CS317
Information Retrieval
Week 07

Muhammad Rafi
March 22, 2021

Agenda

- Evaluation in IR
- Ad Hoc Information Retrieval
- Standard IR Collections
- Evaluation for Unranked Retrieval
  - Precision
  - Recall
  - F-Score or F- measure
  - Fall-out
- Evaluation for Ranked Retrieval

Agenda

- Evaluation for Ranked Retrieval
  - Precision –Recall Curve
  - Average Precision
  - Mean Average Precision (MAP)
  - Cumulative Gain
  - Discount Cumulative Gain
  - Normalized Discount Cumulative Gain
- Conclusion

Different IR Models

- There are many retrieval models/ algorithms/ systems, which one is the best?
- What is the best component for:
  - Ranking function (dot-product, cosine, …)
  - Term selection (stopword removal, stemming…)
  - Term weighting (TF, TF-IDF,…)
- How far down the ranked list will a user need to look to find some/all relevant documents?

Difficulty in IR Evaluation

- Effectiveness is related to the ***relevancy*** of retrieved items.
- Relevancy is not typically binary but continuous.
- Even if relevancy is binary, it can be a difficult judgment to make.
- Relevancy, from a human standpoint, is:
  - Subjective: Depends upon a specific user's judgment.
  - Situational: Relates to user's current needs.
  - Cognitive: Depends on human perception and behavior.
  - Dynamic: Changes over time.

Ad hoc Information Retrieval

- To measure ad hoc information retrieval effectiveness in the standard way,
- we need a test collection consisting of three things:
  - A document collection
  - A test suite of information needs, expressible as queries
  - A set of relevance judgments, standardly a binary assessment of either relevant or non-relevant for each query-document pair.

Evaluation In IR

- Evaluation measures for an information retrieval system are used to assess how well the search results satisfied the user's query intent.
- It is used to compare two IR Systems.
- Evaluation Process is also an active area of research in IR
- Evaluation process started with a small dataset with only 100’s doc and 30 queries now it has grown to 1/15 of web scale.

Standard IR Collections

- The Cranfield collection.
  - Collected in the United Kingdom starting in the late 1950s, it contains 1398 abstracts of aerodynamics journal articles, a set of 225 queries, and exhaustive relevance judgments of all (query, document) pairs.
- 20 Newsgroups
  - It consists of 1000 articles from each of 20 Usenet newsgroups (the newsgroup name being regarded as the category).

Standard IR Collections

- Cross Language Evaluation Forum (CLEF)
  - This evaluation series has concentrated on EU languages and cross-language information retrieval.
- Reuters-21578 and Reuters-RCV1
  - For text classification, the most used test collection has been the Reuters-21578 collection of 21578 newswire articles
- WebKB
  - This data set contains WWW-pages collected from computer science departments of various universities in January 1997

Standard IR Collections

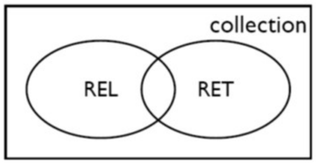
- Modern IR Collections
  - TREC
  - SemEval
  - LSHTC
  - CLEF

TREC Conference Tracks

Evaluation of unranked retrieval sets

- The result-set of a query is unranked (flat results only retrieved one, which systems proposed relevant)
- The result set is a “set” assuming there is no redundant document.
- From the collection, for a given query. We can have set of relevant documents. The system may returned a set of documents called retrieved. A possible subset of relevant document may be retrieved by the system.

Evaluation of unranked retrieval sets



$$\mathcal{P} = \frac{|RET \cap REL|}{|RET|} \qquad \mathcal{R} = \frac{|RET \cap REL|}{|REL|}$$

- Evaluation
  - Precision (P): the proportion of retrieved documents that are relevant
  - Recall (R): the proportion of relevant documents that are retrieved

Evaluation of unranked retrieval sets

- Precision
  - Measure of how much of the information the system returned is correct (accuracy).
  - Precision measures the system's ability to reject any non-relevant documents from the retrieved set
- Recall
  - Measure of how much relevant information the system has extracted (coverage of system).
  - Recall measures the system's ability to find all the relevant documents.

IR Evaluation

		relevant	
		Rel	NotRel
retrieved	Ret	Ret_{Rel}	Ret_{NotRel}
	NotRet	$NotRet_{Rel}$	$NotRet_{NotRel}$

$Ret = Ret_{Rel} + Ret_{NotRel}$

$NotRet = NotRet_{Rel} + NotRet_{NotRel}$

$Relevant = Ret_{Rel} + NotRet_{Rel}$ $\quad$ $Not\ Relevant = Ret_{NotRel} + NotRet_{NotRel}$

Total # of documents available $N = Ret_{Rel} + NotRet_{Rel} + Ret_{NotRel} + NotRet_{NotRel}$

- Precision: $P = Ret_{Rel}$ / Retrieved
- Recall: $R = Ret_{Rel}$ / Relevant

$P = [0,1]$
$R = [0,1]$

Example

	Retrieved	Not retrieved
Relevant	w=3	x=2
Not relevant	y=3	z=2

Relevant = w+x= 5

Not Relevant = y+z = 5

Retrieved = w+y = 6 $\quad$ Not Retrieved = x+z = 4

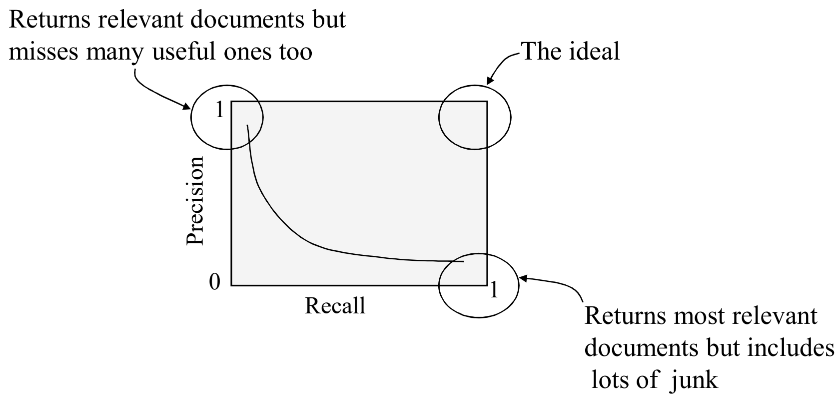
Total documents N = w+x+y+z = 10

- Precision: P= w / w+y =3/6 =.5
- Recall: R = w / w+x = 3/5 =.6

Precision Vs. Recall

- A system can make two types of errors:
  - a false positive error: the system retrieves a document that is non-relevant (should not have been retrieved)
  - a false negative error: the system fails to retrieve a document that is relevant (should have been retrieved)
- How do these types of errors affect precision and recall?
  - Precision <-> false positive errors
  - Recall <-> false negative errors

Precision Vs. Recall



Precision and Recall are inverse proportional

Precision Vs. Recall

Precision Critical Tasks	Recall Critical Tasks
Time matters a lot	Time matter less
Tolerance to missed documents	Non tolerance to missed documents
Redundant – many equal information resources	Less redundant information – only one (few resources)
Example: Web search	Example: legal/patent search
Demand: Very high	Demand: moderate
General optimizations	Specific Optimizations

F- Measure

- Precision and Recall stand in opposition to one another. As precision goes up, recall usually goes down (and vice versa).
- The F-measure combines the two values.
- F-Measure { $((ß^2+1)*P*R) / (ß^2 *P+R)$}
  - When ß = 1, precision and recall are weighted equally. Commonly Called $F_{(ß = 1)}$.
  - When ß is < 1, precision is favored.
  - When ß is > 1, recall is favored.

Fallout Rate

- Problems with both precision and recall:
  - Number of irrelevant documents in the collection is not taken into account.
  - Recall is undefined when there is no relevant document in the collection.
  - Precision is undefined when no document is retrieved.

$$Fallout = \frac{no.\ of\ nonrelevant\ items\ retrieved}{total\ no.\ of\ nonrelevant\ items\ in\ the\ collection}$$

Evaluation of ranked retrieval results

- Precision, recall, and the F measure are set-based measures. They are computed using unordered sets of documents.
- In a ranked retrieval context, appropriate sets of retrieved documents are naturally given by the top k retrieved documents.
- The system can return any number of results.
- How to compute this ranked result?
  - A precision-recall curve

Ranked Retrieval Example

n	doc #	relevant
1	588	x
2	589	x
3	576	
4	590	x
5	986	
6	592	x
7	984	
8	988	
9	578	
10	985	
11	103	
12	591	
13	772	x
14	990	

Let total # of relevant docs = 6
Check each new recall point:

R=1/6=0.167; P=1/1=1

R=2/6=0.333; P=2/2=1

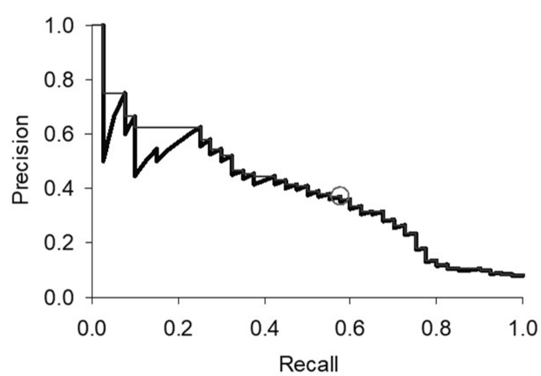
R=3/6=0.5; P=3/4=0.75

R=4/6=0.667; P=4/6=0.667

R=5/6=0.833; P=5/13=0.38

Missing one relevant document. Never reach 100% recall

A precision-recall curve



Average Precision

= 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

$$\text{Ranking \#1: } (1.0 + 0.67 + 0.75 + 0.8 + 0.83 + 0.6)/6 = 0.78$$

$$\text{Ranking \#2: } (0.5 + 0.4 + 0.5 + 0.57 + 0.56 + 0.6)/6 = 0.52$$

MAP

= relevant documents for query 1

Ranking #1

Recall	0.2	0.2	0.4	0.4	0.4	0.6	0.6	0.6	0.8	1.0
Precision	1.0	0.5	0.67	0.5	0.4	0.5	0.43	0.38	0.44	0.5

= relevant documents for query 2

Ranking #2

Recall	0.0	0.33	0.33	0.33	0.67	0.67	1.0	1.0	1.0	1.0
Precision	0.0	0.5	0.33	0.25	0.4	0.33	0.43	0.38	0.33	0.3

$$\textit{average precision query 1} = (1.0 + 0.67 + 0.5 + 0.44 + 0.5)/5 = 0.62$$
$$\textit{average precision query 2} = (0.5 + 0.4 + 0.43)/3 = 0.44$$

$$\textit{mean average precision} = (0.62 + 0.44)/2 = 0.53$$

Mean Average Precision (MAP)

- 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

Kappa Statistics

		Judge 2 Relevance		
		Yes	No	Total
Judge 1 Relevance	Yes	300	20	320
	No	10	70	80
	Total	310	90	400

Observed proportion of the times the judges agreed
$P(A) = (300 + 70)/400 = 370/400 = 0.925$
Pooled marginals
$P(nonrelevant) = (80 + 90)/(400 + 400) = 170/800 = 0.2125$
$P(relevant) = (320 + 310)/(400 + 400) = 630/800 = 0.7878$
Probability that the two judges agreed by chance
$P(E) = P(nonrelevant)^2 + P(relevant)^2 = 0.2125^2 + 0.7878^2 = 0.665$
Kappa statistic
$\kappa = (P(A) - P(E))/(1 - P(E)) = (0.925 - 0.665)/(1 - 0.665) = 0.776$

► **Table 8.2** Calculating the kappa statistic.

Cumulative Gain (CG)

- An old technique called Cumulative Gain(CG)
- It is the sum of the graded relevance values of all results in a search result list.
- Let for a query "q" there are following six documents D1,D2,D3,D4,D5 and D6. The relative relevance of these documents are 3,2,3,0,1,2
- The value of CG = sum of all relevance for all six documents.
- Changing the order of any two documents does not affect the CG measure.

Discount Cumulative Gain (DCG)

- *DCG* is the total gain accumulated at a particular rank *p*:

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

- Alternative formulation:

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i}-1}{log(1+i)}$$

  - used by some web search companies
  - emphasis on retrieving highly relevant documents

Solved Example

$D_1, D_2, D_3, D_4, D_5, D_6$

the user provides the following relevance scores:

$3, 2, 3, 0, 1, 2$

That is: document 1 has a relevance of 3, document 2 has a relevance of 2, etc. The Cumulative Gain of this search result listing is:

$$\mathrm{CG}_6 = \sum_{i=1}^{6} rel_i = 3 + 2 + 3 + 0 + 1 + 2 = 11$$

Solved Example

i	rel_i	$\log_2(i+1)$	$\frac{rel_i}{\log_2(i+1)}$
1	3	1	3
2	2	1.585	1.262
3	3	2	1.5
4	0	2.322	0
5	1	2.585	0.387
6	2	2.807	0.712

$$\mathrm{DCG}_6 = \sum_{i=1}^{6} \frac{rel_i}{\log_2(i+1)} = 3 + 1.262 + 1.5 + 0 + 0.387 + 0.712 = 6.861$$

Normalized DCG

- Normalized Discounted Cumulative Gain (NDCG) at rank *n*
  - Normalize 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 useful for contrasting queries with varying numbers of relevant results
- NDCG is now quite popular in evaluating Web search

Solved Example

i	rel_i	$\log_2(i+1)$	$\frac{rel_i}{\log_2(i+1)}$
1	3	1	3
2	2	1.585	1.262
3	3	2	1.5
4	0	2.322	0
5	1	2.585	0.387
6	2	2.807	0.712

$$\mathrm{DCG}_6 = \sum_{i=1}^{6} \frac{rel_i}{\log_2(i+1)} = 3 + 1.262 + 1.5 + 0 + 0.387 + 0.712 = 6.861$$

Ideal Order of the result : $3, 3, 2, 2, 1, 0$

$$\mathbf{IDCG_6 = 7.141}$$

And so the nDCG for this query is given as:

$$\mathbf{nDCG_6} = \frac{DCG_6}{IDCG_6} = \frac{6.861}{7.141} = 0.961$$

Normalized DCG (Example)

4 documents: d_1, d_2, d_3, d_4

i	Ground Truth		Ranking Function$_1$		Ranking Function$_2$	
	Document Order	r_i	Document Order	r_i	Document Order	r_i
1	d4	2	d3	2	d3	2
2	d3	2	d4	2	d2	1
3	d2	1	d2	1	d4	2
4	d1	0	d1	0	d1	0
	$NDCG_{GT}$=1.00		$NDCG_{RF1}$=1.00		$NDCG_{RF2}$=0.9203	

$$DCG_{GT} = 2 + \left(\frac{2}{\log_2 2} + \frac{1}{\log_2 3} + \frac{0}{\log_2 4} \right) = 4.6309$$

$$DCG_{RF1} = 2 + \left(\frac{2}{\log_2 2} + \frac{1}{\log_2 3} + \frac{0}{\log_2 4} \right) = 4.6309$$

$$DCG_{RF2} = 2 + \left(\frac{1}{\log_2 2} + \frac{2}{\log_2 3} + \frac{0}{\log_2 4} \right) = 4.2619$$

$$MaxDCG = DCG_{GT} = 4.6309$$

System Quality

- There are many practical benchmarks on which to rate an information retrieval system beyond its retrieval quality.
- System Quality is also a concern.
  - How fast does it index, that is, how many documents per hour does it index for a certain distribution over document lengths?
  - How fast does it search, that is, what is its latency as a function of index size?
  - How expressive is its query language? How fast is it on complex queries?
  - How large is its document collection, in terms of the number of documents or the collection having information distributed across a broad range of topics?

User utility

- What we would really like is a way of quantifying aggregate user happiness, based on the relevance, speed, and user interface of a system.
- One indirect measure of such users is that they tend to return to the same engine.

Evaluation of System Changes

- A/B testing
  - For such a test, precisely one thing is changed between the current system and a proposed system, and a small proportion of traffic (say, 1–10% of users) is randomly directed to the variant system, while most users use the current system.
  - Click through log analysis or clickstream mining. To see whether User like it or not.
  - The basis of A/B testing is running a bunch of single variable tests (either in sequence or in parallel): for each test only one parameter is varied from the control (the current live system).

Search Snippets

- Search Snippets is useful for reviewing the search results.
- The two basic kinds of summaries:
  - Static: which are always the same regardless of the query,
  - Dynamic: (or query-dependent), which are customized according to the user's information need as deduced from a query. Dynamic summaries attempt to explain why a particular document was retrieved for the query at hand.
    - keyword-in-context (KWIC) snippets

Conclusion

- Get as much of what we want while at the same time getting as little junk as possible.
- Recall is the percentage of relevant documents returned compared to everything that is available!
- Precision is the percentage of relevant documents compared to what is returned!
- The desired trade-off between precision and recall is specific to the scenario we are in?
- What do we want?
  - Find everything relevant – high recall
  - Only retrieve what is relevant – high precision