CS657A: Information Retrieval
Evaluation Measures

Arnab Bhattacharya
arnabb@cse.iitk.ac.in

Computer Science and Engineering,
Indian Institute of Technology, Kanpur
http://web.cse.iitk.ac.in/~cs657/

2nd semester, 2021-22

Tue 1030-1145, Thu 1200-1315

Errors

- Positives (P): Documents that are "true" answers, i.e., "relevant"
- Negatives (N): Documents that are not relevant: $N = D - P$
- For a particular retrieval algorithm $\mathcal{A}$,
  - P': Documents returned as relevant by $\mathcal{A}$
  - N': Documents not returned as relevant by $\mathcal{A}$
- True Positives (TP): Relevant documents returned by $\mathcal{A}$: $P \cap P'$
- True Negatives (TN): Irrelevant documents not returned by $\mathcal{A}$: $N \cap N'$
- False Positives (FP): Irrelevant documents returned by $\mathcal{A}$: $N \cap P'$
- False Negatives (FN): Relevant documents not returned by $\mathcal{A}$: $P \cap N'$

$$P = TP \cup FN \quad N = TN \cup FP \quad P' = TP \cup FP \quad N' = TN \cup FN$$

- In statistics,
  - Type I error: FP
  - Type II error: FN

Confusion Matrix

- Confusion matrix visually represents the information
- Rows indicate "true" relevance: P and N
- Columns indicate those returned by $\mathcal{A}$: P' and N'
- Shows which error is more

Sets		Returned by $\mathcal{A}$	
		Positives P'	Negatives N'
True answers	Positives P	TP	FN
	Negatives N	FP	TN

- Is more useful when extended for multiple classes (not just relevant versus irrelevant)
- Shows which classes are confused more against which other classes

Error Parameters or Performance Metrics

Parameter	*Interpretation*	*Formula*
Precision	Proportion of positives in those returned by $\mathcal{A}$	$\frac{\lvert TP \rvert}{\lvert TP \cup FP \rvert} = \frac{\lvert TP \rvert}{\lvert P' \rvert}$
Recall or Sensitivity or True positive rate	Proportion of positives returned by $\mathcal{A}$	$\frac{\lvert TP \rvert}{\lvert TP \cup FN \rvert} = \frac{\lvert TP \rvert}{\lvert P \rvert}$
Specificity or True negative rate	Proportion of negatives not returned by $\mathcal{A}$	$\frac{\lvert TN \rvert}{\lvert TN \cup FP \rvert} = \frac{\lvert TN \rvert}{\lvert N \rvert}$
False positive rate	Proportion of negatives returned by $\mathcal{A}$	$\frac{\lvert FP \rvert}{\lvert TN \cup FP \rvert} = \frac{\lvert FP \rvert}{\lvert N \rvert}$
False negative rate	Proportion of positives not returned by $\mathcal{A}$	$\frac{\lvert FN \rvert}{\lvert TP \cup FN \rvert} = \frac{\lvert FN \rvert}{\lvert P \rvert}$
Accuracy	Proportion of positives returned and negatives not returned by $\mathcal{A}$	$\frac{\lvert TP \cup TN \rvert}{\lvert D \rvert}$
Error rate	Proportion of positives not returned and negatives returned by $\mathcal{A}$	$\frac{\lvert FP \cup FN \rvert}{\lvert D \rvert}$

Single Measures

- Single measures capturing both precision and recall
- F-score or F-measure or F1-score is the *harmonic mean* of precision and recall

$$Fscore = \frac{2.Precision.Recall}{Precision + Recall}$$

- In terms of errors

$$Fscore = \frac{2.TP}{2.TP + FN + FP}$$

- Similarly, G-measure is *geometric mean* of precision and recall
- EER or Equal Error Rate is when FP rate is equal to FN rate

Weighting Precision versus Recall

- Suppose recall and precision are weighted at a ratio $\alpha : (1 - \alpha)$
- F-score is the *weighted harmonic mean*

$$\frac{1}{F} = \alpha . \frac{1}{P} + (1 - \alpha) . \frac{1}{R}$$

- $\beta^2 = \frac{1-\alpha}{\alpha}$ measures the relative importance of precision over recall
  - $\alpha \in [0, 1]$ while $\beta \in [0, \infty]$
  - $\beta > 1$ emphasizes precision, while $\beta < 1$ emphasizes recall
- Using β^2, weighted F-score is

$$F = \frac{(\beta^2 + 1) . P . R}{\beta^2 . P + R}$$

- When $\beta = 1$, precision and recall are equally weighted ($\alpha = 1/2$)
- F1-score is the harmonic mean

Example

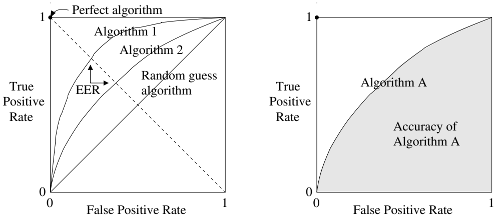
$$
\begin{aligned}
D =& \{d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8\} \\
\text{Correct answer set} =& \{d_1, d_5, d_7\} \\
\text{Algorithm } \mathcal{A} \text{ returns} =& \{d_1, d_3, d_5, d_6\} \\
\therefore \quad P =& \{d_1, d_5, d_7\} \\
N =& \{d_2, d_3, d_4, d_6, d_8\} \\
TP =& \{d_1, d_5\} \\
TN =& \{d_2, d_4, d_8\} \\
FP =& \{d_3, d_6\} \\
FN =& \{d_7\} \\
\therefore \quad \text{Recall} =& 2/3 = 0.667 \\
\text{Precision} =& 2/4 = 0.500 \\
\text{F-score} =& 4/7 = 0.571 \\
\text{Accuracy} =& 5/8 = 0.625
\end{aligned}
$$

ROC Curve

- Performance of an algorithm depends on parameters
- To assess over a range of parameters, ROC curve is used
    - 1 - Specificity (x-axis) versus Sensitivity (y-axis)
    - False positive rate (x-axis) versus True positive rate (y-axis)

ROC Curve

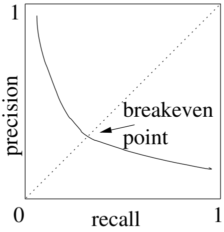
- Performance of an algorithm depends on parameters
- To assess over a range of parameters, ROC curve is used
  - 1 - Specificity (x-axis) versus Sensitivity (y-axis)
  - False positive rate (x-axis) versus True positive rate (y-axis)
- A random guess algorithm is a 45° line



- Area under the ROC curve (AUC or AUROC) measures accuracy (or discrimination)
- What AUC is good?
  - 0.9+: excellent; 0.8+: good; 0.7+: fair; 0.6+: poor; 0.6-: fail
- EER denotes the point in ROC where FP rate is equal to FN rate

Precision-Recall Curve

- Precision versus recall



- Breakeven point where precision is the same as recall

Interpolated Precision

- Interpolated precision at recall level r is *maximum* precision achieved at any recall level $r' \geq r$

$$\text{inter-prec}(r) = \max_{\forall r' \geq r} p(r)$$

- Eleven-point interpolated precision: Interpolated precisions at particular standard recall values
  - 0.0, 0.1, ..., 1.0
  - Uses interpolated precision

Mean Reciprocal Rank

- For some applications, only the *first* answer matters
  - "I'm feeling lucky"
- Reciprocal rank measures the rank of the first relevant document in the retrieved list

$$RR = \frac{1}{rank}$$

- Falls quickly
- Mean over multiple queries produces mean reciprocal rank (MRR)

$$MRR = \frac{1}{|Q|} . \sum_{\forall i=1}^{Q} RR(q_i)$$

Set-Based Measures

- Suppose, the correct relevant set of documents is C
- A set of documents R is retrieved
- Jaccard similarity or Jaccard coefficient is

$$JS(R, C) = \frac{|R \cap C|}{|R \cup C|}$$

- Dice coefficient is

$$DC(R, C) = \frac{2.|R \cap C|}{|R| + |C|}$$

Order of Answers

- Suppose there are 5 relevant documents
- Every method is allowed to retrieve 10 documents
- Method 1 retrieves them at ranks 1, 3, 6, 9, 10
- Method 2 retrieves them at ranks 2, 5, 6, 7, ...
- Method 3 retrieves them at ranks 2, 3, 4, 5, 6
- Which one is better?

Average Precision

- Average precision (AP)
- Suppose there are R relevant documents
- A method retrieves n documents as answer

$$AP@n = \frac{1}{|R|} \sum_{i=1}^{n} \Big(Precision(i).Relevance(i) \Big)$$

- $Precision(i)$ is precision for first i answers
- $Relevance(i)$ is relevance of i^{th} answer
  - 1 if i^{th} answer is relevant
  - 0 otherwise, i.e., when answer is not relevant
- AP is average of precision whenever recall changes
- Between 1 (for an ideal answer) and 0 (for a completely wrong answer)

Example

- Method 1 retrieves them at ranks 1, 3, 6, 9, 10
- Method 2 retrieves them at ranks 2, 5, 6, 7, ...
- Method 3 retrieves them at ranks 2, 3, 4, 5, 6
- Ideal method retrieves them at ranks 1, 2, 3, 4, 5

Method	Position										AP @10
	1	2	3	4	5	6	7	8	9	10	
Method 1	+	-	+	-	-	+	-	-	+	+	
(precision)	1/1	0	2/3	0	0	3/6	0	0	4/9	5/10	0.62
Method 2	-	+	-	-	+	+	+	-	-	-	
(precision)	0	1/2	0	0	2/5	3/6	4/7	0	0	0	0.39
Method 3	-	+	+	+	+	+	-	-	-	-	
(precision)	0	1/2	2/3	3/4	4/5	5/6	0	0	0	0	0.71

- Average precision prefers *correct* answers at *higher* ranks

Mean Average Precision (MAP)

- Mean average precision is mean of average precision for Q queries

$$MAP@k = \frac{1}{|Q|} . \sum_{\forall i=1}^{Q} AP@k(q_i)$$

- MAP is found to be more robust than other measures
- It is, therefore, used widely

R-Precision

- Which k to use for precision@k?
- R-precision: k is set to the number of relevant documents R
- Then, precision is the same as recall
- This is the same as *breakeven point*

Discounted Cumulative Gain

- Not all relevant documents have the *same* relevance
- Suppose d_i have a relevance score rel_i
  - $r_i = 0$ if d_i is irrelevant
- Quality should take into account the relevance
  - Highly relevant documents should rank higher
- At rank p of a retrieved list, discounted cumulative gain accumulates gain discounted by rank

$$DCG(p) = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i}$$

- Alternatively, $\sum_{i=1}^{p} \log_2 (1+i)$ and/or $2^{rel_i} - 1$

Example

$rank_i$	rel_i	*factor*	*gain*	*DCG*
1	4	1.00	4.00	4.00
2	3	1.00	3.00	7.00
3	4	0.63	2.52	9.52
4	2	0.50	1.00	10.52
5	0	0.43	0.00	10.52
6	0	0.39	0.00	10.52
7	0	0.36	0.00	10.52
8	1	0.33	0.33	10.86
9	1	0.32	0.32	11.17
10	0	0.30	0.00	11.17

Normalized Discounted Cumulative Gain

- *DCG* keeps increasing
- Normalization to make it between 0 and 1
- *Ideal ranking* from the set of results available
- If ranking is (4, 3, 4, 2, 0, 0, 0, 1, 1, 0), perfect ranking is (4, 4, 3, 2, 1, 1, 0, 0, 0, 0)
- Ideal discounted cumulative gain (IDCG) is DCG of ideal ranking
- Normalized discounted cumulative gain (NDCG) is ratio of DCG to IDCG at each rank

$$NDCG = \frac{DCG}{IDCG}$$

NDCG

$rank_i$	DCG	$IDCG$	$NDCG$
1	4.00	4.00	1.00
2	7.00	8.00	0.88
3	9.52	9.89	0.96
4	10.52	10.89	0.97
5	10.52	11.32	0.93
6	10.52	11.71	0.90
7	10.52	11.71	0.90
8	10.86	11.71	0.93
9	11.17	11.71	0.95
10	11.17	11.71	0.95

Ranking of Answers

- Which are the 5 closest state capital cities from Kanpur?
  - Lucknow, Jaipur, Bhopal, Patna, Dehradun
- Using relevance
  - Lucknow: 5
  - Jaipur: 4
  - Bhopal: 3
  - Patna: 2
  - Dehradun: 1
  - Any other answer: 0
- Which answer is better?
  - Method 1: L-P-B-J-D
  - Method 2: L-J-D-P-B
  - Method 3: L-J-X-B-P

Example

$rank_i$	$factor$	Method 1		Method 2		Method 3		Ideal Method	
		rel_i	DCG	rel_i	DCG	rel_i	DCG	rel_i	$IDCG$
1	1.00	5	5.00	5	5.00	5	5.00	5	5.00
2	1.00	2	7.00	4	9.00	4	9.00	4	9.00
3	0.63	3	8.89	1	9.63	0	9.00	3	10.89
4	0.50	4	10.89	2	10.63	3	10.50	2	11.89
5	0.43	1	11.32	3	11.92	2	11.36	1	12.32

Kendall's Tau Coefficient

- Two ranked lists can also be compared using rank correlation measures
- Suppose ranked lists are $r_1, \ldots, r_n$ and $s_1, \ldots, s_n$
- There are $n_0 = \binom{n}{2} = n(n-1)/2$ pairs of relative rankings
- $a_{ij} = sign(r_i - r_j)$ and $b_{ij} = sign(s_i - s_j)$
- A ranking pair ij is concordant, c, if they agree, i.e., $a_{ij} = b_{ij}$
- A ranking pair ij is discordant, d, if they disagree, i.e., $a_{ij} \neq b_{ij}$
- Kendall's tau coefficient measures the difference of the two

$$\tau = \frac{(n_c - n_d)}{n_0}$$

- If there are n_1 and n_2 pairs at same ranking

$$\tau = \frac{(n_c - n_d)}{\sqrt{n_0 - n_1}\sqrt{n_0 - n_2}}$$

BPREF

- Binary preference (BPREF) measures how quickly relevant documents are retrieved before irrelevant ones
- For R relevant documents

$$BPREF = \frac{1}{|R|} \sum_{\forall d_r \in R} \left(1 - \frac{n_{d_r}}{|R|} \right)$$

- d_r is a relevant document
- n_{d_r} is the number of irrelevant documents retrieved before d_r
- Documents retrieved but not judged are ignored
- Example: IRII (with $|R| = 2$)
  - $BPREF = 1/2[(1 - 1/2)] = 1/4$
- Example: IRNNRI (with $|R| = 2$)
  - $BPREF = 1/2[(1 - 1/2) + (1 - 1/2)] = 1/2$
- Example: IRNNRIRI (with $|R| = 3$)
  - $BPREF = 1/3[(1 - 1/3) + (1 - 1/3) + (1 - 2/3)] = 5/9$

Pooling

- How is the ground truth created?
- Domain experts are handed only a few documents
- These documents are retrieved using a variety of IR methods
- If none of the IR methods has retrieved a document, it is possibly not relevant at all
- This is called pooling
- Do experts agree?
- Rarely, since every human has her own idiosyncrasies and ideas

Kappa Statistic

- For categorical judgments: "relevant" or "irrelevant"
- Suppose proportion of two judges agreeing is $P(A)$
- Kappa statistic or Cohen's kappa is *agreement rate*

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)}$$

- $P(E)$ is proportion of agreement by chance
- If number of relevant and irrelevant documents is the same, then $P(E) = 0.5$
- Kappa statistic can be negative
- It is 1 when two judges always agree
- It is 0 when two judges always agree only randomly
- > 0.8: excellent; > 0.6: good; > 0.4: fair; > 0.2: slight
- Below that, either the task is too confusing, or the dataset is too poor

Estimating Random Agreements

- $P(E)$ can be estimated in a more robust manner using actual data

	#R(B)	#I(B)	Total
#R(A)	20	12	32
#I(A)	4	4	8
Total	24	16	40

- $P(agreement) = (20 + 4)/40 = 0.60$
- Expected probability of both experts saying "R" is $P(R, R) = 32/40 \times 24/40 = 0.48$
- Expected probability of both experts saying "I" is $P(I, I) = 8/40 \times 16/40 = 0.08$
- Expected agreement is $P(E) = 0.48 + 0.08 = 0.56$
- $\kappa = (0.60 - 0.56)/(1 - 0.56) = 0.09$

Estimating using Pooling

- Chance agreements are better estimated using pooling

	#R(B)	#I(B)	Total
#R(A)	20	12	32
#I(A)	4	4	8
Total	24	16	40

- $P(\mathit{agreement}) = (20 + 4)/40 = 0.60$
- Expected probability of a document being categorized as "R" is $P(R) = (32 + 24)/(40 + 40) = 0.70$
- Expected probability of a document being categorized as "I" is $P(I) = (8 + 16)/(40 + 40) = 0.30$
- Expected probability of *agreement* is $P(E) = P(R)^2 + P(I)^2$ $= 0.70^2 + 0.30^2 = 0.58$
- $\kappa = (0.60 - 0.58)/(1 - 0.58) = 0.05$

Fleiss' Kappa

- What if there are multiple experts and/or multiple categories?
- Fleiss' kappa
- Suppose n documents are judged over k categories by m experts
  - Cohen's kappa: $k = 2,\ m = 2$
- There can be more than m experts and not all of them need to judge all documents
- x_{ij} is the number of experts assigning category j to document i
  - $\forall i,\ \sum_{\forall j} x_{ij} = m$
  - $\sum_{\forall i} \sum_{\forall j} x_{ij} = m.n$
- Proportion of documents assigned to category j is

$$p_j = \frac{1}{m.n} \sum_{\forall i} x_{ij}$$

- Therefore, probability of agreement by chance is

$$P(E) = \sum_{\forall j} p_j^2$$

Proportion of Agreement

- For a document i, for a category j, pairs of experts agreeing is $\binom{x_{ij}}{2}$
- Total number of possible pairs of experts is $\binom{m}{2}$
- Hence, over all categories, proportion of agreement for document i is

$$p_a(i) = \frac{\sum_{\forall j} \binom{x_{ij}}{2}}{\binom{m}{2}} = \frac{(\sum_{\forall j} x_{ij}^2) - m}{m(m-1)}$$

- Average proportion of agreement over all documents is

$$P(A) = \frac{1}{n}.\sum_{\forall i} p_a(i) = \frac{(\sum_{\forall i} \sum_{\forall j} x_{ij}^2) - m.n}{n.m(m-1)}$$

- Fleiss' kappa is

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)}$$

Example

x_{ij}	1	2	3	4	5	$p_a(i)$
1	0	0	0	0	14	1.00
2	0	2	6	4	2	0.25
3	0	0	3	5	6	0.31
4	0	3	9	2	0	0.44
5	2	2	8	1	1	0.33
6	7	7	0	0	0	0.46
7	3	2	6	3	0	0.24
8	2	5	3	2	2	0.18
9	6	5	2	1	0	0.29
10	0	2	2	3	7	0.29
Total	20	28	39	21	32	140
p_j	0.14	0.20	0.28	0.15	0.23	

- Contingency table
- $m = 14$ experts for $n = 10$ documents over $k = 5$ categories
- Proportion of category 1 is $p_1 = 20/140 = 0.14$
- Agreement for document 2 is
  $p_a(2) = (0^2 + 2^2 + 6^2 + 4^2 + 2^2 - 14)/(14 \times 13) = 0.25$

Example (contd.)

x_{ij}	1	2	3	4	5	$p_a(i)$
1	0	0	0	0	14	1.00
2	0	2	6	4	2	0.25
3	0	0	3	5	6	0.31
4	0	3	9	2	0	0.44
5	2	2	8	1	1	0.33
6	7	7	0	0	0	0.46
7	3	2	6	3	0	0.24
8	2	5	3	2	2	0.18
9	6	5	2	1	0	0.29
10	0	2	2	3	7	0.29
Total	20	28	39	21	32	140
p_j	0.14	0.20	0.28	0.15	0.23	

- Overall agreement is $P(A) = (1.00 + 0.25 + \cdots)/10 = 0.38$
- Expected agreement is $P(E) = 0.14^2 + 0.20^2 + \cdots = 0.21$
- Therefore, $\kappa = (0.38 - 0.21)/(1 - 0.21) = 0.22$