a patient's set of variants – Suggest treatments



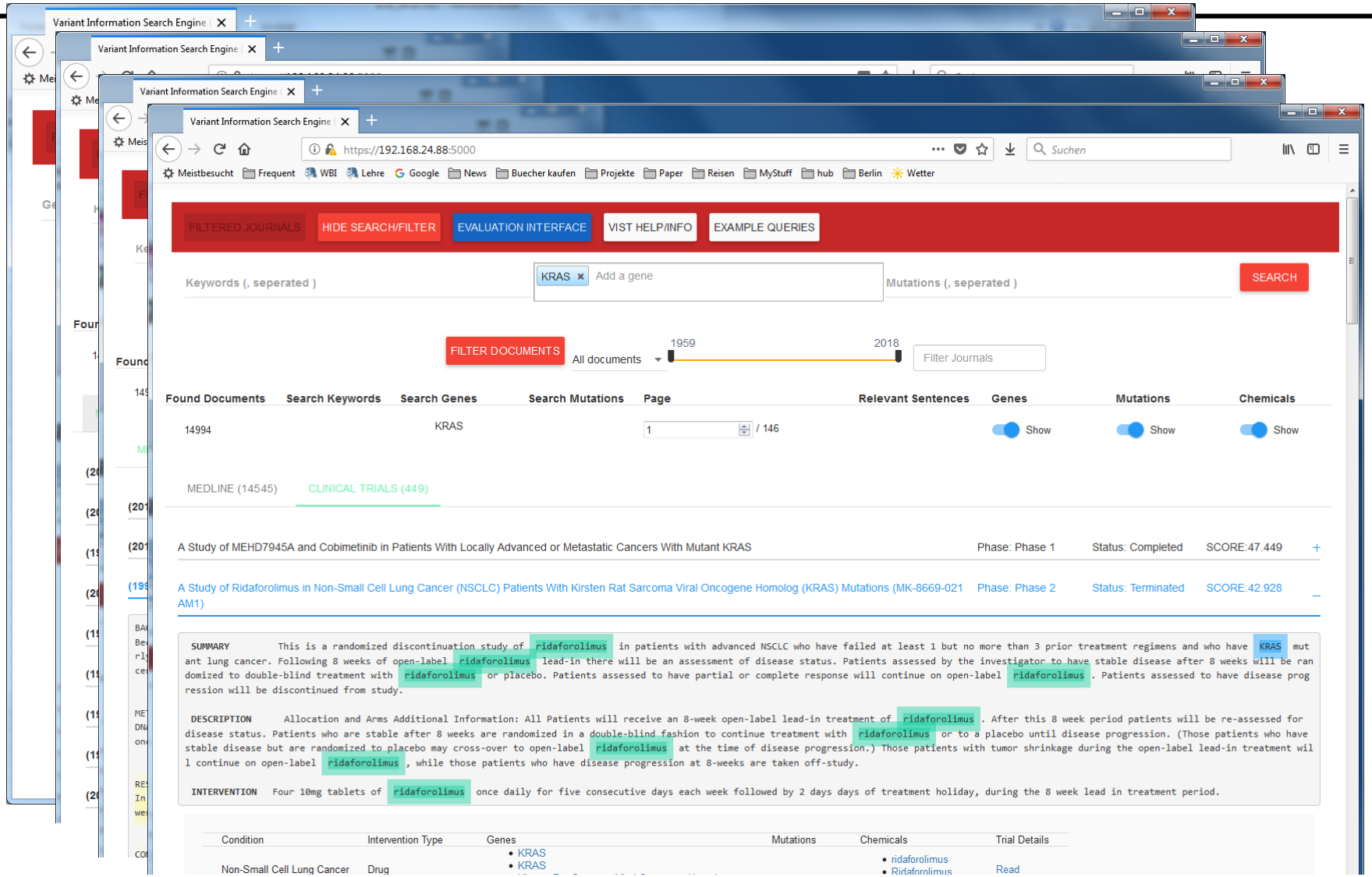
Clinically Relevant?

- Clinicians search information for specific variants / genes with direct impact on treatment of a specific type of cancer
  - Pre-clinical research not in focus (mice, cell lines, …)
- Central issue: Filter/rank by clinical relevance
- VIST: Use classifier trained on clinically relevant documents
  - We compared various scoring and classification methods
  - See paper [Seva et al., BMC Bioinformatics, 2019]

VIST Architecture



VIST – Variant Information Search Tool



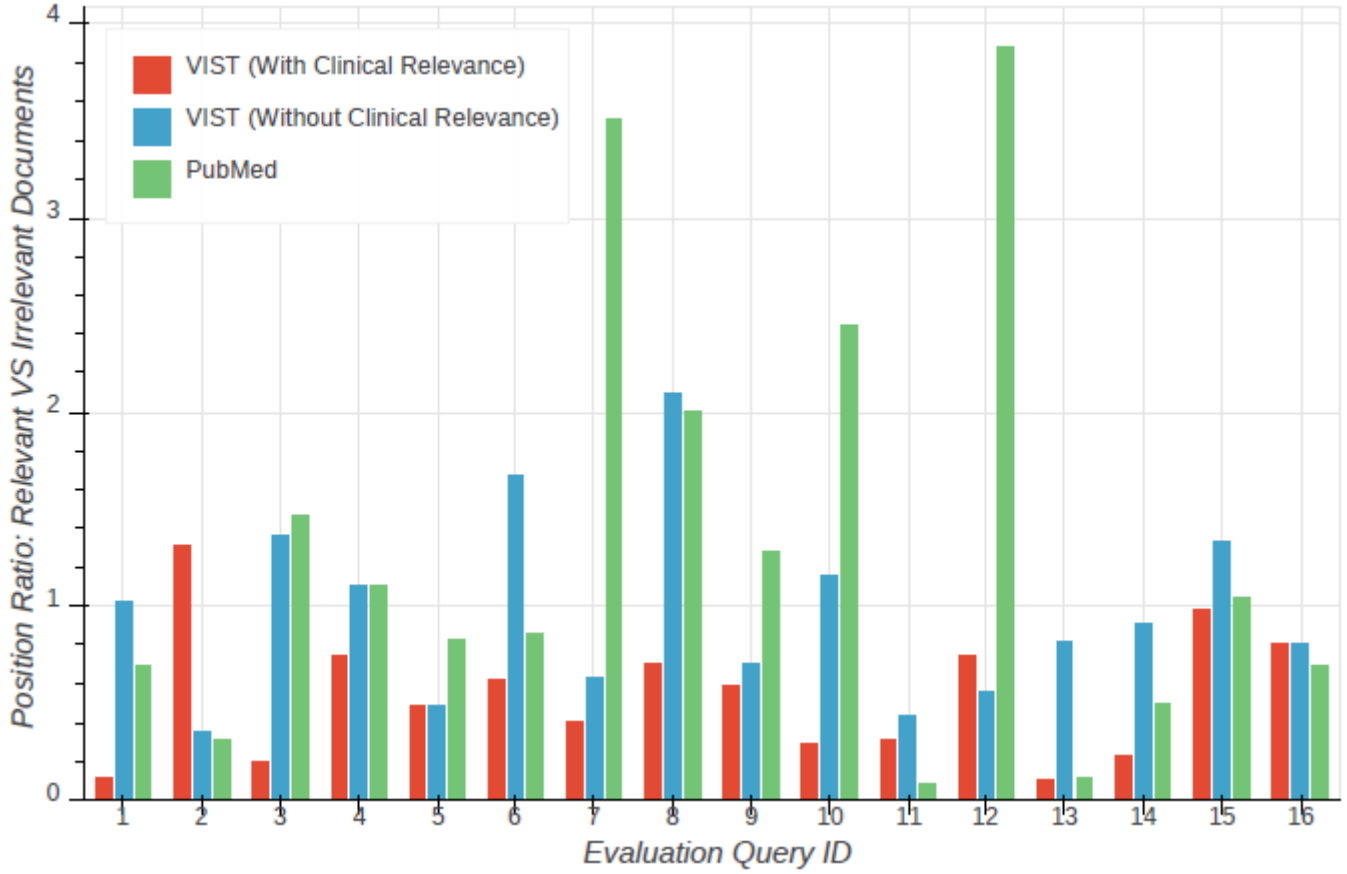
Evaluation – Ask the Expert

- 20 variants, 10 docs per variant, 4 medical experts
- 188 assessments (5-point Lickert scale)
  - ~40% (highly) relevant docs
  - ~40% matching yet clinically irrelevant docs
  - ~20% unknown / wrong NER
- Issue: Low inter-expert agreement
- Filtering "difficult" cases results in 101 assessments

Query	PMID	EV1	EV2	EV3	EV4	LOOSE	STRICT
3	22496619	2	3	3	3	irrelevant	unknown
3	24549645	3	2	3	3	irrelevant	unknown
3	24768329	1	2	1	3	relevant	unknown
3	26125448	1	1	1	1	relevant	relevant
3	26497685	4	3	3	3	irrelevant	irrelevant
3	26662311	1	1	1	2	relevant	relevant
3	26820161	4	2	3		unknown	unknown
3	26855149	2	2	2	2	relevant	relevant
3	28153088	2	3	3	3	irrelevant	unknown

1 Highly relevar
2 Relevant
3 Match but irr
4 Irrelevant
Unknown

VIST versus SOLR versus PubMed



- More evaluations on more corpora in the paper

Discussion

- "Better" very difficult to show
  - Difficult evaluation: Unclear gold standards
  - Difficult baseline: Experts use additional keywords when searching PubMed (how to model?)
- Does it carry over into practice?

Self Assessment

- Give a definition of recall, precision, and accuracy
- Which relevance models produce a Boolean answer, i.e., no ranking?
- What is "recall at k"? How could we turn this into a single value?
- What is the difference between micro and macro average
- How can we cope with the fact that different users may have different expectations for the same query?