

Text Technologies

Evaluation

Victor Lavrenko

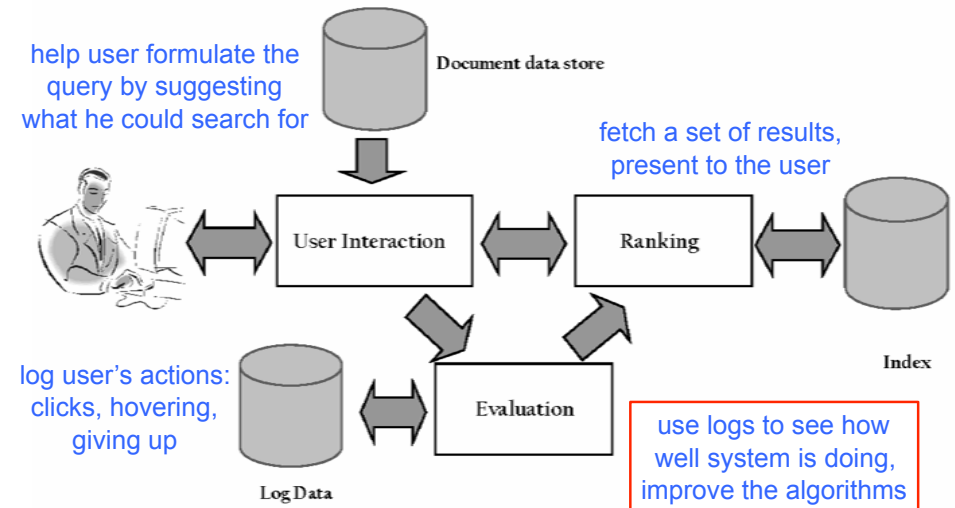
Copyright © 2011 Victor Lavrenko

Overview

- Evaluation: why and what
- Cranfield paradigm
 - topics and corpora
 - relevance judgments
- Set-based evaluation
- Ranked evaluation
- Significance tests

Copyright © 2011 Victor Lavrenko

Search Process



Evaluation

- How do you decide if a search system is good?
- Effectiveness:
 - does your system find good results?
- Efficiency:
 - how fast does it find them?
- Other considerations:
 - do the users like the experience?
 - easy to formulate the query?
 - easy to browse the results?

Copyright © 2011 Victor Lavrenko

Efficiency

- Indexing time (elapsed / CPU)
 - milliseconds it takes to add doc to index
- Index size (as % of raw data)
 - necessary storage for index files
 - temporary space used during indexing
- Query throughput
 - average number of queries per second
- Query latency
 - milliseconds user has to wait for results

Copyright © 2011 Victor Lavrenko

Effectiveness

- Does system A find good results?
- Set up an interactive study
 - group of users, specific task, information need
 - crucial to have a baseline (system B)
 - observe users, elicit feedback
 - did users of A complete task faster? more often?
 - differences in results: greater coverage? more aspects?
 - which aspects of A,B the users liked / disliked?
- Expensive, task-specific, difficult to reproduce
 - though ultimately there is no substitute

Copyright © 2011 Victor Lavrenko

Automated evaluation

- No user in the loop → test early, test often
 - our intuition about what works is often very wrong
 - phrases, WordNet, “core” terms, linguistic processing
 - term weighting, massive expansion, connectedness
 - success stories: IR, MT, ASR...
- Cranfield paradigm:
 - fixed set of queries (topics)
 - fixed set of documents (corpus)
 - fixed set of relevance judgments
 - effectiveness measure: results = relevant docs ?

} test collection

Copyright © 2011 Victor Lavrenko

Topics and corpora

- Topics
 - intended to mimic real information-seeking tasks
 - gov. information analysts, patent officers
- Corpora
 - news, scientific, legal, patents, web-pages
 - beyond text: speech, images, videos, DB records
- Annual competitions in IR (blind evaluation)
 - US: Text REtrieval Conference (TREC) – 117 groups
 - EU: Cross-Language Evaluation Forum (CLEF)
 - Asia: NTCIR

Copyright © 2011 Victor Lavrenko

Topics and corpora: example

Topic example:
TREC query 794

<title> pet therapy

<desc> Description:

How are pets or animals used in therapy for humans and what are the benefits?

<narr> Narrative:

Relevant documents must include details of how pet- or animal-assisted therapy is or has been used. Relevant details include information about pet therapy programs, descriptions of the circumstances in which pet therapy is used, the benefits of this type of therapy, the degree of success of this therapy, and any laws or regulations governing it.

Research corpora:

Collection	Number of documents	Size	Average number of words/doc.	Number of queries	Average number of words/query
CACM	3,204	2.2 Mb	64	64	13.0
AP	242,918	0.7 Gb	474	100	4.3
GOV2	25,205,179	426 Gb	1073	150	3.1

ClueWeb 1,040,809,705 25,000 Gb

Copyright © 2011 Victor Lavrenko

Relevance judgments

- Which documents are relevant for a query
 - usually binary: {relevant, non-relevant}
 - sometimes graded (more expensive to obtain)
 - multiple annotators: reduces accidental mistakes
 - issues: who? instructions? level of agreement?
- Exhaustive: judge **every** document
 - usually infeasible (25m docs x 150 queries)
 - occasionally done for a new task (e.g. TDT1)
- Why not judge just the top 10 / 100?
 - notion of *coverage* important for lawyers, analysts
 - want to reuse judgments to test new algorithms
 - which will find docs noone else found in their top 10/100

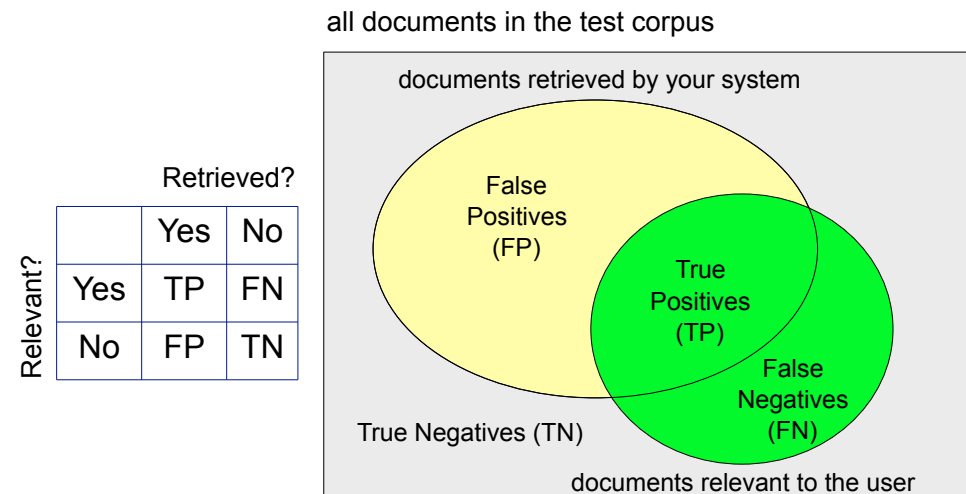
Copyright © 2011 Victor Lavrenko

Relevance judgments: sampling

- Pooled (used in TREC)
 - top *k* docs from every participating system (50-200)
 - merge into a pool, remove duplicates
 - randomize, present to annotators
- Search-guided
 - run query, read until convinced no more rel. docs
 - re-formulate query using found rel. docs, repeat
- Sampling
 - estimate bounds on the size of relevant set

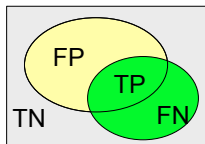
Copyright © 2011 Victor Lavrenko

Evaluation: building blocks



Copyright © 2011 Victor Lavrenko

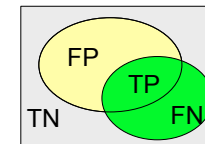
Precision and Recall



- Precision
 - proportion of retrieved set that are in fact relevant
 - $P = \Pr(\text{relevant} \mid \text{retrieved}) = TP / (TP + FP)$
 - intuition: how much junk did we give to the user?
- Recall
 - fraction of **all** relevant documents that were found
 - $R = \Pr(\text{retrieved} \mid \text{relevant}) = TP / (TP + FN)$
 - intuition: how much of the good stuff did we miss?
- Complementary, always report together
 - example: retrieve 1 vs. retrieve all

Copyright © 2011 Victor Lavrenko

Why not accuracy?

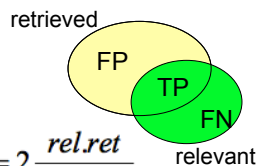


- Retrieval ... a kind of classification
 - document \rightarrow {relevant, non-relevant}
 - standard measure: $Accuracy = \frac{\text{correct}}{\text{total}} = \frac{TP + TN}{N}$
 - or use Error = 1 - Accuracy
- Meaningless:
 - accuracy 99.99% for any search algorithm
 - typical query: 10^2 - 10^3 relevant, same for retrieved
 - but N (total documents) $\sim 10^6$ - 10^7
 - accuracy "swamped" by negative events
 - doesn't matter what you retrieve

Copyright © 2011 Victor Lavrenko

F-measure

- A variant of accuracy not affected by negatives
 - single-value measure (compare, tune systems)
- Harmonic mean of P and R: $F_\beta = \frac{(\beta^2 + 1) \cdot P \cdot R}{\beta^2 P + R}$
 - β ... relative importance of recall and precision
 - popular setting: $\beta=1$, which gives: $F_1 = \frac{2PR}{P+R}$
 - heavily penalizes small values of P and R
- Geometric interpretation:



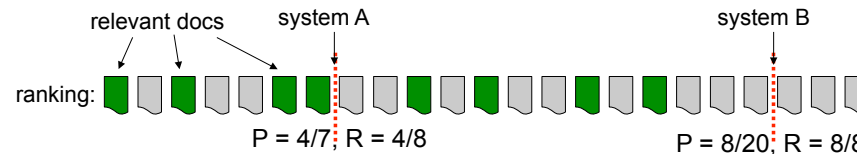
$$F_1 = \frac{2PR}{P+R} = 2 \left(\frac{1}{P} + \frac{1}{R} \right)^{-1} = 2 \left(\frac{TP+FP}{TP} + \frac{TP+FN}{TP} \right)^{-1} = 2 \frac{\text{rel.ret}}{\text{rel+ret}}$$

aka Dice coefficient

Copyright © 2011 Victor Lavrenko

Comparing recall / precision

- Which of the following is a better system?
 - system A: recall = 50%, precision = 57%
 - system B: recall = 100%, precision = 40%
- Could be the same exact system
 - using different threshold settings
 - R/P, F_1 comparisons often meaningless
 - more informative to compare ranking against ranking





Copyright © 2011 Victor Lavrenko

Recall / Precision and ranking

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

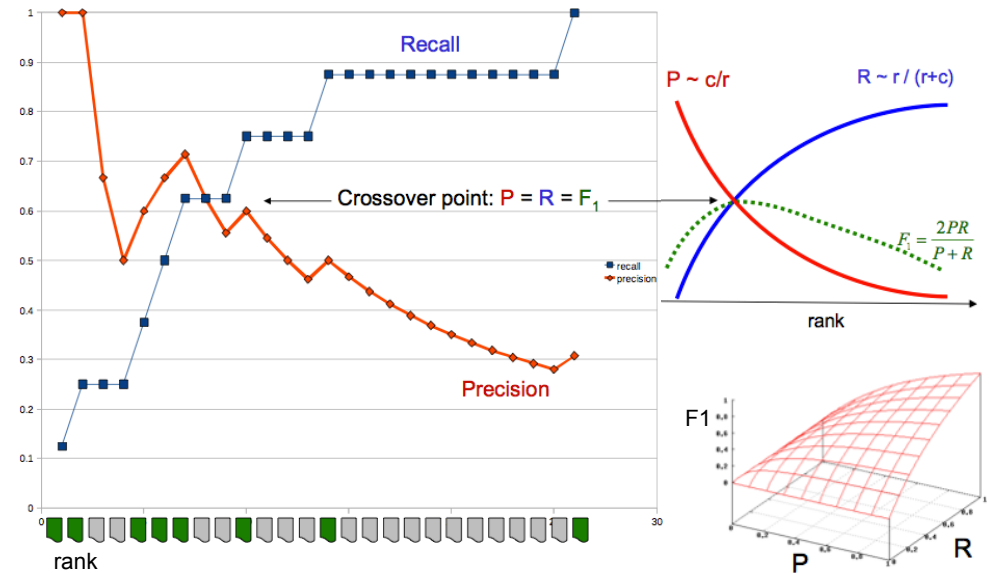
 = the relevant documents

Ranking #1	
Recall	0.17 0.17 0.33 0.5 0.67 0.83 0.83 0.83 0.83 1.0
Precision	1.0 0.5 0.67 0.75 0.8 0.83 0.71 0.63 0.56 0.6

Ranking #2	
Recall	0.0 0.17 0.17 0.17 0.33 0.5 0.67 0.67 0.83 1.0
Precision	0.0 0.5 0.33 0.25 0.4 0.5 0.57 0.5 0.56 0.6

Copyright © 2011 Victor Lavrenko

Recall / Precision / F1 vs rank



Copyright © 2011 Victor Lavrenko

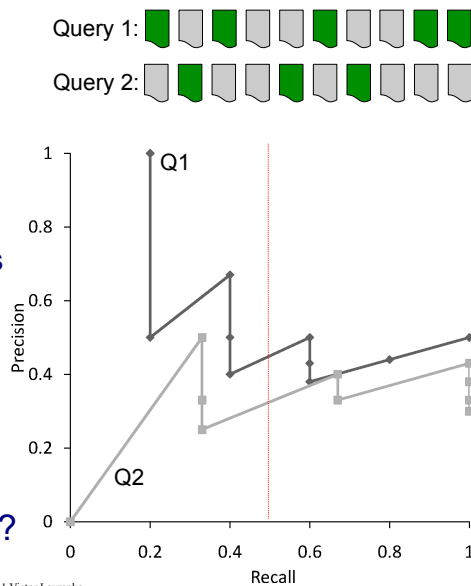
Recall-precision plot (raw)

- Plot precision vs. recall

- one curve per query
- detailed picture, but...
 - erratic behaviour
 - want to “average” curves

- Standard averaging

- at fixed recall levels
 - 0.1, 0.2, 0.3 ... 1.0
- what is precision at 0.5?
- need to interpolate, how?



Copyright © 2011 Victor Lavrenko

Recall-precision plot: interpolation

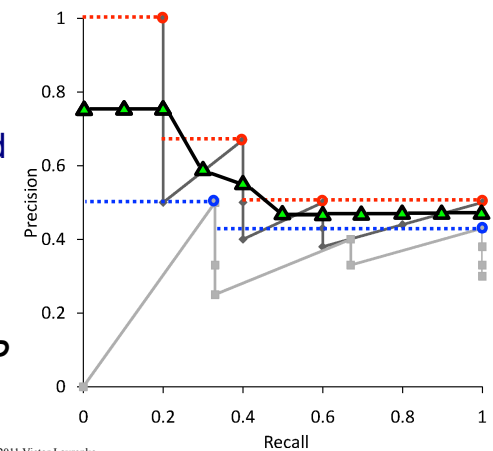
- Interpolation hard: $P(0)$, not a function
- On average precision drops as recall increases
- Define interpolation to preserve monotonicity

- max. precision observed at recall R or higher

$$\hat{P}(R) = \max_i \{P_i : R_i \geq R\}$$

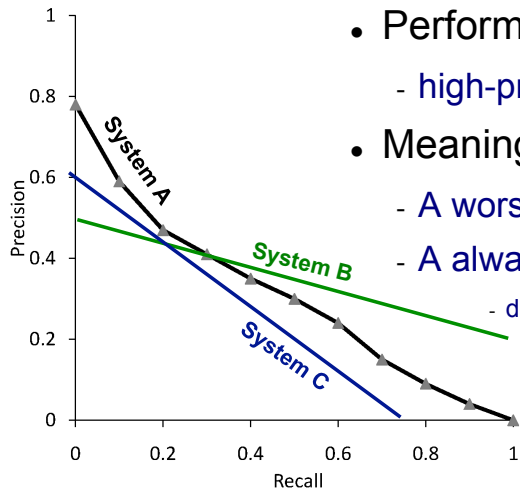
- $(P_i, R_i) \dots$ raw values

- Average interpolated P at standard levels of R



Copyright © 2011 Victor Lavrenko

Interpolated recall-precision plot



- Averaged over 50 queries
- Performance for all user types
 - high-precision and high-recall
- Meaningful system comparisons
 - A worse than B if recall important
 - A always better than C
 - dominates at all recall levels

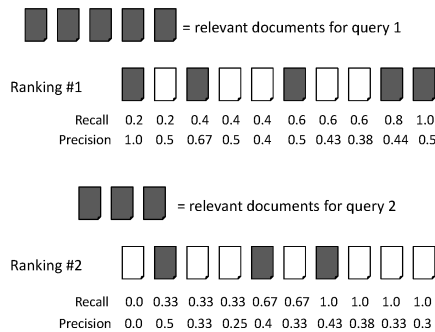
Copyright © 2011 Victor Lavrenko

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 towards top of the ranking (rank1 = 2 * rank2)
 - take the mean of Ave.P values across queries
 - GMAP: geometric average to combine Ave.P
 - heavily penalize if any query has low performance

Copyright © 2011 Victor Lavrenko

Mean Average Precision: example



$$\begin{aligned} \text{average precision query 1} &= (1.0 + 0.67 + 0.5 + 0.44 + 0.5) / 5 = 0.62 \\ \text{average precision query 2} &= (0.5 + 0.4 + 0.43) / 3 = 0.44 \\ \text{mean average precision} &= (0.62 + 0.44) / 2 = 0.53 \end{aligned}$$

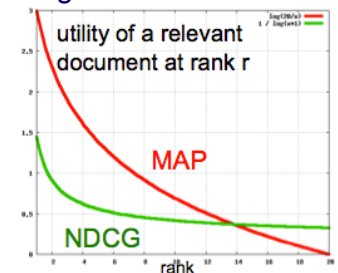
Copyright © 2011 Victor Lavrenko

Discounted Cumulative Gain

- Based on relative utility of relevant documents
 - most useful at top ranks
 - utility decreases with rank
 - allows graded degrees of relevance (rel_r)
- Normalized version (NDCG):
 - divide by DCG of ideal (best possible) ranking
- MAP has a similar effect:

$$DCG_k = \sum_{r=1}^k \frac{rel_r}{\log(r+1)}$$

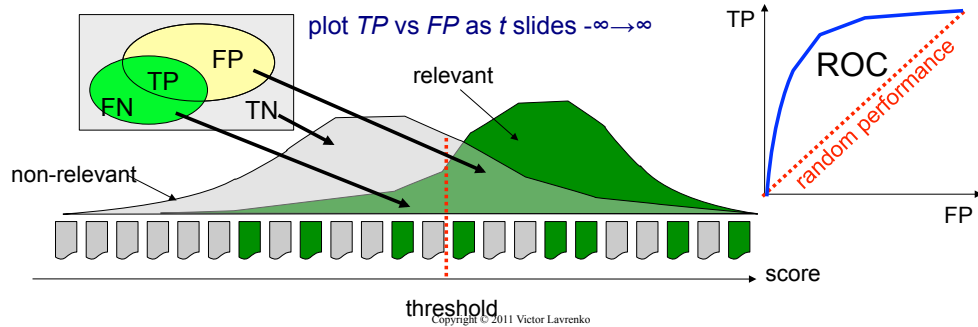
$$\begin{aligned} MAP_k &= \frac{1}{k} \sum_{i=1}^k P_i = \frac{1}{k} \sum_{i=1}^k \frac{1}{i} \sum_{r=1}^i rel_r = \frac{1}{k} \sum_{r=1}^k rel_r \sum_{i=r}^k \frac{1}{i} \\ &\approx \frac{1}{k} \sum_{r=1}^k rel_r \int_r^k \frac{1}{x} dx = \frac{1}{k} \sum_{r=1}^k rel_r \log \frac{k}{r} \end{aligned}$$



Copyright © 2011 Victor Lavrenko

Classification Errors

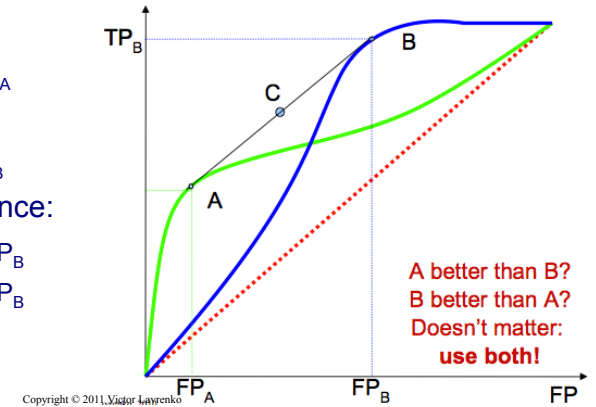
- False Pos. rate: $\Pr(x > t | -) = FP / (FP + TN)$
- False Neg. rate = $\Pr(x < t | +) = FN / (TP + FN)$
- True Pos. rate = $\Pr(x > t | +) = 1 - \text{False Neg.}$
- Receiver Operating Characteristic (ROC):



ROC convex hull

- System A: better at high thresholds (high-precision)
- System B: better at low thresholds (high-recall)
- System C: flip a p -biased coin

- heads: use system A
 - performance: TP_A, FP_A
- tails: run system B
 - performance TP_B, FP_B
- C expected performance:
 - $TP_C = p TP_A + (1-p) TP_B$
 - $FP_C = p FP_A + (1-p) FP_B$

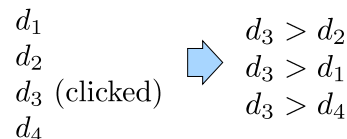


Query Logs

- Logs used to tune / evaluate search engines
 - session id, query, documents, click, timestamps

- Click \neq relevance judgement

- correlated, but noisy
- generate preferences



- Aggregate clicks to reduce noise

- click deviation: $CD(d, p) = O(d, p) - E(p)$
 - $O(d, p)$... click rate for document d in position p
 - $E(p)$... expected click rate for position p

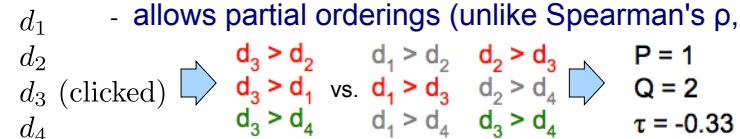
- average click deviation correlates with relevance

- consistently clicked despite low rank \rightarrow probably relevant
- consistently not clicked despite high rank \rightarrow probably not

Evaluation with preferences

- Kendall tau rank correlation coefficient: $\tau = \frac{P - Q}{P + Q}$

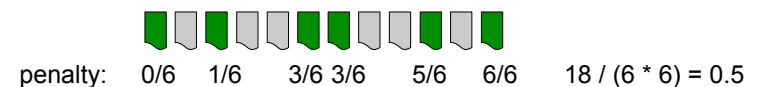
- P, Q ... number of concordant / discordant pairs
- allows partial relevance judgements (unlike Ave.P)
- allows partial orderings (unlike Spearman's ρ , BPREF)



- Binary preference

- R : relevant, N_{dr} : non-relevant above d_r
- allows partial judgements, assumes full ordering

$$BPREF = \frac{1}{R} \sum_{d_r} \left(1 - \frac{N_{dr}}{R}\right) = \frac{P}{P + Q}$$



Testing significance

Query:	Q1	Q2	Q3	Q4	Q5	Mean
System A	0.61	0.52	0.12	0.73	0.22	0.44
System B	0.32	0.55	0.13	0.32	0.12	0.29
difference	-0.29	+0.03	+0.01	-0.41	-0.10	-0.15

- Compare systems A,B on 5 queries
 - observe: A has 50% higher MAP than B
 - is this significant?
- Statistical hypothesis testing:
 - see if differences can be explained by pure chance
 - null hypothesis (H_0): A,B are equivalent
 - if $Pr(H_0) < 0.05$: reject H_0 , conclude A is better

Copyright © 2011 Victor Lavrenko

Testing significance: Sign test

Query:	Q1	Q2	Q3	Q4	Q5	Mean
System A	0.61	0.52	0.12	0.73	0.22	0.44
System B	0.32	0.55	0.13	0.32	0.12	0.29
difference	-0.29	+0.03	+0.01	-0.41	-0.10	-0.15
sign	(-)	(+)	(+)	(-)	(-)	3/5

- No assumptions about differences, just (+) or (-)
- $H_0: Pr(+)=Pr(-)=\frac{1}{2}$, five “coin tosses”
 - odds of observing 3 or more (-): $\sum_{n=3}^5 \frac{5!}{n!(5-n)!} \left(\frac{1}{2}\right)^5 = \frac{15}{32}$
 - 50% likelihood by pure chance → can't reject H_0
 - A and B are not significantly different
- Other tests: Wilcoxon signed rank test, T-test
 - more sensitive, make assumptions about data

Copyright © 2011 Victor Lavrenko

Training and testing

- Search systems require parameter tuning
 - run/evaluate many times on the same data
 - tempting to report the best result (bad idea)
 - training set: find the best parameter values
- cross-validate {
 - training: system can look at relevance judgments
 - validation: tune parameter values to maximize MAP/F1/...
- testing: run your system (once), report
- Splitting the data
 - by query: topics → {training,testing}, run over the same docs
 - different from what's traditional in ML, avoids vocabulary bias
 - by document: docs → {training,testing}, over the same topics
 - typically streaming applications: monitoring news, detecting events

Copyright © 2011 Victor Lavrenko

Summary: evaluation

- Evaluation is key: test early, test often
- Cranfield paradigm:
 - automated evaluation based on test collection
- Evaluation measures:
 - accuracy meaningless (99.9%)
 - set-based measures (R,P) depend on threshold!
 - use: recall-precision plots, MAP
- Query logs → preferences → Kendall tau / BPREF
- Always test significance (sign test)
- Never report best of N trials
 - run only once on testing data (or report every single run)

Copyright © 2011 Victor Lavrenko