normal p/r
range based (p/r@k), r-p
average p/r
mean based
rank correlation based metrics

Jul – Nov 2022
IT458

Evaluating IR Systems

Mean based Metrics

Mean based metrics

- Gives average precision averaged across a set of queries
  - one of the most common metrics in IR evaluation
- Popular metrics
  - Mean Average Precision
  - Geometric Mean Average Precision
  - Mean Reciprocal Rank
  - Harmonic Mean Average Precision
    - F-score

Mean Average Precision

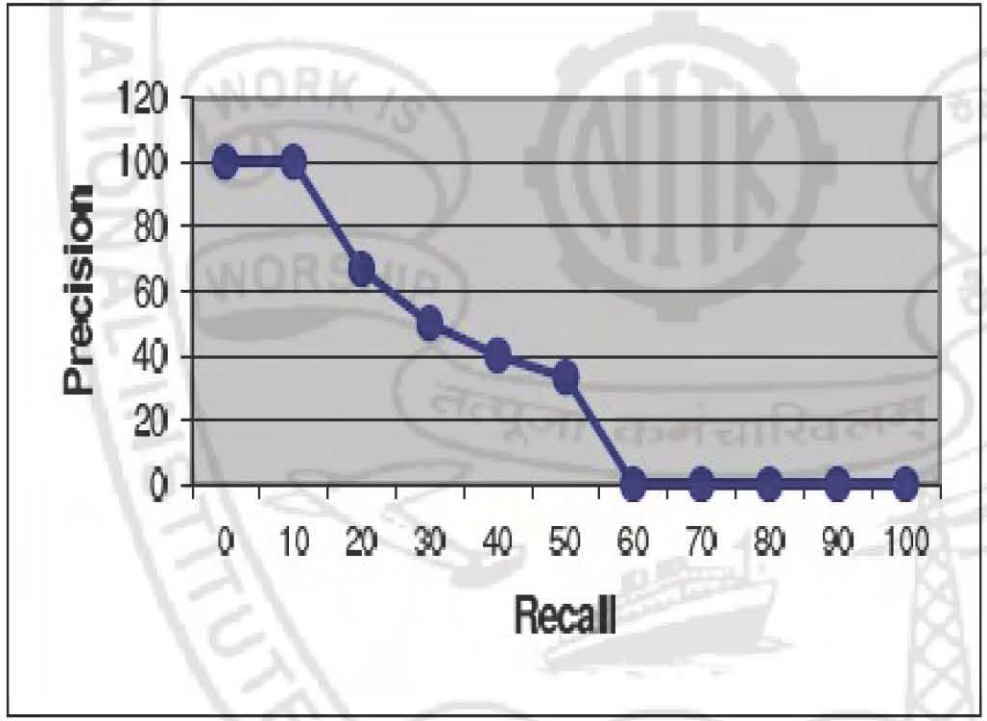
- MAP = Mean average precision for a set of queries is the mean of the average precision scores for each query.

$$\text{MAP} = \frac{\sum_{q=1}^{Q} \text{AveP}(q)}{Q}$$

* When a relevant document is not retrieved at all, the precision value in the above equation is taken to be 0.

MAP – An example

- Consider again the precision-recall curve for the example query q1.



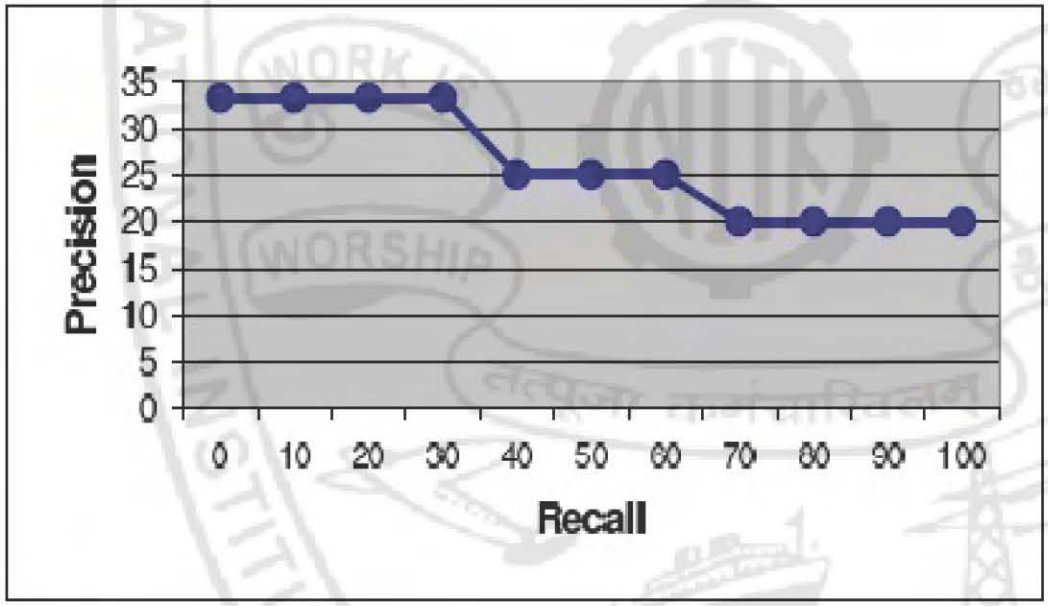
Recall	Precision
0	100
10	100
20	66.6
30	50
40	40
50	33.3
60	0
70	0
80	0
90	0
100	0

- The mean average precision (MAP) for q1 is given by =

$$MAPq_1 = \frac{1 + 0.66 + 0.5 + 0.4 + 0.33 + 0 + 0 + 0 + 0 + 0}{10} = 0.28$$

MAP – An example

- Consider again the precision-recall curve for the other query q2.



Recall	Precision
0	33.3
10	33.3
20	33.3
30	33.3
40	25
50	25
60	25
70	20
80	20
90	20
100	20

- The mean average precision (MAP) for q2 is given by =

$$MAPq_2 = \frac{0.33+0.33+0.33+0.25+0.25+0.25+0.2+0.2+0.2+0.2}{10} = 0.254$$

MAP – An example

- MAP for set of queries Q={q1, q2} is given by –

$$\text{MAP} = \frac{\sum_{q=1}^{Q} \text{AveP}(q)}{Q}$$

$$\text{MAP} = \frac{0.28 + 0.254}{2} = 0.267$$

Geometric Mean Average Precision (GMAP)

- designed for situations where you want to highlight improvements for low-performing topics.
- introduced in the TREC 2004 robust track.
- GMAP is the geometric mean of per-topic average precision,
  - in contrast, MAP is the arithmetic mean

Geometric Mean Average Precision (GMAP)

- geometric mean of the average precision values for an IR system over a set of n queries.

$$\text{GMAP} = \sqrt[n]{\prod_{n} AP_n}$$

$$\text{GMAP} = \exp \frac{1}{n} \sum_{n} \log AP_n$$

  - Where, AP represents the Average Precision value for a given query the evaluation set of n queries.

GMAP - example

	Topic #1	Topic #2	Topic #3	Topic #4	Topic #5
Run 1 AP	0.05	0.10	0.50	0.50	0.75
Run 2 AP	0.10	0.30	0.45	0.45	0.60

E.g. Run 1 and Run 2 have the same MAP of 0.38

Run 1 has a GMAP of 0.25 and Run 2 has a GMAP of 0.33

→ evaluator is interested in a measure of consistency and collective performance across all topics, so GMAP is a better choice as the evaluation metric.

Mean Reciprocal Rank (MRR)

- Used for those cases where user is interested in the first correct answer
- Examples:
  - Question-Answering (QA) systems
  - Search engine queries that look for specific sites
    - URL queries
    - Homepage queries

Mean Reciprocal Rank (MRR)

- Let
  - R_i: ranking relative to a query q_i
  - $S_{correct}(R_i)$: position of the first correct answer in R_i
  - S_h: threshold for ranking position

- Then, the reciprocal rank $RR(R_i)$ for query q_i is -

$$RR(\mathcal{R}_i) = \begin{cases} \frac{1}{S_{correct}(\mathcal{R}_i)} & \text{if } S_{correct}(\mathcal{R}_i) \leq S_h \\ 0 & \text{otherwise} \end{cases}$$

- And, the MRR for a set Q of N_q queries is -

$$MRR(Q) = \sum_i^{N_q} RR(\mathcal{R}_i) \,/\, Q$$

Mean Reciprocal Rank (MRR) – An example

- Consider again the IR algorithm with the following ranking for q_1.

01. d_{123} ●	06. d_9 ●	11. d_{38}
02. d_{84}	07. d_{511}	12. d_{48}
03. d_{56} ●	08. d_{129}	13. d_{250}
04. d_6	09. d_{187}	14. d_{113}
05. d_8	10. d_{25} ●	15. d_3 ●

- Let $S_h = 3$ *(threshold for ranking position)*
- $S_{correct}(R_i) = 1$ *(position of the first correct answer in R_i)*

$$\mathbf{RR(R_{i,q1}) = 1/1 = 1} \quad (as \ S_{correct}(R_i) \leq S_h)$$

Mean Reciprocal Rank (MRR) – An example

- Consider again the IR algorithm with the following ranking for q_2 .

01. d_{425}	06. d_{615}	11. d_{193}
02. d_{87}	07. d_{512}	12. d_{715}
03. d_{56} ●	08. d_{129} ●	13. d_{810}
04. d_{32}	09. d_4	14. d_5
05. d_{124}	10. d_{130}	15. d_3 ●

- Let $S_h = 3$ *(threshold for ranking position)*
- $S_{correct}(R_i) = 3$ *(position of the first correct answer in R_i)*

$$\mathbf{RR(R_{i,q2}) = 0.33} \quad (as \ S_{correct}(R_i) \leq S_h)$$

Mean Reciprocal Rank (MRR) – An example

$$RR(R_{i,q1}) = 1$$

$$RR(R_{i,q2}) = 0.33$$

$$MRR(R_i) = (1 + 0.33)/2 = 0.67$$

F-Measure: Harmonic Mean

- a single measure that combines recall and precision.
  - derived by Prof. Keith van Rijsbergen (1979)

- "measures the effectiveness of retrieval with respect to a user who attaches ß times as much importance to recall as precision".

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

  - for non-negative real values of ß.

F-Measure: Harmonic Mean

- ***F_1 measure*** ***($\beta = 1$)***

$$F_1 \ measure = 2.\left\{\frac{P.R}{P+R}\right\}$$

  - traditional F-measure or balanced F-score.
  - recall and precision are evenly weighted.

- ***F_2 measure*** ***($\beta = 2$)***

$$F_2 \ measure = 5.\left\{\frac{P.R}{4P+R}\right\}$$

  - weights recall higher than precision

- ***$F_{0.5}$ measure*** ***($\beta = 0.5$)***

$$F_{0.5} \ measure = 1.25 \times \left\{\frac{P.R}{(0.5)P+R}\right\}$$

  - weights precision higher than recall

F-Measure – Homework Exercise

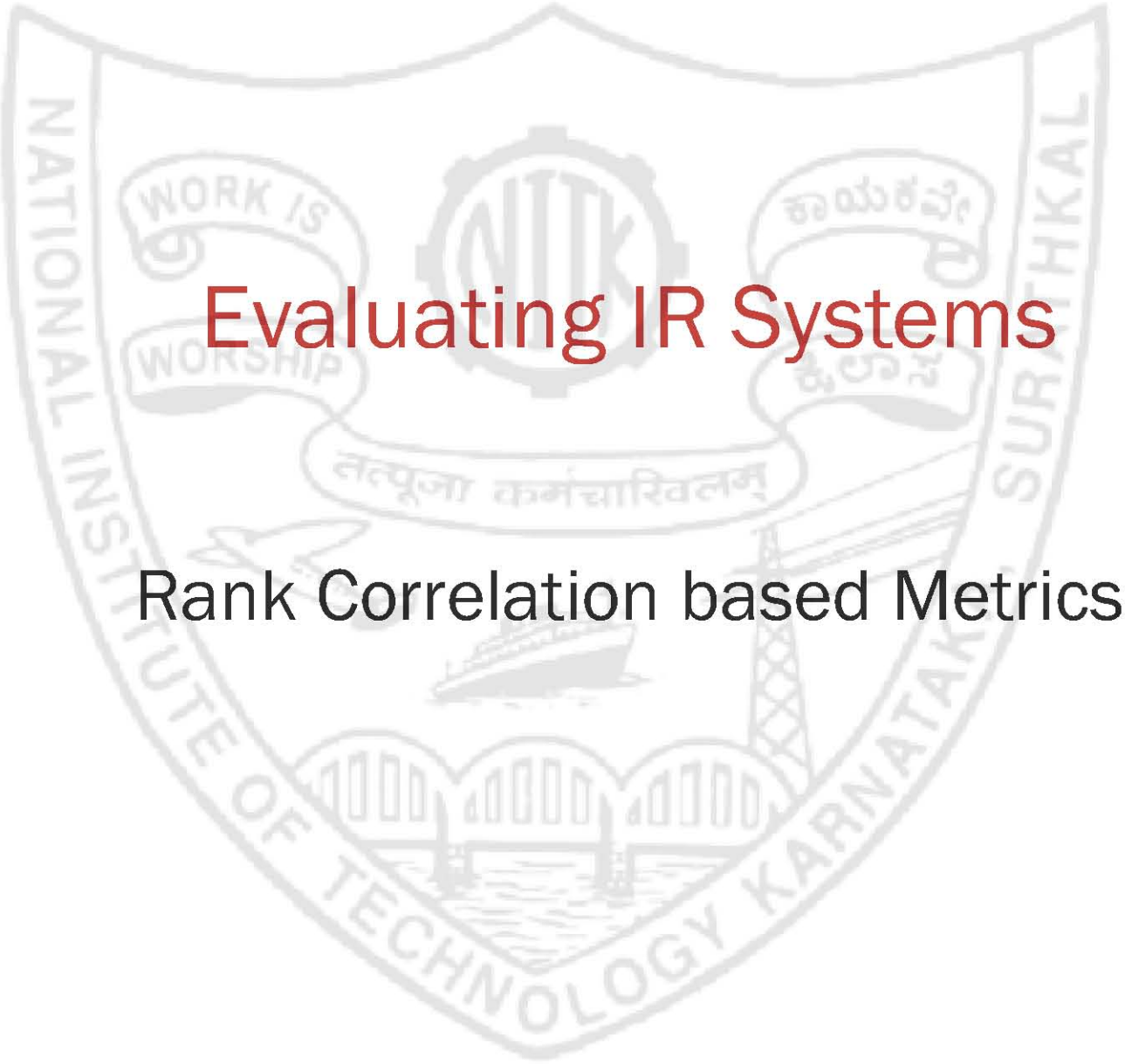
- Compute the various F-measure values using the P and R values computed in class for the given sample queries Q1 and Q2. Note your observations w.r.t to the designed IR algorithm's balanced, recall-oriented and precision-oriented performance.

Q1

01. d_{123} ●	06. d_9 ●	11. d_{38}
02. d_{84}	07. d_{511}	12. d_{48}
03. d_{56} ●	08. d_{129}	13. d_{250}
04. d_6	09. d_{187}	14. d_{113}
05. d_8	10. d_{25} ●	15. d_3 ●

Q2

01. d_{425}	06. d_{615}	11. d_{193}
02. d_{87}	07. d_{512}	12. d_{715}
03. d_{56} ●	08. d_{129} ●	13. d_{810}
04. d_{32}	09. d_4	14. d_5
05. d_{124}	10. d_{130}	15. d_3 ●



Evaluating IR Systems

Rank Correlation based Metrics

Rank Correlation Metrics

- Helpful for determining how differently a new ranking function varies from one that we know well.
  - Allows comparison between the relative ordering produced by the two rankings.
- Statistical functions based evaluation metrics
  - E.g. Spearman Co-efficient Test.

Rank Correlation Metrics

- Let the two rankings to be compared be R_1 and R_2
- A rank correlation metric yields a correlation coefficient $C(R_1, R_2)$ such that $-1 \leq C(R_1, R_2) \leq +1$,
  - If $C(R_1, R_2) = 1$, the two rankings are exactly the same.
  - If $C(R_1, R_2) = -1$, the two rankings are reverse of each other.
  - If $C(R_1, R_2) = 0$, the two rankings are completely independent.

* *increasing values of $C(R_1, R_2)$ imply improved agreement between the two rankings.*

Spearman Rank Correlation Co-efficient (SRCC)

- most used rank correlation metric
  - Computed using the differences between the positions of a same document in two rankings.
  - Method:
    - Examine rank of each document in both rankings
    - Apply correlation coefficient rules to assess relationship
    - Compute SRCC for overall ranking.

Spearman Rank Correlation Co-efficient (SRCC)

- Let $s_{1,j}$ and $s_{2,j}$ be the position of d_j in ranking R_1 and R_2.

- $$\text{SRCC} = S(\mathcal{R}_1, \mathcal{R}_2) = 1 - \frac{6 \times \sum_{j=1}^{K} (s_{1,j} - s_{2,j})^2}{K \times (K^2 - 1)}$$

  - Where, K = the number of the ranked answer sets.

SRCC – class exercise

- Consider the following rankings obtained for a given reference collection. R1 is the ideal ranking, and R2 is the ranking generated by a new ranking function. Calculate the relative ranking performance of the new ranking function.

documents	R1	R2
d_{123}	1	2
d_{84}	2	3
d_{56}	3	1
d_6	4	5
d_8	5	4
d_9	6	7
d_{511}	7	8
d_{129}	8	10
d_{187}	9	6
d_{25}	10	9

SRCC – class exercise

$$S(\mathcal{R}_1, \mathcal{R}_2) = 1 - \frac{6 \times \sum_{j=1}^{K} (s_{1,j} - s_{2,j})^2}{K \times (K^2 - 1)}$$

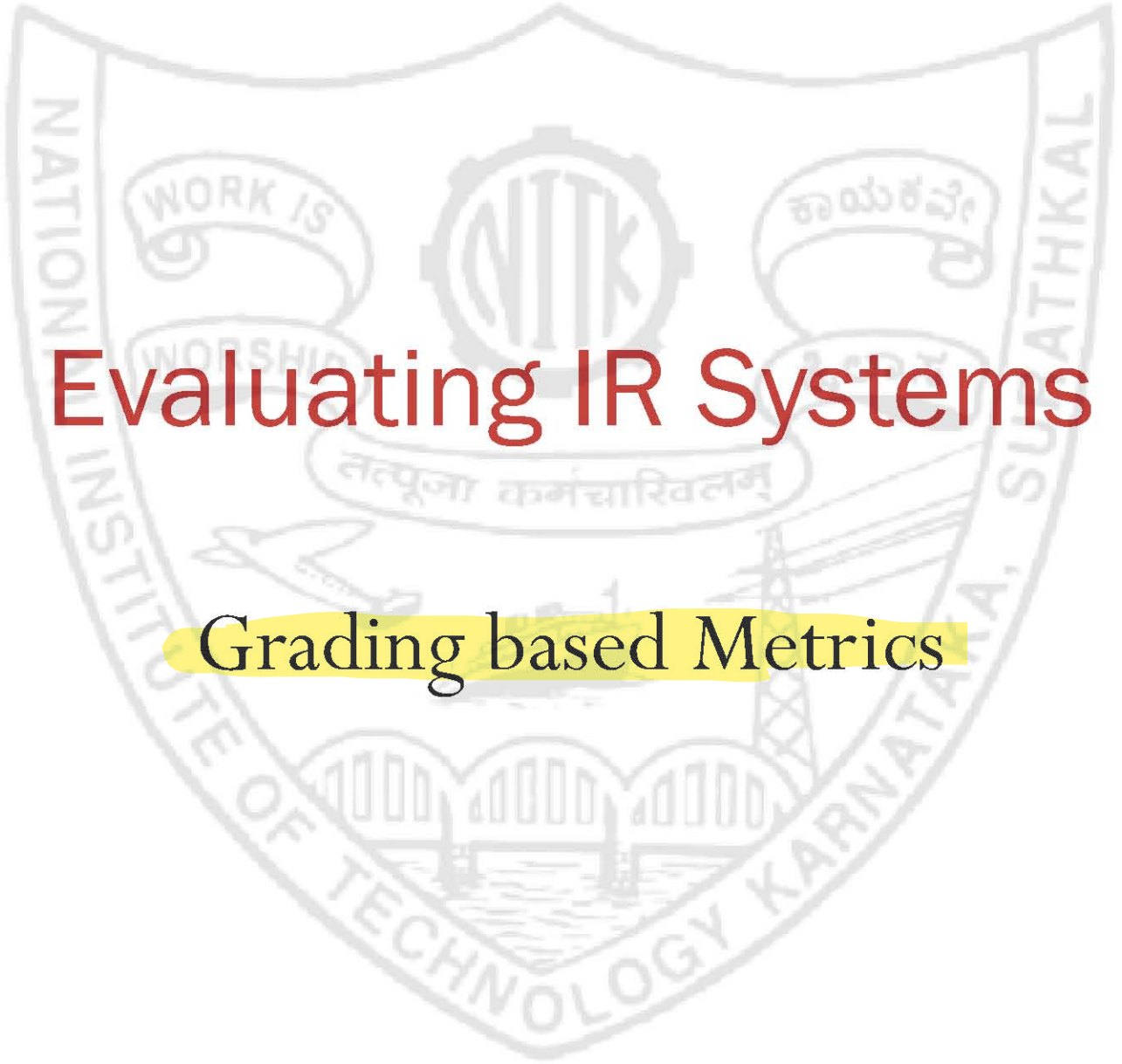
documents	$s_{1,j}$	$s_{2,j}$	$s_{i,j} - s_{2,j}$	$(s_{1,j} - s_{2,j})^2$
d_{123}	1	2	-1	1
d_{84}	2	3	-1	1
d_{56}	3	1	+2	4
d_6	4	5	-1	1
d_8	5	4	+1	1
d_9	6	7	-1	1
d_{511}	7	8	-1	1
d_{129}	8	10	-2	4
d_{187}	9	6	+3	9
d_{25}	10	9	+1	1
Sum of Square Distances				24

SRCC – class exercise

$$S(\mathcal{R}_1, \mathcal{R}_2) = 1 - \frac{6 \times \sum_{j=1}^{K} (s_{1,j} - s_{2,j})^2}{K \times (K^2 - 1)}$$

$$S(\mathcal{R}_1, \mathcal{R}_2) = 1 - \frac{6 \times 24}{10 \times (10^2 - 1)} = 0.854$$

→ Indicates a strong positive correlation between the ranking generated by the new function, with the ideal ranking.



Evaluating IR Systems

Grading based Metrics

Relevance assessment

- Issue with most metrics: binary relevance assessments
- no distinction between highly relevant docs and mildly relevant docs.
- Need:
  - Graded relevance assessments
  - Metrics that support such graded assessments.
- Solution: **Normalized Discounted Cumulative Gain (NDCG)**

Discounted Cumulated Gain (DCG)

- Idea:
  - Use graded relevance as a measure of usefulness, or gain, by examining a document.
  - Gain is accumulated starting at the top of the ranking
    - may be reduced or discounted for documents at lower ranks.

Discounted Cumulated Gain (DCG)

- Process:
  - Examine the results of a query conditionally.
  - Conditions applied - Graded relevance
    - highly relevant documents are preferable **at the top** of the ranking than mildly relevant ones.
    - relevant documents that appear at the end of the ranking are less valuable.

Towards NDCG Formulation

- Let the rankings of the ***n*** documents be $\mathbf{rel_1, rel_2, \ldots rel_n}$ (in ranked order)

- **Cumulative Gain (CG) at rank *n***

$$CG = rel_1 + rel_2 + \ldots rel_n$$

** Change in ranking order at any rank does not affect the CG metric.*

Towards NDCG Formulation

- **Discounted Cumulative Gain (DCG) at rank *n***

$$DCG_n = rel_1 + rel_2/\log_2 2 + rel_3/\log_2 3 + \ldots rel_n/\log_2 n$$

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

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

log base 2!!!!!!!!!!

- Final formulation:

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

  - Emphasizes the retrieval of highly relevant documents

Revised Reference Collections

- Consider the relevance judgments for sample query q1 .

$$R_{q_1} = \{d_3, d_5, d_9, d_{25}, d_{39}, d_{44}, d_{56}, d_{71}, d_{89}, d_{123}\}$$

- Revise the above by graded relevant documents on a scale 0–3
  - 0 - non-relevant …… **3 for strongly relevant docs**
- Then, graded relevance scores for q1 are –

$$R_{q_1} = \{ [d_3, 3], [d_5, 3], [d_9, 3], [d_{25}, 2], [d_{39}, 2], [d_{44}, 2], [d_{56}, 1], [d_{71}, 1], [d_{89}, 1], [d_{123}, 1] \}$$

$$= 3,3,3,2,2,2,1,1,1,1$$ → ***Ideal Ranking***

Revised Reference Collections

- Consider the relevance judgments for sample query q1 .

$$R_{q_1} = \{d_3, d_5, d_9, d_{25}, d_{39}, d_{44}, d_{56}, d_{71}, d_{89}, d_{123}\}$$

- Revise the above by graded relevant documents on a scale 0–3
  - 0 - non-relevant …… 3 for strongly relevant docs

- Then, graded relevance scores for q1 are –

$$R_{q_1} = \{ [d_3, 3], [d_5, 3], [d_9, 3], [d_{25}, 2], [d_{39}, 2], [d_{44}, 2], [d_{56}, 1], [d_{71}, 1], [d_{89}, 1], [d_{123}, 1] \}$$

$$= 3,3,3,2,2,2,1,1,1,1$$ → ***Ideal Ranking***

** i.e, while doc d3 is highly relevant for query q1, doc d56 is just mildly relevant.*

Normalized Discounted Cumulative Gain (NDCG)

- *Ideal ranking*
  - that which returns documents of highest relevance level first, then the next highest relevance level, *and so on*…

- NDCG at rank n -
  - Normalize DCG at rank n using the DCG value at rank n of ideal ranking.

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

Graded Relevance Assessment – Class Exercise

- Consider that an algorithm ranks 6 documents as –

  (d1, d2, d3, d4, d5 ,d6)

  An expert user grades the results on a scale 0–3 *(0 for non-relevant docs ….. 3 for strongly relevant docs)*. Assume that, expert graded relevance scores for above ranking are – 3,2,3,0,1,2.

  Evaluate the new IR algorithm appropriately.

CG, DCG and NDCG – Class Exercise

- Process:
  - Compute CG
  - Compute DCG
  - Compute IDCG
  - Find NDCG -- assess how well your IR system measures up to the Ideal ranking.

Class Exercise

- **Step 2: Compute DCG**
  - Algorithm ranks 6 documents as – (d1, d2, d3, d4, d5 ,d6)
  - Graded relevance scores rel_i - 3,2,3,0,1,2
  - DCG = $\sum_{i=1}^{6} \frac{2^{rel_i}-1}{log(1+i)}$

Class Exercise - *Compute DCG*

- Algorithm ranks 6 documents as (d1, d2, d3, d4, d5 ,d6)
- Graded relevance scores rel_i - 3,2,3,0,1,2

i	rel_i	$\log_2(1+i)$	$(2^{rel_i} - 1)/ \log_2(1+i)$
1	3	1	7
2	2	1.585	1.892
3	3	2	3.5
4	0	2.322	0
5	1	2.585	0.386
6	2	2.807	1.068

- $DCG = \sum_{i=1}^{6} \frac{2^{rel_i}-1}{log(1+i)} = 7+1.892+3.5+0+0.386+1.068 = 13.846$

Class Exercise

- Step 3: Compute IDCG
  - Algorithm ranks 6 documents as – (d1, d2, d3, d4, d5 ,d6)
  - Given graded relevance scores rel_i - 3,2,3,0,1,2
  - Ideal relevance scores $rel_{i(ideal)}$ - **3,3,2,2,1,0**

Class Exercise – *Compute IDCG*

- Algorithm ranks 6 documents as (d1, d2, d3, d4, d5 ,d6)
- Ideal relevance scores rel_i - 3,3,2,2,1,0

i	Rel_i	$\log_2(1+i)$	$(2^{rel_i} - 1)/ \log_2(1+i)$
1	3	1	7
2	3	1.585	4.416
3	2	2	1.5
4	2	2.322	1.286
5	1	2.585	0.386
6	0	2.807	0

- $IDCG = \sum_{i=1}^{6} \frac{2^{rel_i}-1}{log(1+i)} = \mathbf{14.588}$

Class Exercise – *Compute NDCG*

- DCG of IR system's ordering IDCG = **13.846**
- DCG of this ideal ordering IDCG = **14.588**
- Normalized DCG $NDCG = \frac{DCG}{IDCG} = \mathbf{0.949}$
- **Inference: New IR algorithm has performed very well.**

DCG and NDCG - Limitations

- does not penalize for including bad documents in the result.
  - E.g. if a query returns two results with scores {1,1,1} and {1,1,1,0} respectively, both would be considered equally good even if the latter contains a bad result.

DCG and NDCG - Limitations

- does not penalize for missing documents in the result.
  - For example: if a query returns two results with scores {1,1,1} and {1,1,1,1,1} respectively, both would be considered equally good.

DCG and NDCG - Limitations

- may not be suitable to measure performance of queries that may typically have several equally good results.
  - E.g. for queries such as "restaurants", if one result set contains only 1 restaurant from the nearby area while the other contains 5 *(may be less relevant, but still are useful to user)*, both may have same score even though latter is more comprehensive.

Evaluating IR Systems

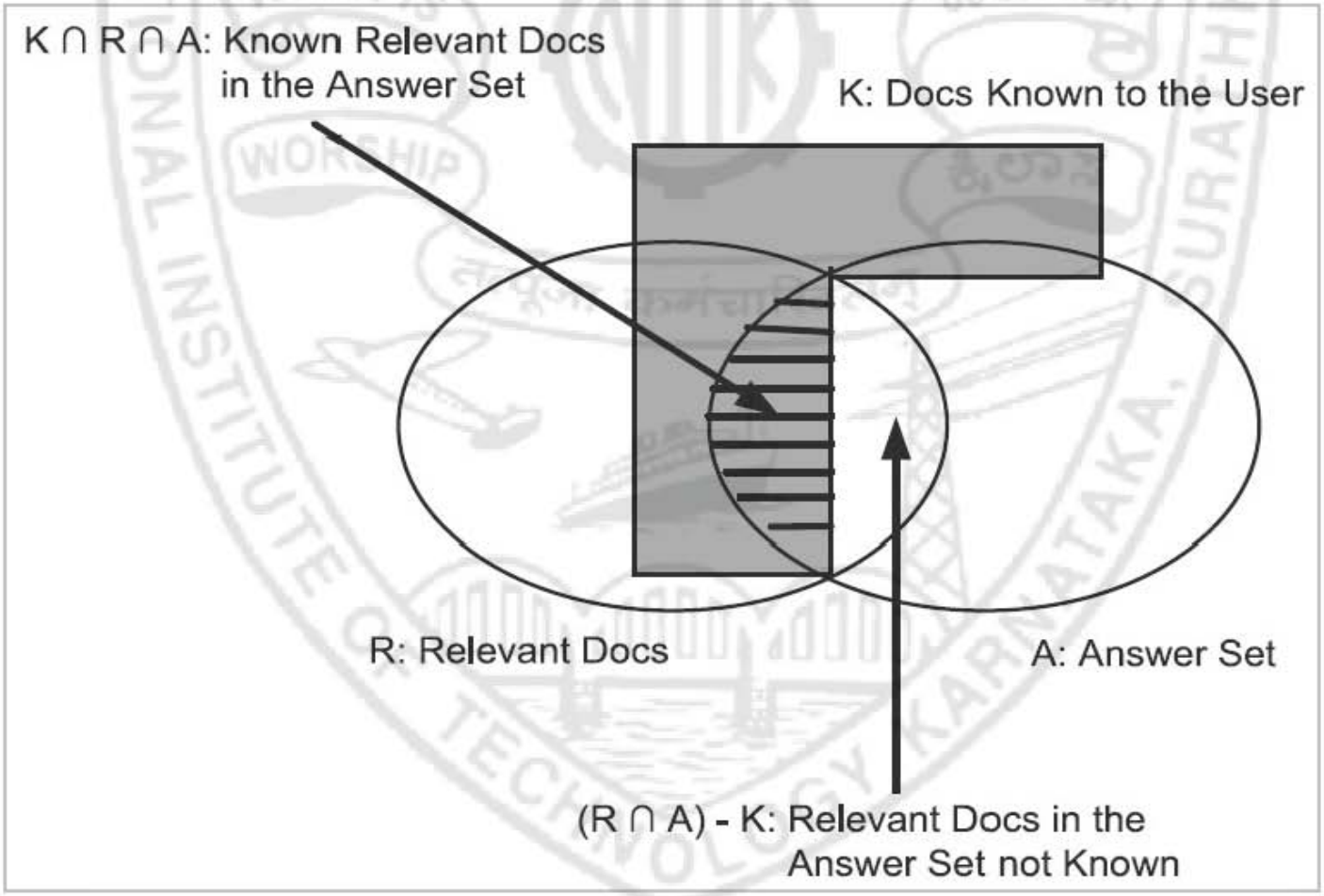
User oriented Metrics

User-oriented Metrics

- Recall and precision -
  - Assumption is that a query's set of relevant docs is independent of users.
- In reality, relevance is highly subjective!!
  - Solution: User-oriented measures.

- Given: a reference collection, information request I, & IR algo
  - For I : R = set of relevant documents

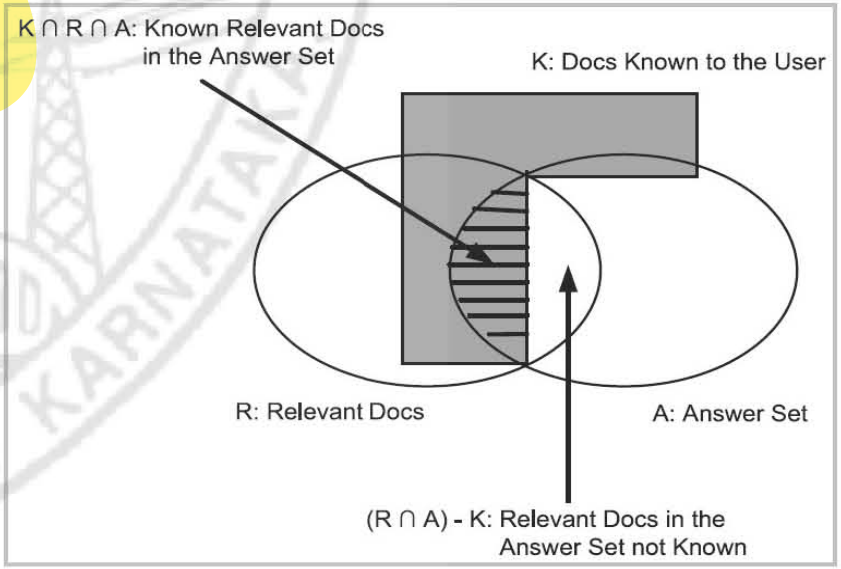
    A = set of answers retrieved.



User-oriented Metrics

- **Coverage ratio**
  - fraction of documents known and relevant that are in the answer set generated by IR algorithm.

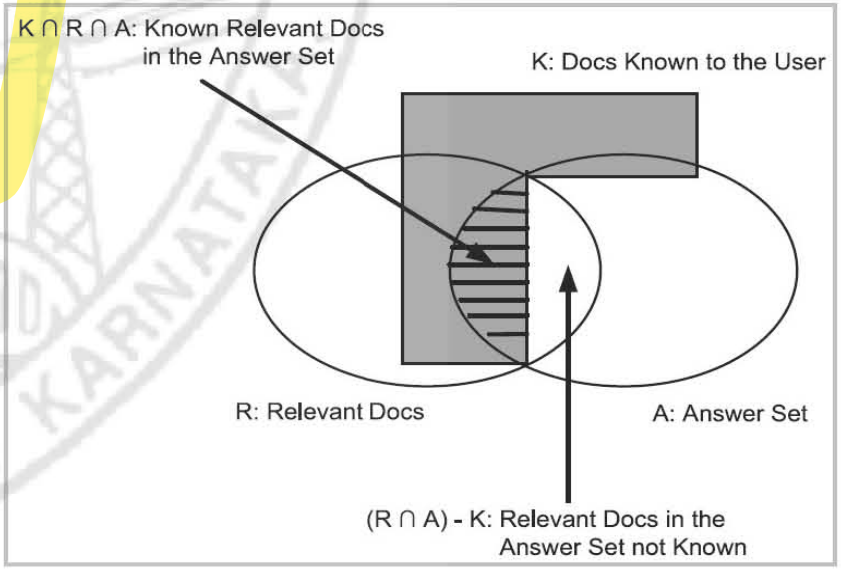
$$coverage = \frac{|K \cap R \cap A|}{|K \cap R|}$$



User-oriented Metrics

- **Novelty ratio**
  - fraction of the relevant docs in the answer set that are not known to the user

$$novelty = \frac{|(R \cap A) - K|}{|R \cap A|}$$



User-oriented Metrics

- **Relative Recall**
  - ratio of the number of relevant docs found to the number of relevant docs the user expected to find.

Relative Recall = # found / # expected

User-oriented Metrics

- **Recall Effort**
  - ratio of number of relevant docs the user expected to find to the number of documents examined in an attempt to find the expected relevant documents.

Recall effort = # expected / # examined until all are found

Other user-oriented metrics

- **Expected search length**
  - a way to estimate the number of documents that a user has to go through in order to find the desired number of relevant documents.
  - good for weakly ordered rankings.
- **Satisfaction**
  - considers only % relevant docs returned
- **Frustration**
  - considers only % non-relevant docs returned

User-oriented Metrics - Summary

- **High coverage** - indicates that the system has found most of the relevant docs the user expected to see.
- **High novelty** - indicates that the system is revealing many new relevant docs which were unknown.
- **High satisfaction/low frustration** - indicates good IR performance.
- **Reasonably low** expected search length is good from user's perspective.

Further Reading...

- "Introduction to Information Retrieval - Evaluation" (PDF). Stanford University.

- TREC Common IR Metrics
  https://trec.nist.gov/pubs/trec15/appendices/CE.MEASURES06.pdf

- S Gupta, M Kutlu, V Khetan and M Lease, *Correlation, Prediction and Ranking of Evaluation Metrics in Information Retrieval, ECIR*