IR Evaluation

CS4422/7263 Information Retrieval
Lecture 01

Jiho Noh

Department of Computer Science
Kennesaw State University

CS4422/7263 Spring 2025

Example – Result Table [1]

Models	Tmall				Diginetica				30music			
	P@10	MRR@10	P@20	MRR@20	P@10	MRR@10	P@20	MRR@20	P@10	MRR@10	P@20	MRR@20
FPMC	13.10	7.12	16.06	7.32	15.43	6.20	26.53	6.95	1.51	0.55	2.40	0.61
GRU4REC	9.47	5.78	10.93	5.89	17.93	7.33	29.45	8.33	15.91	10.46	18.28	10.95
NARM	19.17	10.42	23.30	10.70	35.44	15.13	49.70	16.17	37.81	25.95	39.40	26.55
STAMP	22.63	13.12	26.47	13.36	33.98	14.26	45.64	14.32	36.13	25.97	42.57	26.27
SR-GNN	23.41	13.45	27.57	13.72	36.86	15.52	50.73	17.59	36.49	26.71	39.93	26.94
GCE-GNN	29.19	15.55	34.35	15.91	41.54	**18.29**	54.64	**19.20**	39.93	21.21	44.71	21.55
S^2-DHCN	26.22	14.60	31.42	15.05	41.16	18.15	53.18	18.44	40.05	17.58	45.49	17.97
COTREC	30.62	17.65	36.35	18.04	**41.88**	18.16	54.18	19.07	39.88	17.42	45.15	17.79
MGS	**35.39***	**18.15***	**42.12***	**18.62***	41.80	18.20	**55.05***	19.13	**41.51***	**27.67***	**46.46***	**28.01***

Evaluation metrics: P@n, MRR

[1] table from SIGIR '22 "An Attribute-Driven Mirror Graph Network for Session-based Recommendation"

Example – Result Table 2

Datasets	Models	NDCG@N			MAP@N			Recall@N		
		$N = 1$	$N = 3$	$N = 5$	$N = 1$	$N = 3$	$N = 5$	$N = 1$	$N = 3$	$N = 5$
Coat	MF [17]	0.3748	0.3441	0.3714	0.1346	0.2100	0.2566	0.1346	0.2592	0.3705
	UAE [41]	0.3610	0.3546	0.3815	0.1265	0.2165	0.2648	0.1265	0.2785	0.3869
	IAE [41]	0.3655	0.3560	0.3812	0.1311	0.2185	0.2651	0.1311	0.2769	0.3847
	RelMF [39]	0.3959	0.3659	0.3922	0.1484	0.2281	0.2758	0.1484	0.2819	0.3926
	AT [37]	0.4017	0.3652	0.3912	0.1517	0.2286	0.2753	0.1517	0.2772	0.3908
	PD [61]	0.3997	0.3543	0.3737	0.1433	0.2182	0.2606	0.1433	0.2622	0.3627
	MACR [54]	0.4176	0.3798	0.3973	0.1559	0.2389	0.2834	0.1559	0.2875	0.3870
	CJMF† [63]	0.4093	0.3856	0.4097	0.1500	0.2408	0.2900	0.1500	0.2984	0.4075
	BISER (ours)	**0.4503***	**0.4109***	**0.4378****	**0.1725***	**0.2663****	**0.3192****	**0.1725***	**0.3185***	**0.4367****
	Gain (%)	10.03	6.56	6.85	14.99	10.58	10.08	14.99	6.74	7.16
Yahoo! R3	MF [17]	0.1797	0.2081	0.2411	0.1071	0.1688	0.1970	0.1071	0.2225	0.3040
	UAE [41]	0.1983	0.2235	0.2532	0.1198	0.1836	0.2104	0.1198	0.2362	0.3111
	IAE [41]	0.2137	0.2355	0.2653	0.1309	0.1956	0.2232	0.1309	0.2461	0.3211
	RelMF [39]	0.1837	0.2122	0.2453	0.1102	0.1728	0.2014	0.1102	0.2266	0.3080
	AT [37]	0.1912	0.2179	0.2506	0.1149	0.1786	0.2071	0.1149	0.2310	0.3125
	PD [61]	0.1994	0.2308	0.2647	0.1211	0.1901	0.2207	0.1211	0.2459	0.3297
	MACR [54]	0.2044	0.2274	0.2571	0.1243	0.1882	0.2154	0.1243	0.2382	0.3133
	CJMF† [63]	0.2151	0.2426	0.2715	0.1320	0.2018	0.2291	0.1320	0.2564	0.3297
	BISER (ours)	**0.2323****	**0.2608****	**0.2894****	**0.1446****	**0.2195****	**0.2479****	**0.1446****	**0.2748****	**0.3477****
	Gain (%)	7.99	7.52	6.60	9.58	8.77	8.20	9.58	7.18	5.47

Evaluation metrics: NDCG, MAP, Recall

[1] Table from SIGIR '22 "Bilateral Self-unbiased Learning from Biased Implicit Feedback"

Example – Result Table 3

Table 3: Results of stance detection.

Dataset	RumourEval2019-S							SemEval8						
Method	AUC	MicF	MacF	S F_1	D F_1	Q F_1	C F_1	AUC	MicF	MacF	S F_1	D F_1	Q F_1	C F_1
Zero-Shot	–	0.369	0.324	0.301	0.168	0.342	0.486	–	0.383	0.344	0.278	0.162	0.480	0.456
Pre-Rule	–	0.605	0.478	0.657	0.419	–	–	–	0.429	0.389	0.432	0.644	–	–
C-GCN	0.633	0.629	0.416	0.331	0.173	0.429	0.730	0.610	0.625	0.411	0.327	0.161	0.430	0.728
BrLSTM(V)	0.710	0.660	0.420	0.460	0.000	0.391	0.758	0.676	0.665	0.401	0.493	0.000	0.381	0.730
BiGRU(V)	0.700	0.630	0.417	0.392	0.162	0.360	0.754	0.660	0.633	0.416	0.460	0.168	0.328	0.708
MT-GRU(V)	0.714	0.636	0.432	0.313	0.156	0.506	0.748	0.669	0.630	0.413	**0.498**	0.116	0.312	0.729
TD-MIL(V)	0.712	0.650	0.432	0.438	0.156	0.408	0.688	0.668	0.626	0.416	0.473	0.127	**0.463**	0.602
BU-MIL(V)	0.710	0.630	0.431	**0.485**	0.166	0.396	0.688	0.669	0.623	0.415	0.470	0.128	0.460	0.602
TD-MIL(T15)	0.706	0.668	0.427	0.339	0.173	0.444	0.752	0.663	0.642	0.418	0.330	0.174	0.420	0.750
TD-MIL(T16)	0.713	0.665	**0.436**	0.350	**0.182**	0.446	0.758	0.660	**0.671**	0.421	0.334	0.173	0.422	0.754
TD-MIL(PHE)	**0.722**	**0.691**	0.434	0.344	0.179	**0.467**	**0.767**	**0.669**	0.651	**0.426**	0.335	**0.175**	0.430	**0.763**
BU-MIL(T15)	0.706	0.662	0.428	0.341	0.173	0.436	0.756	0.661	0.638	0.415	0.326	0.168	0.420	0.748
BU-MIL(T16)	0.701	0.660	0.426	0.340	0.170	0.438	0.749	0.659	0.637	0.416	0.324	0.169	0.419	0.753
BU-MIL(PHE)	0.707	0.665	0.432	0.344	0.174	0.445	0.762	0.666	0.642	0.420	0.329	0.169	0.423	0.758

Evaluation metrics: AUC, MicF, MacF, F1

[1] Table from SIGIR '22 "A Weakly Supervised Propagation Model for Rumor Verification..."

Example – Result Table 4

Training Domains	Query Domain	Methods	ZSL		GZSL	
			mAP@200	Prec@200	mAP@200	Prec@200
Real, Sketch Infograph, Painting Clipart	Sketch	EISNet [49]	0.3719	0.3136	0.3355	0.2822
		CuMix [31]	0.3689	0.3069	0.3300	0.2714
		SnMpNet [34]	0.4221	0.3496	0.3767	0.3109
		SASA (Ours)	**0.5487**	**0.4655**	**0.4865**	**0.4146**
Real, Quickdraw Infograph, Painting Clipart	Quickdraw	EISNet [49]	0.2475	0.1906	0.2118	0.1627
		CuMix [31]	0.2546	0.1967	0.2177	0.1699
		SnMpNet [34]	0.2888	0.2314	0.2366	0.1918
		SASA (Ours)	**0.3819**	**0.2993**	**0.3118**	**0.2488**

Evaluation metrics: mAP, Prec

[1] Table from SIGIR '22 "Structure-Aware Semantic-Aligned Network..."

IR Evaluation

- How do we compare these search engines? How do we evaluate the IR systems behind?
  - How are the results relevant to the user query?
  - How fast does it search?
  - How fast does it index (update contents)?
  - Is the result presented effectively?
- In this lecture, we will mainly focus on the retrieval effectiveness
  - How good an IR system retrieve relevant items to the user's information need.

IR Evaluation (Cont.)

- Note: **user need** is translated into a **query**
- Relevance is assessed relative to the user need, not the query
- E.g.,
  - Query: *"pool cleaner"*
  - Information need: [*My swimming pool bottom is becoming black and needs to be cleaned*]
- Assess whether the doc addresses the underlying need, not whether it has these words

Precision and Recall

- The relevance of a document to a query is **subjective**.
- We need precise definition of "relevance" and quantitative evaluation metrics.
- How can we evaluate the retrieval performance?
  - Manual vs. Automatic

rank	model A	model B
1	87641	1851596
2	57182758	8722556
3	6165392	13769
4	157692	2910525
5	878262	37619
6	4718	995166

Table: Two ranked list of documents, Which are better?

Confusion Matrix – Binary Assessment

- Confusion matrix is a table that shows the performance of a supervised learning model.
  - Each row represents the ground truth class assignments,
  - while each column represents the predicted classes, or vice versa.

	Retrieved	Non Retrieved
Relevant	True Positive (TP)	False Negative (FN)
Not Relevant	False Positive (FP)	True Negative (TN)

Precision

- Precision is the fraction of retrieved documents that were relevant.

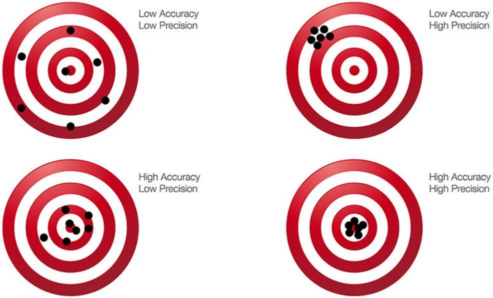
$$
\begin{aligned}
prec. &= \frac{|\text{retrieved and relevant}|}{|\text{retrieved}|} \\
&= \frac{TP}{TP + FP}
\end{aligned}
$$

Recall

- Recall is the fraction of relevant documents retrieved by the system.

$$\begin{aligned} prec. &= \frac{|\text{retrieved and relevant}|}{|\text{relevant}|} \\ &= \frac{TP}{TP + FN} \end{aligned}$$

Precision vs. Accuracy



- Precision is how close/dispersed the measurements are **to each other**.
- Accuracy is how close or far off a given set of measurements are **to their true value**. Accuracy in Binary Classification?

Precision and Recall

- Suppose the total number of relevant documents is 10
- Green's are TPs; predicted "relevant" and that is correct
- Red's are FPs; predicted "relevant" and that is incorrect
- What are the precision and recall of the models?

rank	model A	model B
1	87641	1851596
2	57182758	8722556
3	6165392	13769
4	157692	2910525
5	878262	37619
6	4718	995166

F-score

- **F-Measure** combines both recall and precision, so systems that favor are penalized for whichever is lower.

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall}$$

- Commonly used F-Measure is the F1 score, where $\beta = 1$

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

- What are the F1 scores of the model A and B?

1. What do we evaluate?

2. **Evaluation Metrics**
   - Binary assessment
   - Rank-based measures
   - Diagnostic Tools
   - Beyond Binary Relevance: NDCG

3. Test Collections and the Cranfield Paradigm

4. Online Evaluation

Rank-based Measures

- Precision/Recall/F-1 are set-based measures
- Now, we will discuss rank-based measures
- Binary relevance
  - precision@K (P@K)
  - Mean Average Precision (MAP)
  - Mean Reciprocal Rank (MRR)
- Multiple levels of relevance
  - Normalized Discounted Cumulative Gain (NDCG)

P@K: Precision at K

- **Precision at K** corresponds to the number of relevant results among the top K retrieved documents
  1. set a rank threshold k
  2. compute the percentage of relevant in top k
  3. ignore documents ranked lower than k
- Model A: $P@5 = 2/5 = .4$
- Model B: $P@5 = 2/5 = .4$

What are the shortcomings of *precisionK* measure?

rank	**model A**	**model B**
1	87641	1851596
2	57182758	8722556
3	6165392	13769
4	157692	2910525
5	878262	37619
6	4718	995166

AP: Average Precision

- It is desirable to also consider the order (i.e., the positions of the relevant documents) in which the returned documents are presented.
- Average precision computes the average value of **precision at K scores**.

$$AP(\vec{r}, n) = \frac{1}{R} \sum_{k:\vec{r}_k=1} P@K(\vec{r}, k)$$

Average Precision – Example

rank	model A	P@K	AP
1	87641		
2	57182758		
3	6165392		
4	157692		
5	878262		
6	4718		
7	957174		
8	5827561		

- Suppose $\vec{r} = (0, 1, 1, 0, 0, 0, 1, 0)$
- Consider the rank of each relevant documents: $k_{\vec{r}_k = 1} = (2, 3, 7)$

Average Precision – Example (Cont.)

rank	model A	P@K	AP
1	87641	0/1	0
2	57182758	1/2	(1/2) / 1
3	6165392	2/3	(1/2+2/3) / 2
4	157692	2/4	same as above
5	878262	2/5	same as above
6	4718	2/6	same as above
7	957174	3/7	(1/2+2/3+3/7) / 3
8	5827561	3/8	same as above

- Compute the $P@K$ for each k: $P@k = (1/2, 2/3, 3/7)$
- Average Precision at $k = 7$:

$$AP(\vec{r}, 7) = \frac{1}{3} \cdot \left(\frac{1}{2} + \frac{2}{3} + \frac{3}{7} \right) \approx 0.53$$

What would be the precision at K from a perfect system?

MAP: Mean Average Precision – Example

- Related metric is **Mean Average Precision (MAP)**, which is the mean of Average Precision across multiple queries/rankings.
- E.g.,
  - $AP(\vec{r}(q_1), 10) = (1.0 + 0.67 + 0.5 + 0.44 + 0.5)/5 = 0.62$
  - $AP(\vec{r}(q_2), 10) = (0.5 + 0.4 + 0.43)/3 = 0.44$

$$MAP((q_1, q_2), 10) = \frac{1}{2} \cdot (0.62 + 0.44) = 0.53$$

Mean Average Precision (Cont.)

- If a relevant document never gets retrieved, we assume the precision corresponding to that relevant doc to be zero.
- MAP is macro-averaging: each query counts equally.
- Now, perhaps most commonly used measure in research papers.
- Good for web search?
  - MAP assumes user is interested in finding many relevant documents for each query.
  - MAP requires many relevance judgments in text collection.

R-Precision

- **R-precision** is the precision@R where R is the number of relevant documents to the given query.
- R is used as the cutoff, which varies from query to query
  - Suppose there are 20 documents (i.e., $R = 20$) to “apple store” in a corpus.
  - If your ranking system retrieved 5 relevant documents in the top 20 list, then R-precision is $5/20 = 0.25$.
- *R-precision requires knowing all documents that are relevant to a query.*

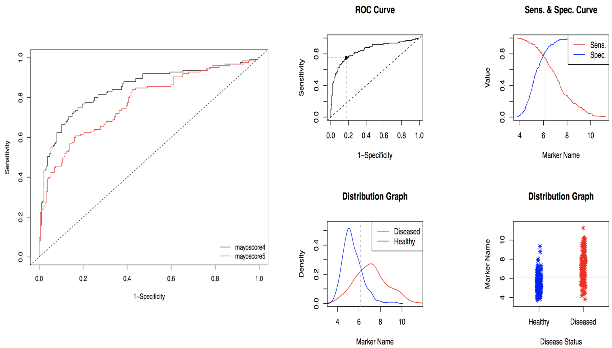
1. What do we evaluate?

2. **Evaluation Metrics**
   - Binary assessment
   - Rank-based measures
   - Diagnostic Tools
   - Beyond Binary Relevance: NDCG

3. Test Collections and the Cranfield Paradigm

4. Online Evaluation

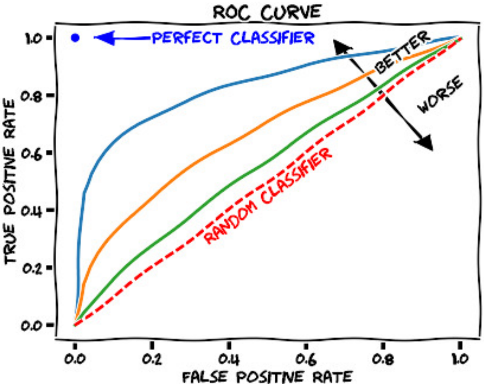
Diagnostic Tools



- In this lecture, we will discuss two diagnostic tools: ROC Curve and Precision-Recall Curve
- The area under the curve can be used to directly compare the models.

[1] image from: http://www.biosoft.hacettepe.edu.tr/easyROC/

ROC Curves



ROC (Receiver Operating Characteristic) curves illustrate the *relationships between correct predictions and wrong predictions on the* **positive** *class.*

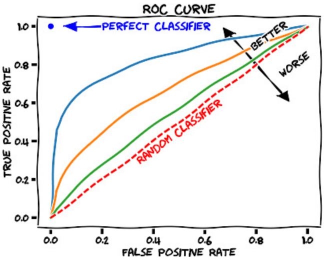
ROC Curves (Cont.)

- ROC Curves plot the **True Positive Rate** (TPR) against **False Positive Rate** (FPR).
- True Positive Rate (also called *recall*, *sensitivity*, *hit rate*).

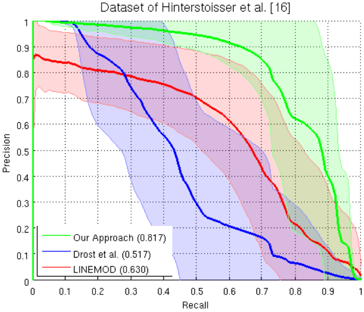
$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN}$$

- False Positive Rate (also called *fall-out*)

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN}$$

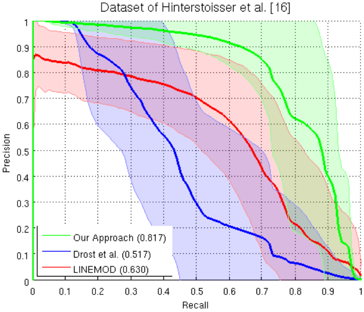


Precision-Recall Curves



- **Precision-Recall curves** illustrate the trade-off between **precision** and **recall** for every possible *discrimination threshold cut-off*.
- *What happen, if you classifier predicts everything positive? Or, vice versa?*

Precision-Recall Curves (Cont.)



- Precision-Recall is plotting the precision against recall
  - $precision = TP/(TP + FP)$
  - $recall = TP/(TP + FN)$
- Precision-Recall makes it possible to assess the performance of a model on the minority class

MRR: Mean Reciprocal Rank

- MRR considers the rank position (k) of the first relevant document – Could be the clicked website of the web search results.
- Reciprocal Rank:
$$RR = \frac{1}{k}$$
- MRR is the mean RR across multiple queries.

Normalized Discounted Cumulative Gain (NDCG)

Normalized Discounted Cumulative Gain

Assumptions:

- Graded relevance (beyond binary): For example, in 0-3 relevance scale, we have 0-irrelevant, 1-marginally relevant, 2-fairly relevant, 3-highly relevant.
- The relevant documents ranked lower are less useful for the user, since it is less likely to be examined.

Normalized Discounted Cumulative Gain (NDCG)

Normalized Discounted Cumulative Gain

- Graded relevance is used as a measure of usefulness, or **gain**
- Gain is **accumulated** starting at the top of the ranking and may be reduced, or discounted, at lower ranks
- Typical **discount** is $1/\log(rank)$
- DCG (Discounted Cumulative Gain) is the total gain accumulated at a particular rank p:

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}$$

DCG Example

- 10 ranked documents judged on 0–3 relevance scale:
  (3, 2, 3, 0, 0, 1, 2, 2, 3, 0)
- Discounted gain:
  (3, 2/1, 3/1.59, 0, 0, 1/2.59, 2/2.81, 2/3, 3/3.17, 0)
  = (3, 2, 1.89, 0, 0, 0.39, 0.71, 0.67, 0.95, 0)
- Discounted Cumulative Gain:
  (3, 5, 6.89, 6.89, 6.89, 7.28, 7.99, 8.66, 9.61, 9.61$(= DCG_{10})$)

NDCG for summarizing rankings

Normalized Discounted Cumulative Gain

- Normalize DCG at rank n
  - DCG at rank n by the DCG value at rank n of the ideal ranking
- The ideal ranking would first return the documents with the highest relevance level, then the next highest relevance level, etc
- Normalization is useful for contrasting queries with varying numbers of relevant results
- **NDCG** is now quite popular in evaluating Web search

NDCG Example

Suppose we have four documents: d_1, d_2, d_3, d_4 and the relevance judgment.

i	Ground Truth	Ranking Model A	Ranking Model B
1	$d_4(r_1 = 2)$	$d_3(r_2 = 2)$	$d_3(r_2 = 2)$
2	$d_3(r_2 = 2)$	$d_4(r_1 = 2)$	$d_2(r_3 = 1)$
3	$d_2(r_3 = 1)$	$d_2(r_3 = 1)$	$d_4(r_1 = 2)$
4	$d_1(r_4 = 0)$	$d_1(r_4 = 0)$	$d_1(r_4 = 0)$

NDCG Example (Cont.)

i	Ground Truth	Ranking Model A	Ranking Model B
1	$d_4(r_1 = 2)$	$d_3(r_2 = 2)$	$d_3(r_2 = 2)$
2	$d_3(r_2 = 2)$	$d_4(r_1 = 2)$	$d_2(r_3 = 1)$
3	$d_2(r_3 = 1)$	$d_2(r_3 = 1)$	$d_4(r_1 = 2)$
4	$d_1(r_4 = 0)$	$d_1(r_4 = 0)$	$d_1(r_4 = 0)$

- Ground Truth

$$DCG_{gt} = 2 + \left(\frac{2}{\log_2 2} + \frac{1}{\log_2 3} + \frac{0}{\log_2 4}\right) = 4.6309 \quad nDCG_{gt} = 4.6309/4.6309 = 1$$

- Model A

$$DCG_{A} = 2 + \left(\frac{2}{\log_2 2} + \frac{1}{\log_2 3} + \frac{0}{\log_2 4}\right) = 4.6309 \quad nDCG_{A} = 4.6309/4.6309 = 1$$

- Model B

$$DCG_{B} = 2 + \left(\frac{1}{\log_2 2} + \frac{2}{\log_2 3} + \frac{0}{\log_2 4}\right) = 4.2619 \quad nDCG_{B} = 4.2619/4.6309 = 0.9203$$

MRR: Mean Reciprocal Rank

- MRR considers rank position, k, of the first relevant document; it could be the clicked website in web search scenarios.
- Reciprocal Rank score: $RR = \frac{1}{k}$
- MRR is the mean RR across multiple queries.

TREC_EVAL Example Results

num_q	all	30
num_ret	all	30000
num_rel	all	3875
num_rel_ret	all	1555
map	all	0.1206
gm_ap	all	0.0614
R-prec	all	0.1906
bpref	all	0.2482
recip_rank	all	0.4924
ircl_prn.0.00	all	0.5676
ircl_prn.0.10	all	0.3467
ircl_prn.0.20	all	0.2443
ircl_prn.0.30	all	0.1773
ircl_prn.0.40	all	0.1206
ircl_prn.0.50	all	0.0678
ircl_prn.0.60	all	0.0345
ircl_prn.0.70	all	0.0194
ircl_prn.0.80	all	0.0051
ircl_prn.0.90	all	0.0000
ircl_prn.1.00	all	0.0000
P5	all	0.3933
P10	all	0.3833
P15	all	0.3600
P20	all	0.3300
P30	all	0.2933
P100	all	0.1777
P200	all	0.1297
P500	all	0.0807
P1000	all	0.0518

- **map**: mean average precision
- **gm_ap**: geometric mean average precision
- **R-prec**: R-precision
- **bpref**: preference-based IR measure
- **recip_rank**: reciprocal rank
- **ircl_prn**: interpolated recall – precision average at *r* recall
- **p@k**: precision at *k*

User-based vs. Laboratory Evaluation

User-based

- Manual by actual users
- Expensive
- Slower and most likely one-off
- Difficult to eliminate the sources of biases
- inconsistency between raters and over time
- Human judgments are not always representative of "real users"

Laboratory

- Automatic by an evaluation system
- Not expensive
- Faster and easier to replicate
- Difficult to build fair test collections

Test Collections

- To compare the performances of the IR systems
- Test collection is **a laboratory environment** that does not change in which we test and compare retrieval models
- It is wrong to report results on a test collection which were obtained by tuning the model parameters to maximize performance on the test collection.
- You should *tune* your model on one or more development test collections and obtain results from the test collection.

The Cranfield Paradigm

- Cyril Cleverdon (1914-1997) and his colleagues established the foundation for the evaluation of IR systems
- He started a series of projects, called **the Cranfield projects** whereby retrieval experiments were conducted on test databases in a controlled, laboratory-like setting.

Figure: A British librarian and computer scientist

The Cranfield Project

- He worked at the library of the College of Aeronautics at Cranfield, UK, in the late 1950's long before computers came around.
- The objective of his study was to find what kinds of **indexing languages** were most effective.
- **Findings:**
  - For indexing purpose, using the terms from the corpus is more effective as opposed to expert-selected topical or category terms (i.e., controlled vocabulary)
  - The inverse relationship of precision and recall

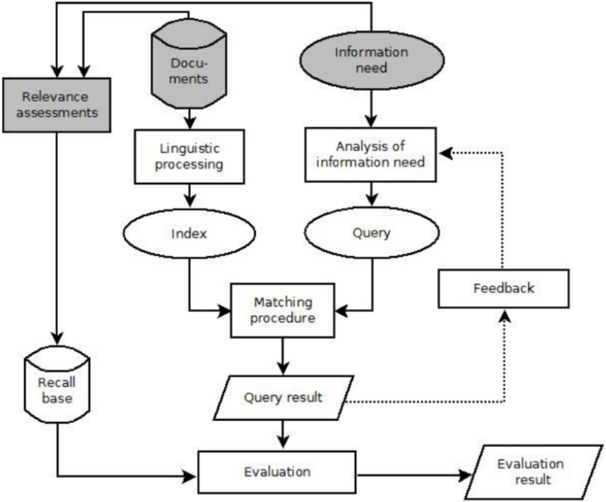
The Cranfield Paradigm – Today

- Model a real user application, with realistic information needs.
- Scale up the observational examples to allow significant testing.
- Make relevance judgments before the experiments.

Three elements of test collections

1. A benchmark document collection
2. A benchmark set of queries
3. An assessment of either **Relevant** or **Non-relevant** for each query and each document (Also called *relevance judgements*)

Schematic of Test Collection



TREC (Text REtrieval Conference)

- Run by the U.S. National Institute for Standards and Technology (NIST) in Maryland
- Has been a major driver of IR research and source of test collections from 1992 onward.
- Several language-specific conferences have been created internationally to follow its approach

TREC-style Conferences

name	focus	started	recent topics
TREC	English (mainly)	1992	conversational assistance, deep learning, health misinformation, news, podcast, precision medicine
NTCIR	East Asian languages	1999	financial tweets, government data, Wikipedia, ad hoc web search
CLEF	European languages, cross-language IR	2000	math QA, biomedical QA, multimedia retrieval, text translation (scientific to simple text)
FIRE	Indian and South Asian languages	2008	hate speech detection, AI for legal assistant, sentiment analysis, fake news detection, causality-driven ad hoc IR

TREC – the Precision Medicine (PM) Track

2021 TREC-PM Objective

TREC-PM focuses on **identifying high-quality evidence for a specific cancer treatment**. Each case will describe the patient's disease (type of cancer), the relevant genetic variants (which genes are mutated), and the proposed treatment. Participants of the track will be challenged with retrieving biomedical articles providing strong evidence for/against the treatment in the specific population.

Document Collections

- The MEDLINE baseline will be used for the scientific abstracts, which is a repository for biomedical and life science journal articles.

Document Collection Example

- NLM produces a baseline set of MEDLINE/PubMed citation records in XML format.


```xml
<?xml version="1.0"?>
 <!DOCTYPE article PUBLIC "-//NLM//DTD Journal Archiving and Interchange
   DTD v2.3 20070202//EN" "archivearticle.dtd">
 <article xmlns:xlink="http://www.w3.org/1999/xlink" article-type="review-article">
   <?properties open_access?>
   <front>
     <journal-meta>
       <journal-id journal-id-type="nlm-ta">Crit Care</journal-id>
       <journal-title>Critical Care</journal-title>
       <issn pub-type="ppub">1364-8535</issn>
       <issn pub-type="epub">1466-609X</issn>
       <publisher>
         <publisher-name>BioMed Central</publisher-name>
         <publisher-loc>London</publisher-loc>
       </publisher>
     </journal-meta>
     ...
```

Topics

The topics for the track consist of synthetic patient cases consist of the disease, genetic variants, and the proposed treatment.

	Patient 1	**Patient 2**
Disease:	melanoma	melanoma
Variant:	BRAF (V600E)	BRAF (V600E)
Treatment:	Dabrafenib	Cobimetinib

Topics in XML format: topic2020.xml

Relevance Judgments

- Binary (*relevant* vs. *non-relevant*) in the simplest case
  - More nuanced relevance levels also used (e.g., 0-3 scale)
- Any Issues?
  - Human judges often disagree
  - We expect to pay human experts (doctors, medical coders, etc.)

```
1 0 1065003 1
1 0 1065013 1
1 0 1065027 1
1 0 1065055 0
1 0 1065094 2
1 0 107740 0
1 0 1079876 0
1 0 1160567 1
1 0 1160569 1
1 0 117132 0
1 0 1175863 1
1 0 1175871 0
1 0 1175911 0
1 0 1180426 0
```

Pooling technique and Why?

- In most collections, it is not feasible to assess the relevance of each document for each query;
  - For example, 10,000 predicted documents for 50 topics from 100 participating systems. Assuming that judging each document takes 5 mins, then it will take up to 475 years to evaluate!!
- Hence, pooling technique is used on a large scale IR evaluation tasks.

Typical Pooling Sequence

1. The documents and queries are created.
2. Multiple IR systems are run on the queries.
3. Each system returns top-ranking m documents, which are collected into a pool.
4. The resulting pool of documents is assessed in random order, typically by multiple judges.
5. When judges disagree, they meet and discuss the document until they reach consensus.

Issues of Static Test Collections

- IR systems are becoming increasingly **contextual and personal**;
- Users and their information needs are too **diverse and dynamic** to be adequately captured in static test collections;
- No longer can users or experts be asked to provide **objective assessments**.

Online evaluation for IR addresses the challenges that require assessment of systems in terms of their utility for the user.

Online Evaluation

- **Online evaluation** is the evaluation of a fully functioning system based on implicit measurement of real users' experiences of the system in a natural usage environment.
- Implicit measurement, include any measurements that can be derived from observable user activity that is part of users' natural or normal interaction with the system.
- Distinction between implicit and explicit measurements is that **implicit measurements** are a by-product of users' natural interaction, while **explicit ones** are specifically collected for feedback purposes.

Offline vs. Online Evaluation

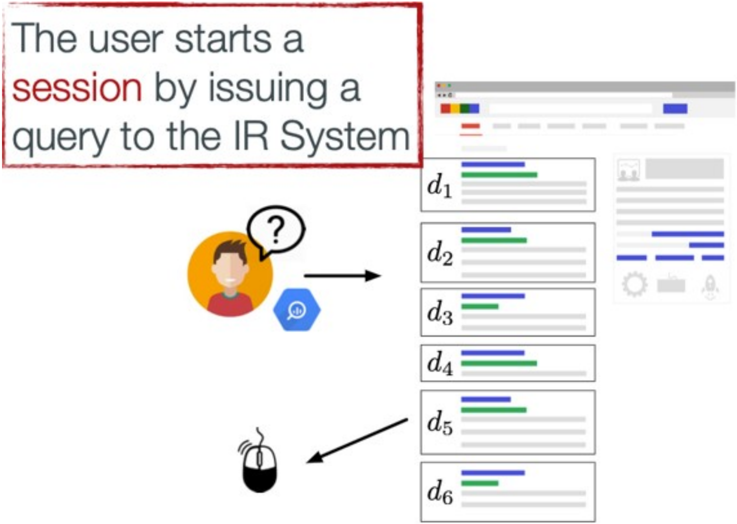
- **Offline Evaluation**:
  - Direct comparison of one engine to another
  - Simple and easy to replicate and reproduce
- **Online Evaluation**:
  - Accounts for the actual user behavior during interactive sessions
  - Complex and costly and it is not reproducible
- **Offline Estimation**:
  - The estimation of online evaluation based on past observations of users' behavior

Click Log Data

Query logs have proven to be a valuable and informative source of implicit user feedback:

- they can be easily collected by search engines;
- they are available in real time;
- they represent personalized user preferences.

Terminology

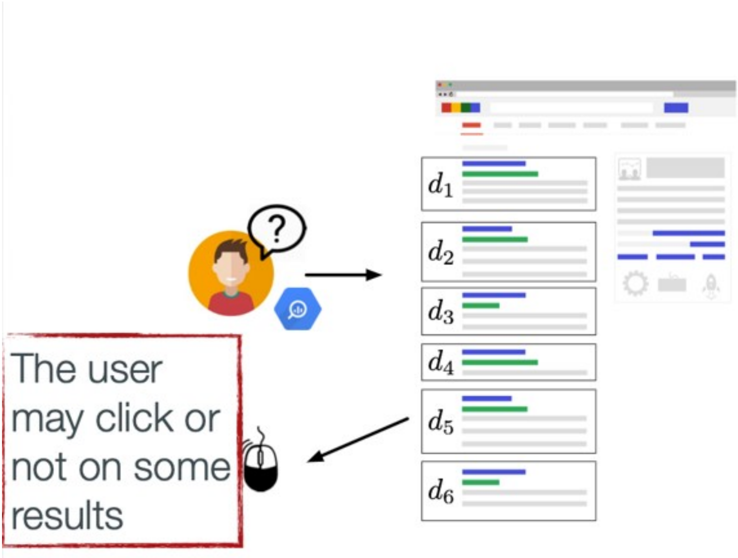


Terminology (Cont.)

- **Session**: series of interactions by the user toward addressing a single information need;
- Session, task, and goal are often used as synonymous, depending on whether a study was based on logs (where extracting session is easier than identifying tasks) or in a laboratory (where study participants complete one task or search goal at a time).

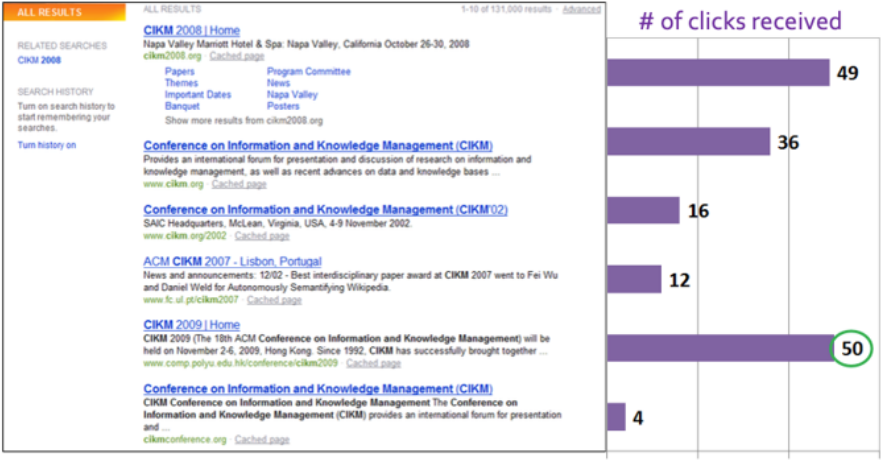
Terminology (Cont.)

Terminology (Cont.)



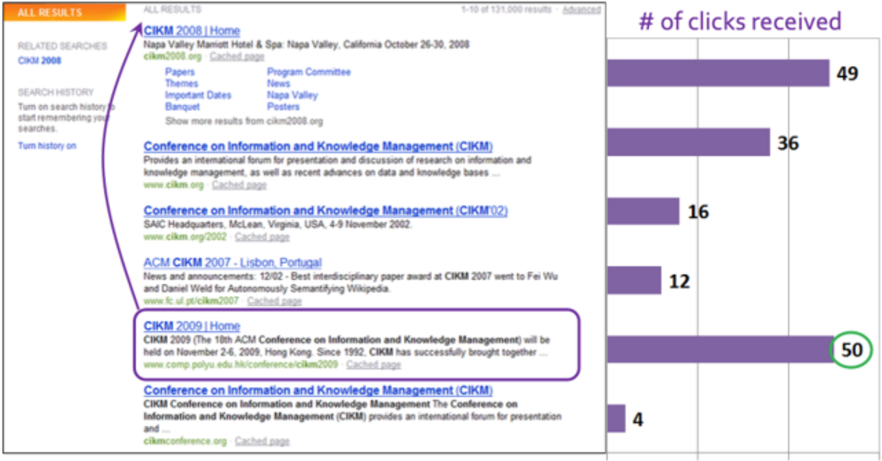
User Behavior

Search results for “CIKM” (in 2009!)



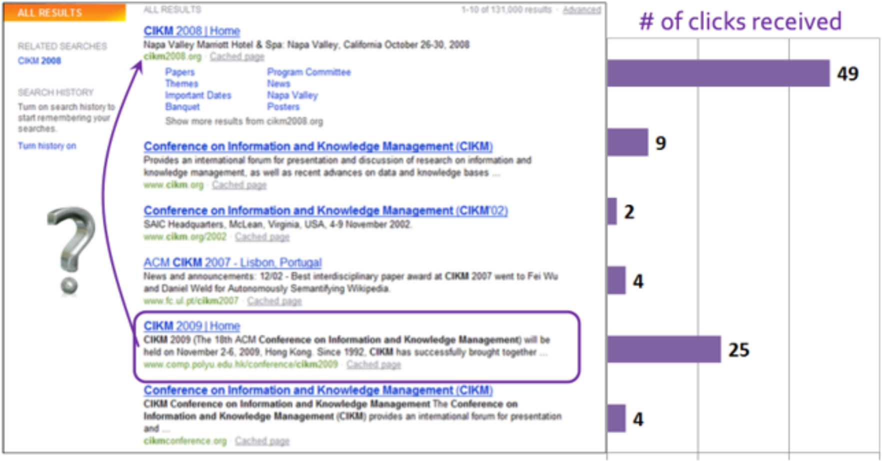
User Behavior (Cont.)

Adapt ranking to user clicks?

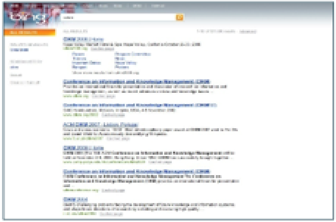


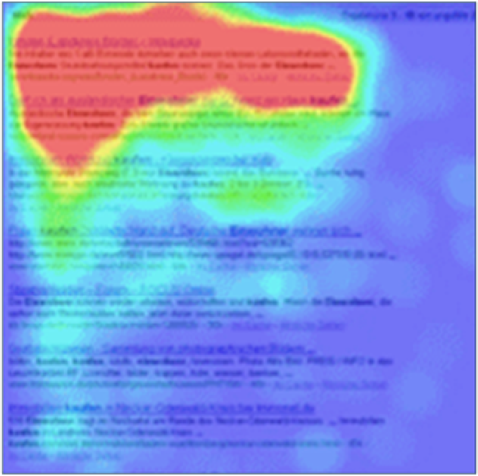
User Behavior (Cont.)

Tools needed for non-trivial cases



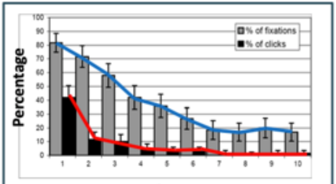
Eye-Tracking User Study



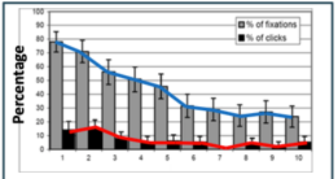


Strong position bias exists, so absolute click rates are unreliable.

Click Position-Bias



Normal Position

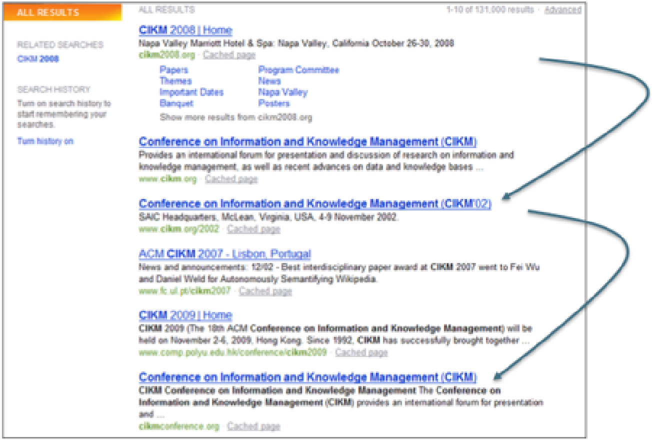


Reversed Impression

- Higher positions receive more user attention (eye fixation) and clicks than lower positions
- This is true even in the extreme setting where the order of positions is reversed
- "Clicks are informative but biased"

[1] [Joachims+07]

Relative vs. Absolute Ratings



User's click sequence

- Hard to conclude Result1 > Result3
- Probably can conclude Result3 > Result2

Comparing two rankings via clicks

Query: [support vector machines]

Ranking A	Ranking B
Kernel machines	Kernel machines
SVM-light	SVMs
Lucent SVM demo	Intro to SVMs
Royal Holl. SVM	Archives of SVM
SVM software	SVM-light
SVM tutorial	SVM software

Interleaving two rankings

This interleaving starts with the ranking B.

Kernel machines
Kernel machines
SVMs
SVM-light
Intro to SVMs
Lucent SVM demo
Archives of SVM
Royal Holl. SVM
SVM-light

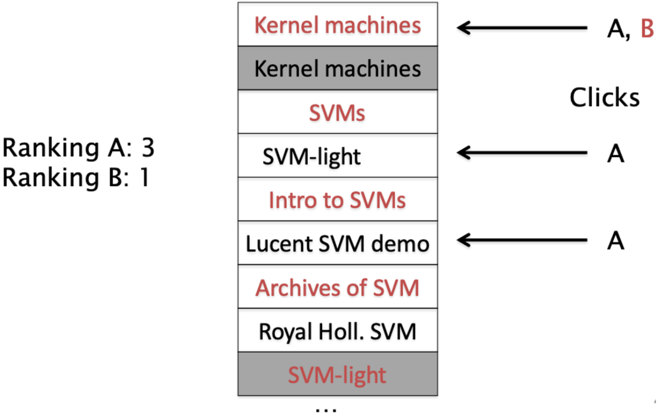
...

Remove duplicate results

Kernel machines
Kernel machines
SVMs
SVM-light
Intro to SVMs
Lucent SVM demo
Archives of SVM
Royal Holl. SVM
SVM-light

...

Count User Clicks



Interleaved Rankings

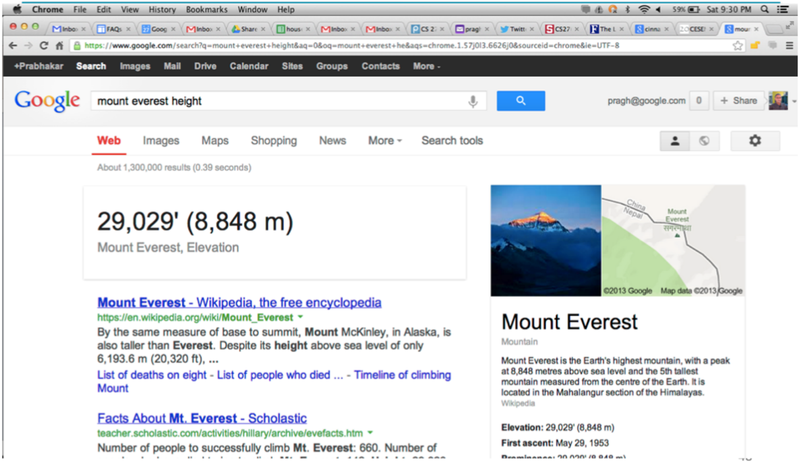
- Present interleaved ranking to users
- Start randomly with ranking A or ranking B to even out presentation bias
- Count clicks on results from A versus results from B
- Better ranking will (on average) get more clicks

A/B testing at web search engines

- Purpose: Test a single innovation
- Prerequisite: You have a large search engine up and running.
- Have most users use old system
- Divert a small proportion of traffic (e.g., 0.1%) to an experiment to evaluate an innovation
  - Interleaved experiment
  - Full page experiment

Question-answering / Knowledge Panel

What happens to clicks?



User Behavior

- User behavior is an intriguing source of relevance data
  - Users make (somewhat) informed choices when they interact with search engines
  - Potentially a lot of data available in search logs
- But there are significant caveats
  - User behavior data can be very noisy
  - Interpreting user behavior can be tricky
  - Spam can be a significant problem
  - Not all queries will have user behavior

Incorporating user behavior into ranking algorithms

- Incorporate user behavior features into a ranking function
  - BM25F, a BM25 extension version for weighting terms presented in multiple fields (e.g., user behavior features)
- Incorporate user behavior features into learned ranking function
- Either of these ways of incorporating user behavior signals improve ranking

Summary

- and discussion