Evaluation & Benchmark Collections

Kuan-Yu Chen (陳冠宇)

2020/10/30 @ TR-313, NTUST

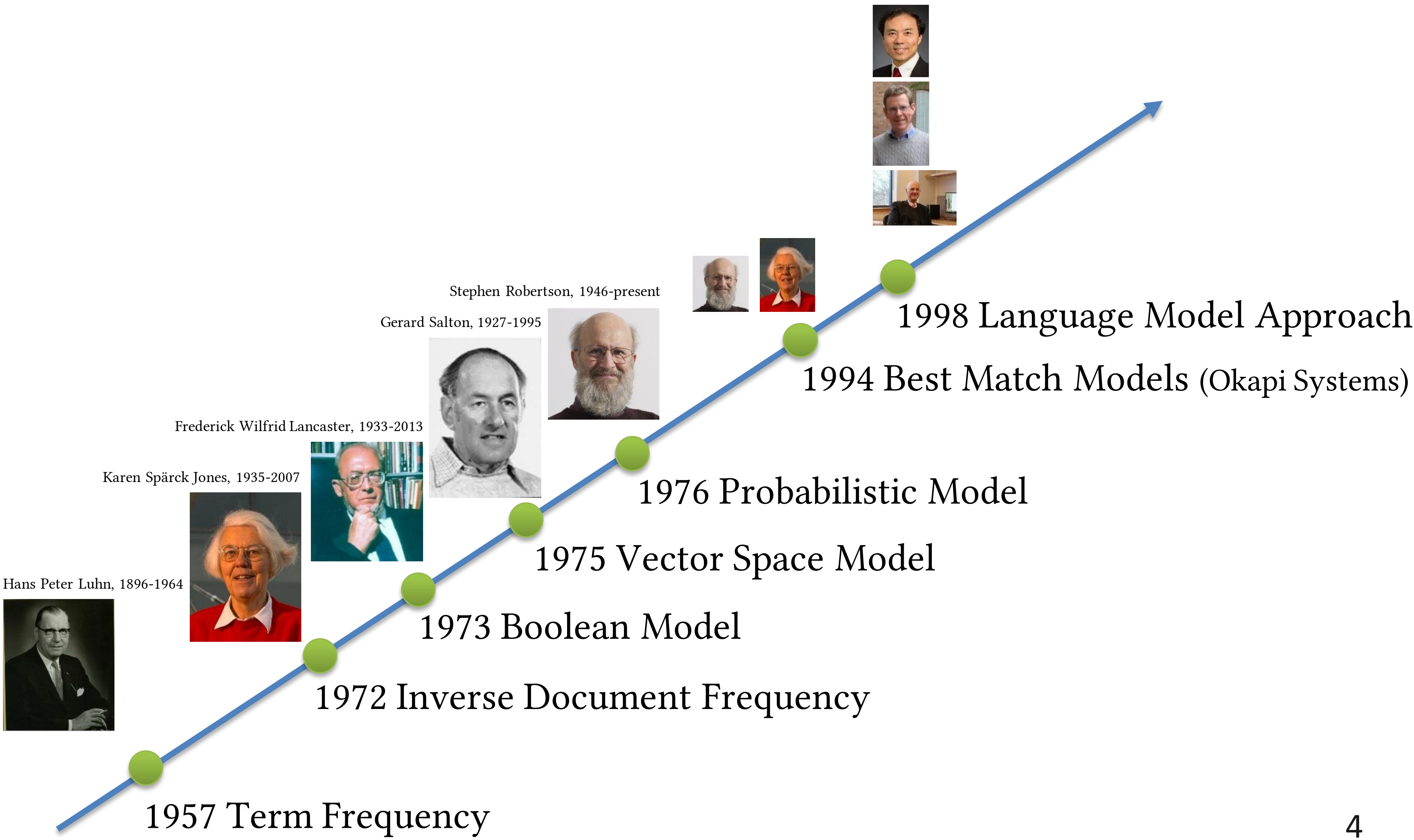
HW1 – VSM

#	△pub	Team Name	Notebook	Team Members	Score	Entries	Last
1	—	M10915010_盧克函			0.78812	86	17h
2	—	__incognito__			0.77476	30	3d
3	—	M10907505_游照臨			0.75714	45	21h
4	—	M10915201_陳牧凡			0.75664	11	12d
5	—	M10915036_王繹崴			0.75478	20	2d
6	—	M10915012_黃偉愷			0.75474	48	7d
7	—	M10815036_王仁德			0.75342	54	1d
8	—	M10915031_鄭善謙			0.74897	12	3d
9	—	B10615034_黃柏翰			0.74689	28	6d
10	—	M10815048_張晏銘			0.74659	26	2d

HW2 – BM25

#	Team Name	Notebook	Team Members	Score	Entries	Last
1	我們不要BM25了			0.77480	32	3d
2	這是VSM			0.75719	81	3d
3	QQ baseline			0.74656	42	17h
4	快使用!!!神奇喵喵黑魔法!!!			0.74532	28	21h
5	Enilesab			0.73148	44	10h
6	了不讓			0.72760	60	14h
7	B10615022_姜宏昀			0.72613	32	4d
8	M10915201_陳牧凡			0.72585	62	15h
9	B10615034_黃柏翰			0.72549	59	2d
10	80847002S_羅天宏			0.72395	2	1d

Review


1998 Language Model Approach
1994 Best Match Models (Okapi Systems)
Stephen Robertson, 1946-present
Gerard Salton, 1927-1995
1976 Probabilistic Model
Frederick Wilfrid Lancaster, 1933-2013
1975 Vector Space Model
Karen Spärck Jones, 1935-2007
1973 Boolean Model
Hans Peter Luhn, 1896-1964
1972 Inverse Document Frequency
1957 Term Frequency

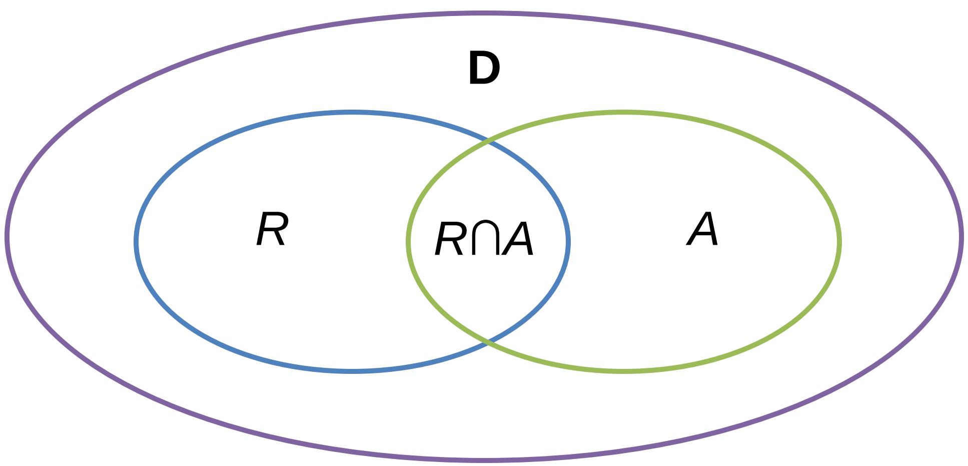
Introduction

- To evaluate an IR system is to measure how well the system meets the information needs of the users
  - This is troublesome, given that a same result set might be interpreted differently by distinct users

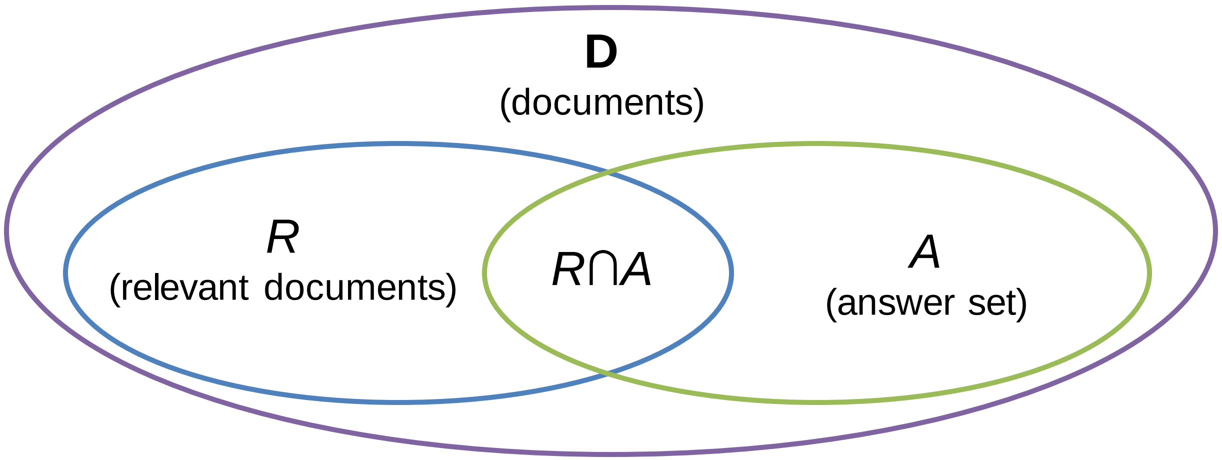
- Without proper retrieval evaluation, one cannot
  - determine how well the IR system is performing
  - objectively compare the performance of the IR system with that of other systems

Notations

- For a given query (information need)
  - **D**: the set of documents
  - R: the set of relevant documents
  - A: the answer set generated by an IR system
  - $R \cap A$: relevant documents in the answer set



Precision & Recall – Definition



- **Precision** (準確率) is the fraction of the retrieved documents which is relevant

$$Precision = \frac{|R \cap A|}{|A|}$$

- **Recall** (召回率) is the fraction of the relevant documents which has been retrieved

$$Recall = \frac{|R \cap A|}{|R|}$$

Precision & Recall

- The definition of precision and recall assumes that all documents in the answer set have been examined

- In reality, user sees a ranked set of documents and examines them starting from the top
  - Precision and recall vary as the user proceeds with their examination of the answer set

- Most appropriate then is to plot a **curve of precision versus recall**

Example – 1.

- For a given query q and a set of relevant documents R_q for the query

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

- If an IR model that provides a ranking list for the query q

1. d_{123}	6. d_9	11. d_{38}
2. d_{84}	7. d_{511}	12. d_{48}
3. d_{56}	8. d_{129}	13. d_{250}
4. d_6	9. d_{187}	14. d_{113}
5. d_8	10. d_{25}	15. d_3

Example – 1..

$Recall = \frac{|R \cap A|}{|R|}$

$Precision = \frac{|R \cap A|}{|A|}$

- If we examine this ranking, we observe that
  - The document d_{123}, ranked as number 1, is relevant
    - This document corresponds to 10% of all relevant documents
    - Thus, we say that we have a **precision of 100% at 10% recall**
  - The document d_{56}, ranked as number 3, is the next relevant
    - At this point, two documents out of three are relevant, and two of the ten relevant documents have been seen
    - Thus, we say that we have a **precision of 66.6% at 20% recall**

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
	●		●			●				●					●
R(%)	10		20			30				40					50
P(%)	100		66.6			50				40					33.3

Example – 2.

- For a given query q and a set of relevant documents R_q for the query

$$R_q = \{d_3, d_{56}, d_{129}\}$$

- If an IR model that provides a ranking list for the query q

1. d_{123}	6. d_9	11. d_{38}
2. d_{84}	7. d_{511}	12. d_{48}
3. d_{56}	8. d_{129}	13. d_{250}
4. d_6	9. d_{187}	14. d_{113}
5. d_8	10. d_{25}	15. d_3

Example – 2..

- If we examine this ranking, we observe that
  - The first relevant document is d_{56}
    - It provides a recall and precision levels equal to 33.3%
  - The second relevant document is d_{129}
    - It provides a recall level of 66.6% (with precision equal to 25%)
  - The third relevant document is d_3
    - It provides a recall level of 100% (with precision equal to 20%)

$$Recall = \frac{|R \cap A|}{|R|} \quad Precision = \frac{|R \cap A|}{|A|}$$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
			●					●							●
R(%)			33.3					66.6							100
P(%)			33.3					25							20

Interpolated Precision.

- An **interpolated precision** at a standard 11 recall level can be calculated

R	0	10	20	30	40	50	60	70	80	90	100
P											

$(R,P) = (33.3\%, 33.3\%)$ $(R,P) = (66.6\%, 25\%)$ $(R,P) = (100\%, 20\%)$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
			●					●							●
R(%)			33.3					66.6							100
P(%)			33.3					25							20

Interpolated Precision..

- An **interpolated precision** at a standard 11 recall level can be calculated

$$\bar{P}(r) = max_{r' \geq r} P(r')$$

$$\bar{P}(20) = max_{r' \geq 20} P(r') = P(33.3) = 33.3\%$$

$$\bar{P}(0) = max_{r' \geq 0} P(r') = P(33.3) = 33.3\%$$

R	0	10	20	30	40	50	60	70	80	90	100
P	33.3	33.3	33.3	33.3							

$(R, P) = (33.3\%, 33.3\%)$ $(R, P) = (66.6\%, 25\%)$ $(R, P) = (100\%, 20\%)$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
			●					●							●
R(%)			33.3					66.6							100
P(%)			33.3					25							20

Interpolated Precision...

- An **interpolated precision** at a standard 11 recall level can be calculated

$$\bar{P}(r) = max_{r' \geq r} P(r')$$

$$\bar{P}(40) = max_{r' \geq 40} P(r') = P(66.6) = 25\%$$

R	0	10	20	30	40	50	60	70	80	90	100
P	33.3	33.3	33.3	33.3	25	25	25				

$(R, P) = (33.3\%, 33.3\%)$ $(R, P) = (66.6\%, 25\%)$ $(R, P) = (100\%, 20\%)$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
			●					●							●
R(%)			33.3					66.6							100
P(%)			33.3					25							20

Interpolated Precision….

- An **interpolated precision** at a standard 11 recall level can be calculated

$$\bar{P}(r) = max_{r' \geq r} P(r')$$

$$\bar{P}(70) = max_{r' \geq 70} P(r') = P(100) = 20\%$$

R	0	10	20	30	40	50	60	70	80	90	100
P	33.3	33.3	33.3	33.3	25	25	25	20	20	20	20

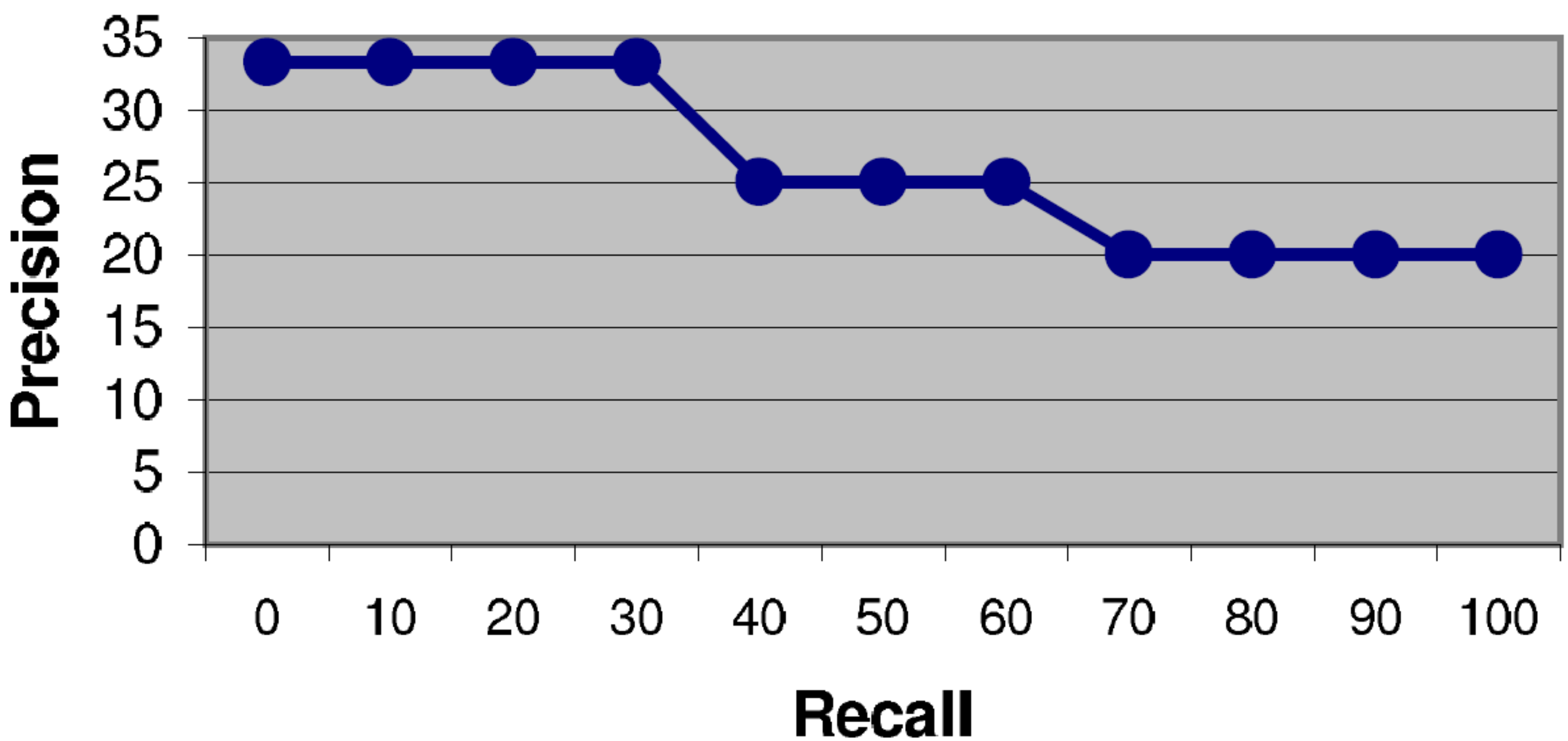
$(R, P) = (33.3\%, 33.3\%)$ $(R, P) = (66.6\%, 25\%)$ $(R, P) = (100\%, 20\%)$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
			●					●							●
R(%)			33.3					66.6							100
P(%)			33.3					25							20

Interpolated Recall-Precision Curve

- Based on the interpolated precision, an **interpolated recall-precision curve** can be illustrated

R	0	10	20	30	40	50	60	70	80	90	100
P	33.3	33.3	33.3	33.3	25	25	25	20	20	20	20



Average Recall-Precision Curve – 1

- Usually, retrieval algorithms are evaluated by running them for several distinct test queries

- To evaluate the retrieval performance for $|\mathbf{Q}|$ queries, we average the precision at each recall level as follows

$$\bar{P}'(r) = \sum_{i=1}^{|\mathbf{Q}|} \frac{\bar{P}_i(r)}{|\mathbf{Q}|}$$

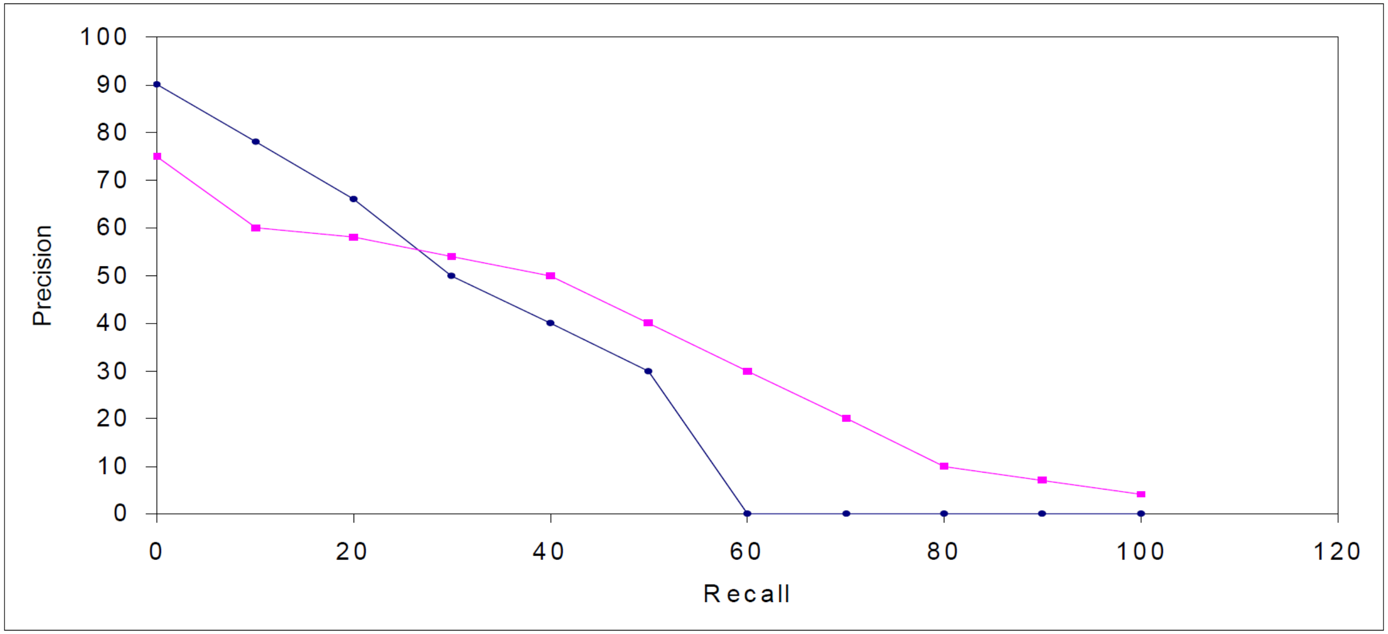
  - $\bar{P}'(r)$ is the average precision at the recall level r
  - $\bar{P}_i(r)$ is the precision at recall level r for the i-th query

	R	0	10	20	30	40	50	60	70	80	90	100
q_1	P	33.3	33.3	33.3	33.3	25	25	25	20	20	20	20
q_2	P	50	50	50	40	30	30	30	20	20	20	10
	Avg.	41.65	41.65	41.65	36.65	27.5	27.5	27.5	20	20	20	15

Average Recall-Precision Curve – 2

- Average precision-recall curves are normally used to compare the performance of distinct IR algorithms

- The figure below illustrates average precision-recall curves for two distinct retrieval algorithms
  - Difficult to figure out that which system is better!

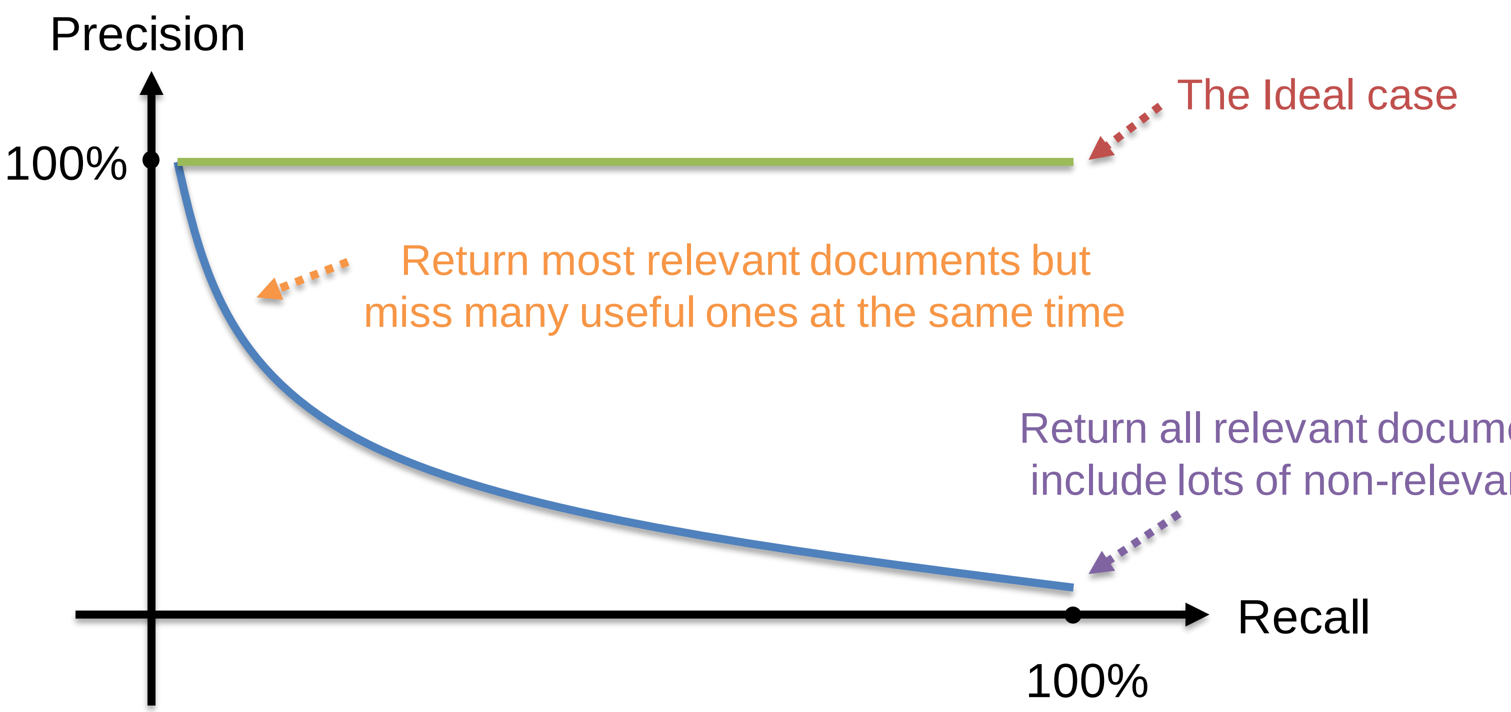


$$Recall = \frac{|R \cap A|}{|R|}$$

$$Precision = \frac{|R \cap A|}{|A|}$$

Recall-Precision Curve

- Trade-off between recall and precision



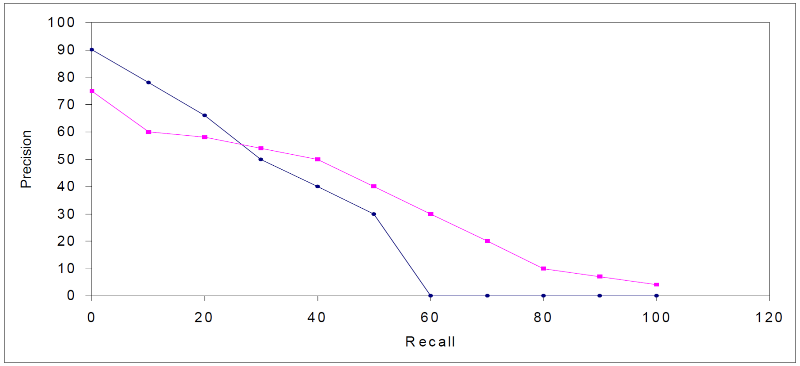
$$Recall = \frac{|R \cap A|}{|R|}$$

$$Precision = \frac{|R \cap A|}{|A|}$$

Pros and Cons

- Advantages
  - Simple, intuitive, and combined in single curve
  - Provide quantitative evaluation of the answer set and comparison among retrieval algorithms
  - A standard evaluation strategy for IR systems

- Disadvantages
  - The estimation of recall score for a query requires detailed knowledge of all the documents in the collection
  - For systems which require a weak ordering though, recall and precision might be inadequate



Single Value Summaries – Precision@K

- Precision@K
  - A single value summary measure the precision when first K retrieved documents have been seen
  - **It favors systems which retrieve relevant docs quickly**
    - In the case of Web search engines, the majority of searches does not require high recall
    - Higher the number of relevant documents at the top of the ranking, more positive is the impression of the users

$$P@5 = \frac{2}{5} = 0.4 \qquad P@15 = \frac{5}{15} = 0.33$$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
	●		●			●				●					●
P(%)	100		66.6			50				40					33.3

Single Value Summaries – *R*-Precision

- R is the total number of relevant documents for a given query

- R-Precision is to compute the precision at the R-th position in the ranking list
  - For the first query: $R - Precision = \frac{2}{5} = 40\%$

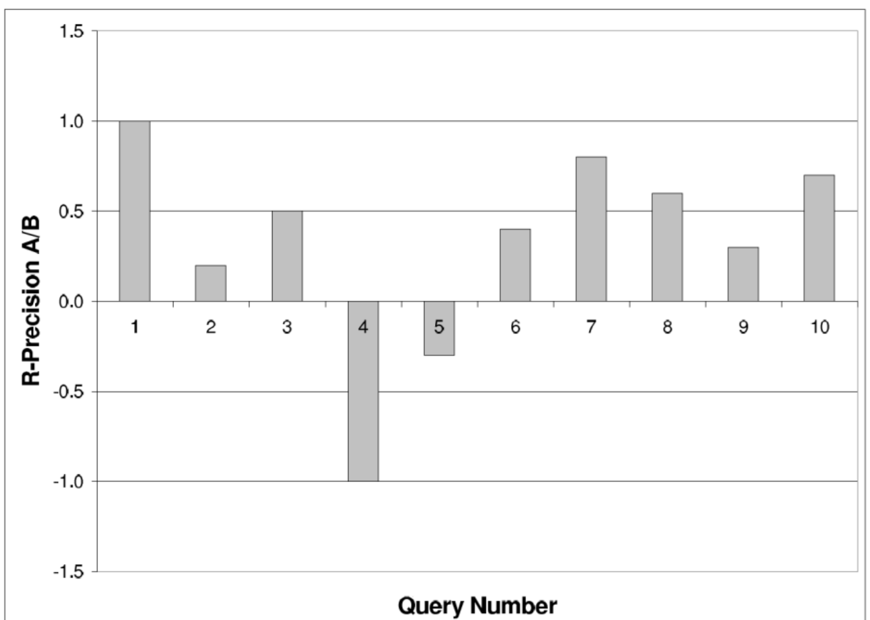
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
	●		●			●				●					●
P(%)	100		66.6			50				40					33.3

  - For the second query: $R - Precision = \frac{1}{3} = 33.3\%$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{84}	d_{56}	d_{123}	d_{129}	d_8	d_6	d_{511}	d_9	d_{187}	d_3	d_{48}	d_{38}	d_{25}	d_{113}	d_{250}
			●			●				●					
P(%)			33.3			33.3				30					

Single Value Summaries – Precision Histograms

- R-Precision can be used to compare two algorithms
  - A visual inspection
  - For each query, the difference of R-Precision for two algorithms (A and B) can be computed
    - $RP_A(i)$: R-precision for algorithm A for the i-th query
    - $RP_B(i)$: R-precision for algorithm B for the i-th query



$$RP_{A/B}(i) = RP_A(i) - RP_B(i)$$

Single Value Summaries – MAP.

- Precision@K and R-Precision give scores for queries individually
  - They are still hard to compare the performance between **systems**!

- Mean Average Precision (MAP)
  - The idea here is to average the precision figures obtained after each new relevant document is observed
    - Averaged at relevant documents and across queries
    - Widely used in IR performance evaluation

$$MAP = \frac{1}{|\mathbf{Q}|}\sum_{q \in \mathbf{Q}} MAP_q$$

Single Value Summaries – MAP..

$$MAP = \frac{1}{|\mathbf{Q}|} \sum_{q \in \mathbf{Q}} MAP_q$$

- For example (MAP):
  - the ranking model returns fifteen documents for each query
  - the first query has five relevant documents $\{d_{123}, d_{56}, d_9, d_{25}, d_3\}$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
	●		●			●				●					●
P(%)	100		66.6			50				40					33.3

  - the second query has three relevant documents $\{d_{123}, d_6, d_3\}$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	d_{84}	d_{56}	d_{123}	d_{129}	d_8	d_6	d_{511}	d_9	d_{187}	d_3	d_{48}	d_{38}	d_{25}	d_{113}	d_{250}
			●			●				●					
P(%)			33.3			33.3				30					

$$MAP = \frac{1}{2} \times \left(\frac{1.0 + 0.66 + 0.5 + 0.4 + 0.33}{5} + \frac{0.33 + 0.33 + 0.30}{3} \right) = 0.449$$

Average Precision

Single Value Summaries – MAP...

$$MAP = \frac{1}{|\mathbf{Q}|}\sum_{q\in\mathbf{Q}} MAP_q$$

- For example (MAP):
  - the ranking model returns eight documents for each query
  - the first query has five relevant documents $\{d_{123}, d_{56}, d_9, d_{25}, d_3\}$

	1	2	3	4	5	6	7	8
	d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}
	●		●			●		
P(%)	100		66.6			50		

  - the second query has three relevant documents $\{d_{123}, d_6, d_3\}$

	1	2	3	4	5	6	7	8
	d_{84}	d_{56}	d_{123}	d_{129}	d_8	d_6	d_{511}	d_9
			●			●		
P(%)			33.3			33.3		

$$MAP = \frac{1}{2} \times \left(\frac{1.0 + 0.66 + 0.5 + 0.0 + 0.0}{5} + \frac{0.33 + 0.33 + 0.00}{3}\right) = 0.326$$

Average Precision

Single Value Summaries – MRR

- Mean Reciprocal Rank is a good metric for those cases in which we are interested in the first correct answer
  - Question-Answering (QA) systems
  - Search engine queries that look for specific sites
    - URL queries
    - Homepage queries
  - It can be treated as a combination of **Precision@*K*** and ***R*-Precision**
    - The **position of the first relevant document** + a **position constrain**!

$$MRR_i(q) = \begin{cases} \frac{1}{rank} & , if\ the\ position\ of\ the\ first\ relevant\ document < i \\ 0 & , otherwise \end{cases}$$

$$MRR_i(\mathbf{Q}) = \frac{1}{|\mathbf{Q}|} \sum_{q \in \mathbf{Q}} MRR_i(q)$$

Single Value Summaries – MRR

$$MRR_5(\mathbf{Q}) = \frac{1}{3} \times \left(\frac{1}{1} + 0 + \frac{1}{3}\right) = \frac{4}{9}$$

- For the first query

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
d_{123}	d_{84}	d_{56}	d_6	d_8	d_9	d_{511}	d_{129}	d_{187}	d_{25}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
●		●			●				●					●

- For the second query

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
d_{511}	d_8	d_6	d_{56}	d_{84}	d_9	d_{123}	d_{25}	d_{129}	d_{187}	d_{38}	d_{48}	d_{250}	d_{113}	d_3
					●									●

- For the third query

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
d_{84}	d_{56}	d_{123}	d_{129}	d_8	d_6	d_{511}	d_9	d_{187}	d_3	d_{48}	d_{38}	d_{25}	d_{113}	d_{250}
		●			●				●					

Single Value Summaries – F-Measure

- F-Measure combines recall and precision
  - Harmonic Mean (調和平均)

$$F(i) = \frac{2}{\frac{1}{R(i)} + \frac{1}{P(i)}} = \frac{2 \times P(i) \times R(i)}{P(i) + R(i)}$$

  - $R(i)$ is the recall at the i-th position in the ranking
  - $P(i)$ is the precision at the i-th position in the ranking

- Properties
  - $0 \leq F(i) \leq 1$
  - $F(i) = 0$: no relevant documents were retrieved
  - $F(i) = 1$: all ranked documents are relevant
  - A high $F(i)$ achieved when both recall and precision are high

Single Value Summaries – E-Measure

- E-Measure combines recall and precision
  - It allows the user to specify whether he is more interested in recall or precision

$$E(i) = 1 - \frac{1 + b^2}{\frac{b^2}{R(i)} + \frac{1}{P(i)}} = 1 - \frac{(1 + b^2) \times P(i) \times R(i)}{b^2 \times P(i) + R(i)}$$

  - $E(i)$ is the *E-Measure* at the i-th position in the ranking
  - $R(i)$ is the recall at the i-th position in the ranking
  - $P(i)$ is the precision at the i-th position in the ranking
  - $b \geq 0$ is a user specified parameter
    - $b = 0 \;\Rightarrow\; E(i) = 1 - P(i)$
    - $b \to \infty \;\Rightarrow\; lim_{b \to \infty} E(i) = 1 - R(i)$
    - $b = 1 \;\Rightarrow\; E(i) = 1 - \underline{\frac{2 \times P(i) \times R(i)}{P(i) + R(i)}}$

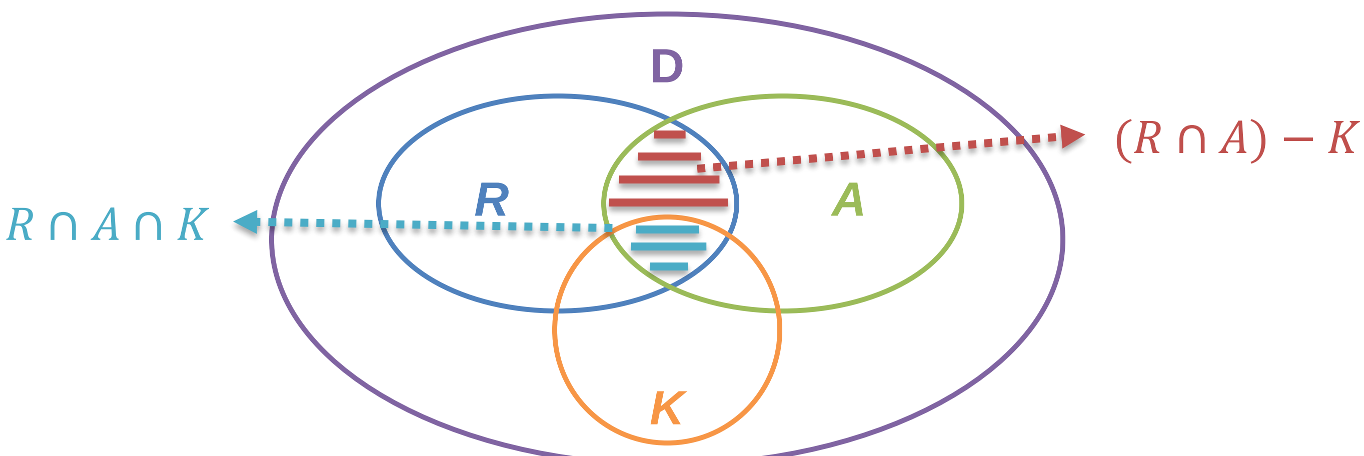
F-Measure

User-Oriented Measures

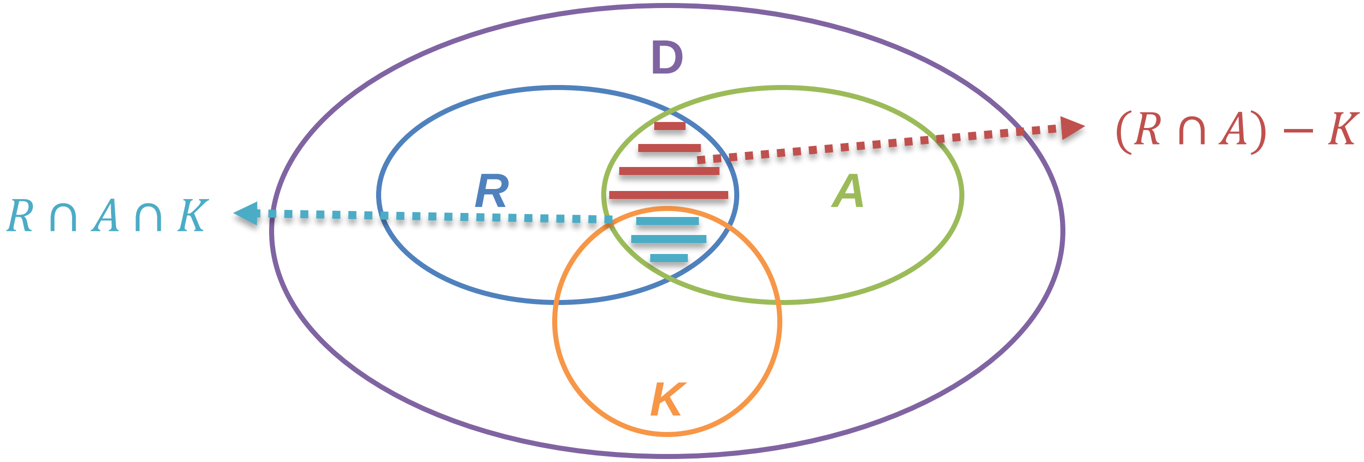
- Recall and precision assume that the set of relevant documents for a query is independent of the users
- However, different users might have different relevance interpretations
- User-oriented measures have been proposed
  - Coverage ratio
  - Novelty ratio
  - Relative recall
  - Recall effort

User-Oriented Measures – Notations

- For a given query (information need)
  - **D**: the set of documents
  - R: the set of relevant documents
  - A: the answer set generated by an IR system
  - K: the set of documents known to the user
    - $R \cap A \cap K$: the set of relevant documents that have been retrieved and are known to the user
    - $(R \cap A) - K$: the set of relevant documents that have been retrieved but are not known to the user



User-Oriented Measures.



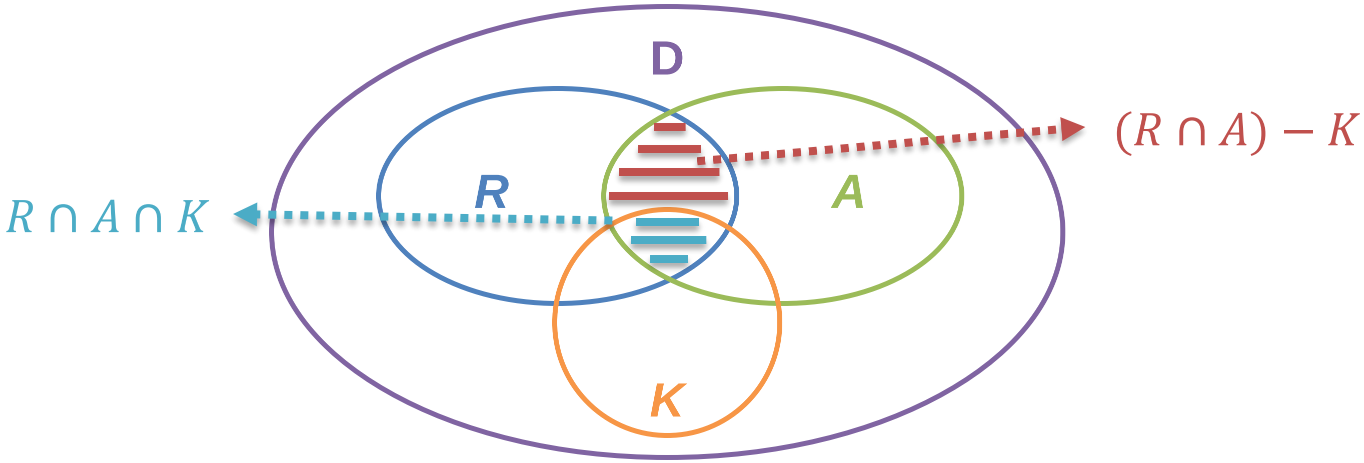
- The **coverage ratio** is the fraction of the documents known and relevant that are in the answer set

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

- The **novelty ratio** is the fraction of the relevant documents in the answer set that are not known to the user

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

User-Oriented Measures..



- The **relative recall** is the ratio between the number of relevant docs found by the system and the number of relevant documents known to the user

$$Relative\ Recall = \frac{|R \cap A|}{|R \cap K|}$$

- The **recall effort** is the ratio between the number of relevant documents known to the user and the number of documents found by the system

$$Recall\ Effort = \frac{|R \cap K|}{|A|}$$

Discounted Cumulated Gain (DCG)

- Precision and recall allow only binary relevance assessments
  - No distinction between highly relevant documents and mildly relevant documents

- These limitations can be overcome by adopting graded relevance assessments and metrics that combine them

- The **discounted cumulated gain** (DCG) is a metric that combines graded relevance assessments effectively
  - 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

DCG – 1

- Consider that the results of the queries are graded on a scale 0–3
  - 0 for non-relevant, 3 for strong relevant docs

- For instance
  - For queries q_1 and q_2, consider that the graded relevance scores are as follows:

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

$$R_{q_2} = \{[d_3, 3], [d_{56}, 2], [d_{129}, 1]\}$$

    - For query q_1, document d_3 is highly relevant and document d_{56} is just mildly relevant

DCG – 2

- For a ranking algorithm, top 15 documents are generated for both queries

$$A_{q_1} = \{d_{71}, d_2, d_{56}, d_3, d_4, d_9, d_{11}, d_{12}, d_{13}, d_{25}, d_{21}, d_{22}, d_{23}, d_{24}, d_5\}$$

$$A_{q_2} = \{d_{71}, d_2, d_{56}, d_5, d_4, d_9, d_{11}, d_{129}, d_{13}, d_{25}, d_{21}, d_{22}, d_{23}, d_{24}, d_3\}$$

- The **gain vectors** for the two queries are

$$G_{q_1} = \{1,0,1,0,0,3,0,0,0,2,0,0,0,0,3\}$$

$$G_{q_2} = \{0,0,2,0,0,0,0,1,0,0,0,0,0,0,3\}$$

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

$$R_{q_2} = \{[d_3, 3], [d_{56}, 2], [d_{129}, 1]\}$$

DCG – 3

- The **cumulated gain vectors** can then be obtained

$$CG\,[i] = \begin{cases} G[1] & ,if\ i = 1 \\ G[i] + CG[i-1] & ,otherwise \end{cases}$$

  - For the first query

$$G_{q_1} = \{1, 0, 1, 0, 0, 3, 0, 0, 0, 2, 0, 0, 0, 0, 3\}$$

$$CG_{q_1} = \{1, 1, 2, 2, 2, 5, 5, 5, 5, 7, 7, 7, 7, 7, 10\}$$

  - For the second query

$$G_{q_2} = \{0, 0, 2, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 3\}$$

$$CG_{q_2} = \{0, 0, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 6\}$$

DCG – 4

- Let's introduce a **discount factor** that reduces the impact of the gain as we move upper in the ranking
  - A simple discount factor is the logarithm of the ranking position
  - If we consider logs in base 2
    - For position 2, the discounting factor is $log_2 2$
    - For position 3, the discounting factor is $log_2 3$

- The **discounted cumulated gain vectors** can be obtained

$$DCG\ [i] = \begin{cases} G[1] & ,if\ i = 1 \\ \dfrac{G[i]}{log_2(i)} + DCG[i-1] & ,otherwise \end{cases}$$

DCG – 5.

$$DCG\ [i] = \begin{cases} G[1] & ,if\ i = 1 \\ \dfrac{G[i]}{log_2(i)} + DCG[i-1] & ,otherwise \end{cases}$$

– For the first query

$$G_{q_1} = \{1, 0, 1, 0, 0, 3, 0, 0, 0, 2, 0, 0, 0, 0, 3\}$$

$$DCG_{q_1} = \{1, 1, 1.6, 1.6, 1.6, 2.8, 2.8, 2.8, 2.8, 3.4, 3.4, 3.4, 3.4, 3.4, 4.2\}$$

– For the second query

$$G_{q_2} = \{0, 0, 2, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 3\}$$

$$DCG_{q_2} = \{0, 0, 1.3, 1.3, 1.3, 1.3, 1.3, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 2.4\}$$

DCG – 5..

$$DCG\ [i] = \begin{cases} G[1] & , if\ i = 1 \\ \dfrac{G[i]}{log_2(i)} + DCG[i-1] & , otherwise \end{cases}$$

– For the first query

$$G_{q_1} = \{1, 0, 1, 0, 0, 3, 0, 0, 0, 2, 0, 0, 0, 0, 3\}$$

$$\frac{0}{log_2 2} + 1$$

$$DCG_{q_1} = \{1, 1, 1.6, 1.6, 1.6, 2.8, 2.8, 2.8, 2.8, 3.4, 3.4, 3.4, 3.4, 3.4, 4.2\}$$

– For the second query

$$G_{q_2} = \{0, 0, 2, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 3\}$$

$$DCG_{q_2} = \{0, 0, 1.3, 1.3, 1.3, 1.3, 1.3, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 2.4\}$$

DCG – 5…

$$DCG\ [i] = \begin{cases} G[1] & , if\ i = 1 \\ \dfrac{G[i]}{log_2(i)} + DCG[i-1] & , otherwise \end{cases}$$

– For the first query

$$G_{q_1} = \{1, 0, 1, 0, 0, 3, 0, 0, 0, 2, 0, 0, 0, 0, 3\}$$

$$\frac{1}{log_2 3} + 1$$

$$DCG_{q_1} = \{1, 1, 1.6, 1.6, 1.6, 2.8, 2.8, 2.8, 2.8, 3.4, 3.4, 3.4, 3.4, 3.4, 4.2\}$$

– For the second query

$$G_{q_2} = \{0, 0, 2, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 3\}$$

$$DCG_{q_2} = \{0, 0, 1.3, 1.3, 1.3, 1.3, 1.3, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 2.4\}$$

DCG – 5….

$$DCG\,[i] = \begin{cases} G[1] & ,if\ i = 1 \\ \dfrac{G[i]}{log_2(i)} + DCG[i-1] & ,otherwise \end{cases}$$

– For the first query

$$G_{q_1} = \{1, 0, 1, 0, 0, 3, 0, 0, 0, 2, 0, 0, 0, 0, 3\}$$

$$\frac{0}{log_2 4} + 1.6$$

$$DCG_{q_1} = \{1, 1, 1.6, 1.6, 1.6, 2.8, 2.8, 2.8, 2.8, 3.4, 3.4, 3.4, 3.4, 3.4, 4.2\}$$

– For the second query

$$G_{q_2} = \{0, 0, 2, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 3\}$$

$$DCG_{q_2} = \{0, 0, 1.3, 1.3, 1.3, 1.3, 1.3, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 2.4\}$$

DCG – 5.....

$$DCG\ [i] = \begin{cases} G[1] & ,if\ i = 1 \\ \dfrac{G[i]}{log_2(i)} + DCG[i-1] & ,otherwise \end{cases}$$

– For the first query

$$G_{q_1} = \{1, 0, 1, 0, 0, 3, 0, 0, 0, 2, 0, 0, 0, 0, 3\}$$

$$\frac{3}{log_2 6} + 1.6$$

$$DCG_{q_1} = \{1, 1, 1.6, 1.6, 1.6, 2.8, 2.8, 2.8, 2.8, 3.4, 3.4, 3.4, 3.4, 3.4, 4.2\}$$

– For the second query

$$G_{q_2} = \{0, 0, 2, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 3\}$$

$$DCG_{q_2} = \{0, 0, 1.3, 1.3, 1.3, 1.3, 1.3, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 2.4\}$$

CG vs. DCG

- Discounted cumulated gains are much less affected by relevant documents at the end of the ranking

$$CG_{q_1} = \{1, 1, 2, 2, 2, 5, 5, 5, 5, 7, 7, 7, 7, 7, 10\}$$

$$DCG_{q_1} = \{1, 1, 1.6, 1.6, 1.6, 2.8, 2.8, 2.8, 2.8, 3.4, 3.4, 3.4, 3.4, 3.4, 4.2\}$$

$$CG_{q_2} = \{0, 0, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 6\}$$

$$DCG_{q_2} = \{0, 0, 1.3, 1.3, 1.3, 1.3, 1.3, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 2.4\}$$

CG & DCG Curves – 1

- To produce CG and DCG curves over a set of test queries, we need to average them over all queries

- Given a set of queries $\mathbf{Q}$, average $\overline{CG}[i]$ and $\overline{DCG}[i]$ over all queries are computed as follows

$$\overline{CG}[i] = \sum_{q \in \mathbf{Q}} \frac{CG_q[i]}{|\mathbf{Q}|}$$

$$CG_{q_1} = \{1, 1, 2, 2, 2, 5, 5, 5, 5, 7, 7, 7, 7, 7, 10\}$$

$$CG_{q_2} = \{0, 0, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 6\}$$

$$\overline{CG} = \{0.5, 0.5, 2.0, 2.0, 2.0, 3.5, 3.5, 4.0, 4.0, 5.0, 5.0, 5.0, 5.0, 5.0, 8.0\}$$

$$\overline{DCG}[i] = \sum_{q \in \mathbf{Q}} \frac{DCG_q[i]}{|\mathbf{Q}|}$$

$$DCG_{q_1} = \{1, 1, 1.6, 1.6, 1.6, 2.8, 2.8, 2.8, 2.8, 3.4, 3.4, 3.4, 3.4, 3.4, 4.2\}$$

$$DCG_{q_2} = \{0, 0, 1.3, 1.3, 1.3, 1.3, 1.3, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 1.6, 2.4\}$$

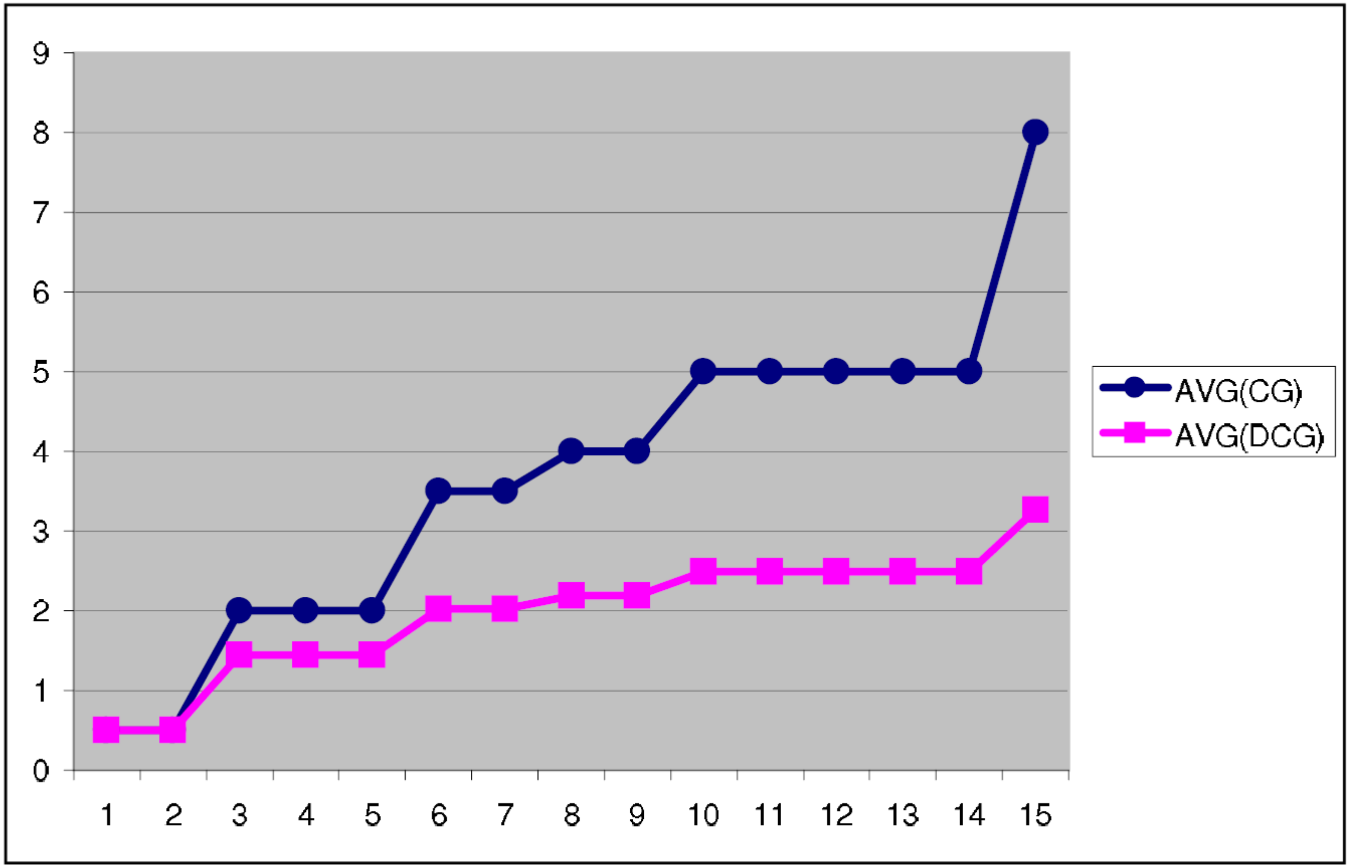
$$\overline{DCG} = \{0.5, 0.5, 1.5, 1.5, 1.5, 2.1, 2.1, 2.2, 2.2, 2.5, 2.5, 2.5, 2.5, 2.5, 3.3\}$$

CG & DCG Curves – 2

- Average curves can then be drawn by varying the rank positions from 1 to a pre-established threshold

$$\overline{CG} = \{0.5, 0.5, 2.0, 2.0, 2.0, 3.5, 3.5, 4.0, 4.0, 5.0, 5.0, 5.0, 5.0, 5.0, 8.0\}$$

$$\overline{DCG} = \{0.5, 0.5, 1.5, 1.5, 1.5, 2.1, 2.1, 2.2, 2.2, 2.5, 2.5, 2.5, 2.5, 2.5, 3.3\}$$



Ideal G & CG & DCG – 1

- Since the relevant documents with their graded score for queries q_1 and q_2 are:

$$R_{q_1} = \{[d_3, 3], [d_5, 3], [d_9, 3], [d_{25}, 2], [d_{39}, 2], [d_{44}, 2], [d_{56}, 1], [d_{71}, 1], [d_{89}, 1], [d_{123}, 1]\}$$
$$R_{q_2} = \{[d_3, 3], [d_{56}, 2], [d_{129}, 1]\}$$

- The ideal gain vectors are:

$$IG_{q_1} = \{3,3,3,2,2,2,1,1,1,1,0,0,0,0,0\}$$
$$IG_{q_2} = \{3,2,1,0,0,0,0,0,0,0,0,0,0,0,0\}$$

- The ideal cumulated gain vectors

$$ICG_{q_1} = \{3,6,9,11,13,15,16,17,18,19,19,19,19,19,19\}$$
$$ICG_{q_2} = \{3,5,6,6,6,6,6,6,6,6,6,6,6,6,6\}$$

Ideal G & CG & DCG – 2

- Consequently, the ideal discounted cumulated gain vectors

$$IDCG_{q_1} = \{3.0, 6.0\ , 7.9, 8.9, 9.8, 10.5, 10.9, 11.2, 11.5, 11.8, 11.8, 11.8, 11.8, 11.8, 11.8\}$$

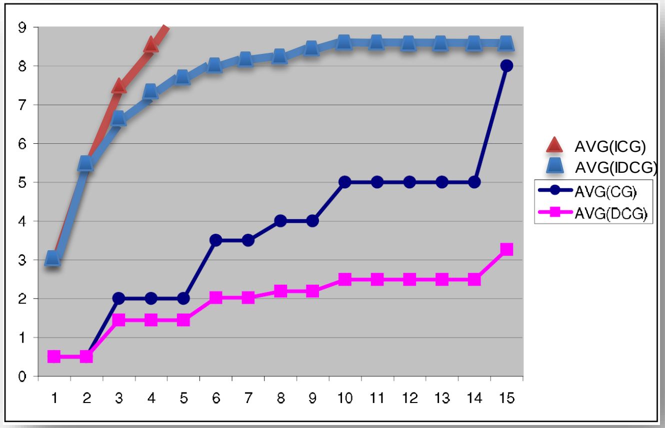
$$IDCG_{q_2} = \{3.0, 5.0, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6, 5.6\}$$

- Further, the average $\overline{ICG}[i]$ and $\overline{IDCG}[i]$ can also be obtained

$$\overline{ICG} = \{3.0, 5.5, 7.5, 8.5, 9.5, 10.5, 11.0, 11.5, 12.0, 12.5, 12.5, 12.5, 12.5, 12.5, 12.5\}$$

$$\overline{IDCG} = \{3.0, 5.5, 6.8, 7.3, 7.7, 8.1, 8.3, 8.4, 8.6, 8.7, 8.7, 8.7, 8.7, 8.7, 8.7\}$$

- By comparing the average CG and DCG curves for an algorithm with the average ideal curves, we gain insight on how much room for improvement there is



Normalized CG & DCG – 1

- Given a set of queries, the normalized CG and DCG can be computed by:

$$NCG[i] = \frac{\overline{CG}[i]}{\overline{ICG}[i]} \qquad NDCG[i] = \frac{\overline{DCG}[i]}{\overline{IDCG}[i]}$$

- In our example, the NCG and NDCG vectors are:

$$\overline{CG} = \{0.5, 0.5, 2.0, 2.0, 2.0, 3.5, 3.5, 4.0, 4.0, 5.0, 5.0, 5.0, 5.0, 5.0, 8.0\}$$

$$\overline{ICG} = \{3.0, 5.5, 7.5, 8.5, 9.5, 10.5, 11.0, 11.5, 12.0, 12.5, 12.5, 12.5, 12.5, 12.5, 12.5\}$$

$$NCG = \{0.17, 0.09, 0.27, 0.24, 0.21, 0.33, 0.32, 0.35, 0.33, 0.40, 0.40, 0.40, 0.40, 0.40, 0.64\}$$

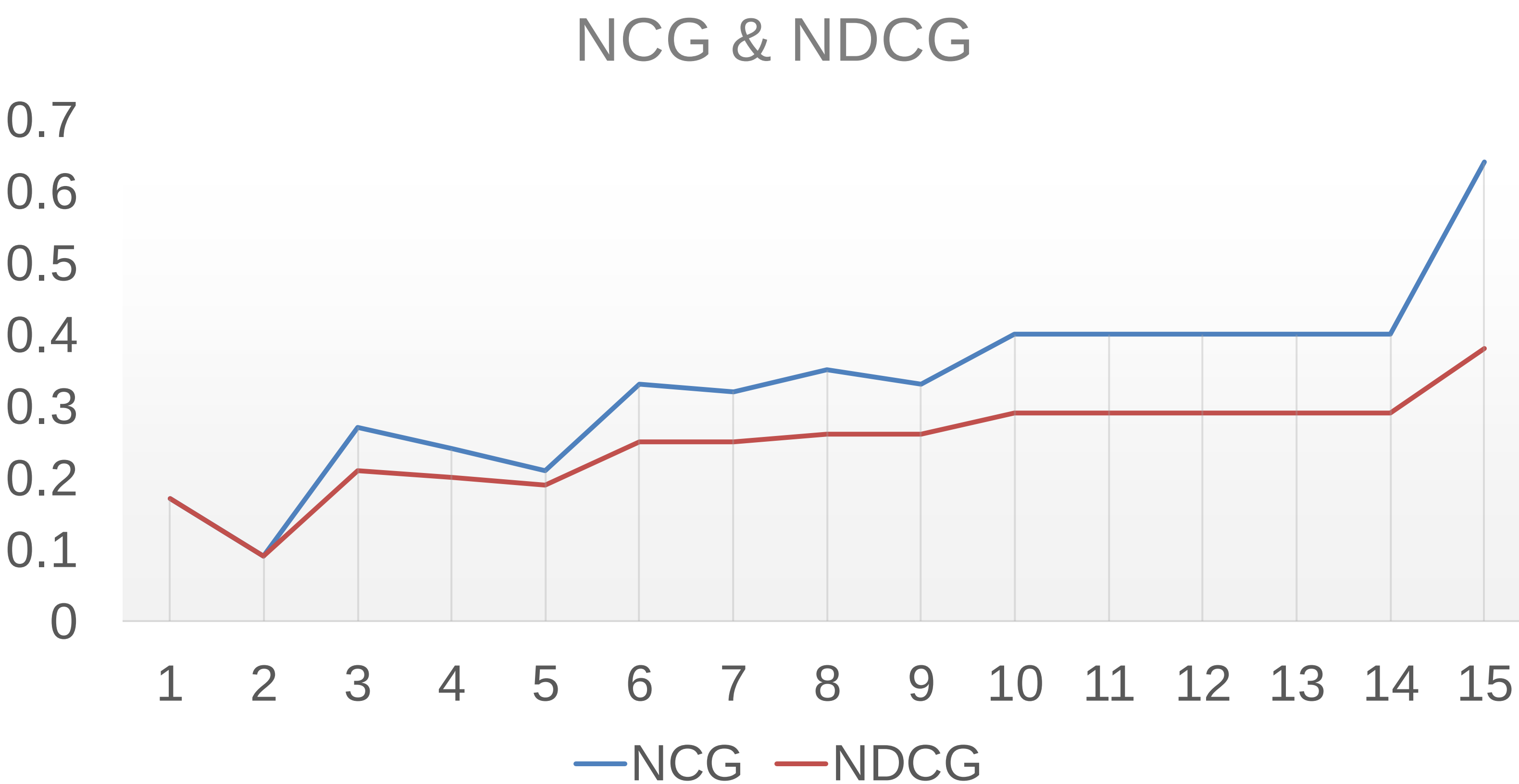
$$\overline{DCG} = \{0.5, 0.5, 1.5, 1.5, 1.5, 2.1, 2.1, 2.2, 2.2, 2.5, 2.5, 2.5, 2.5, 2.5, 3.3\}$$

$$\overline{IDCG} = \{3.0, 5.5, 6.8, 7.3, 7.7, 8.1, 8.3, 8.4, 8.6, 8.7, 8.7, 8.7, 8.7, 8.7, 8.7\}$$

$$NDCG = \{0.17, 0.09, 0.21, 0.20, 0.19, 0.25, 0.25, 0.26, 0.26, 0.29, 0.29, 0.29, 0.29, 0.29, 0.38\}$$

Normalized CG & DCG – 2

- The area under the NCG and NDCG curves represent the quality of the ranking algorithm
  - Larger the area, better the results



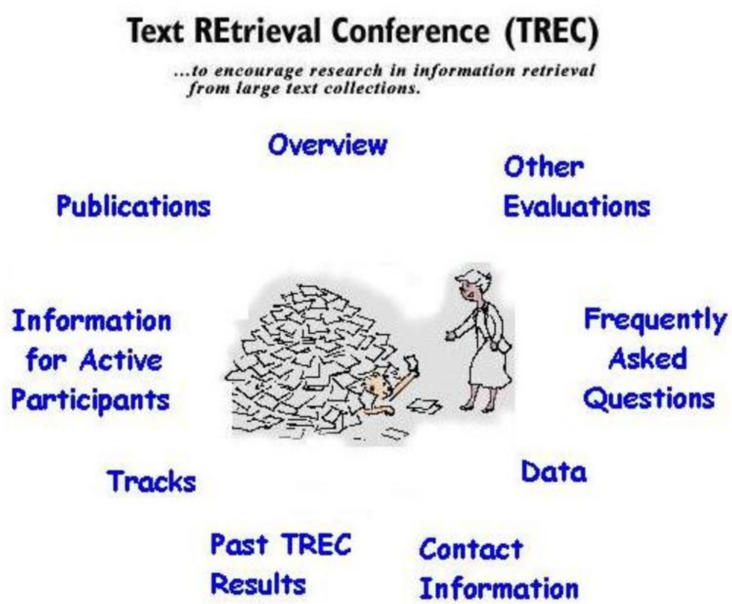
Pros & Cons for NDCG

- Advantages
  - CG and DCG metrics aim at taking into account multiple level relevance assessments
    - It can distinguish highly relevant documents from mildly relevant ones
  - Discounted cumulated gain allows down weighting the impact of relevant documents found late in the ranking

- Disadvantages
  - The relevance assessments are harder and more time consuming to generate

The TREC Collection

- **T**ext **RE**trieval **C**onference (TREC)
  - Established in 1991, co-sponsored by the National Institute of Standards and Technology (NIST, 美國國家標準技術研究所) and the Defense Advanced Research Projects Agency (DARPA, 國防高等研究計劃署)
  - Evaluation of large scale IR problems
  - The premier annual conference was held at NIST in Nov. 1992



http://trec.nist.gov/

The Goal of TREC

- To encourage **research in information retrieval** based on large test collections
- To increase **communication among industry, academia, and government** by creating an open forum for the exchange of research ideas
- To speed the **transfer of technology from research labs into commercial products**
- To increase the availability of **appropriate evaluation techniques** for use by industry and academia

TREC Collection

- A TREC collection is composed of three parts:
  - the documents
  - the example information requests (called **topics**)
  - a set of relevant documents for each example information request

- The main TREC collection has been growing steadily over the years
  - The TREC-3 collection has roughly 2 gigabytes
  - The TREC-6 collection has roughly 5.8 gigabytes
  - The TREC-15 collection has roughly 426 gigabytes
    - 25 million (25,000,000) Web documents

TREC Document

- An example of a TREC document

```
<doc>

<docno> WSJ880406-0090 </docno>
<hl> AT&T Unveils Services to Upgrade Phone Networks
Under Global Plan </hl>
<author> Janet Guyon (WSJ Staff) </author>
<dateline> New York </dateline>

<text>
American Telephone & Telegraph Co introduced the first
of a new generation of phone services with broad ...
</text>

</doc>
```

TREC Topic

- An example of an information request is the topic numbered 168 used in TREC-3

```
<top>

<num> Number:  168
<title> Topic:  Financing AMTRAK

<desc> Description:
A document will address the role of the Federal Government in
financing the operation of the National Railroad Transportation
Corporation (AMTRAK)
<narr> Narrative:  A relevant document must provide information on
the government's responsibility to make AMTRAK an economically viable
entity.  It could also discuss the privatization of AMTRAK as an
alternative to continuing government subsidies.  Documents comparing
government subsidies given to air and bus transportation with those
provided to AMTRAK would also be relevant

</top>
```

taken as a short query

taken as a long query

describe the criteria for relevance, used by the people doing relevance judgments, and not taken as a query

TREC Judgments – Pooling Method

- The set of relevant documents for each topic is obtained from a pool of possible relevant documents
  - This pool is created by taking the top *K* documents (usually, *K*=100) in the rankings generated by various retrieval systems

- The documents in the pool are then shown to human assessors who ultimately decide on the relevance of each document

- This technique of assessing relevance is called the **pooling method** and is based on two assumptions:
  - Vast majority of relevant documents is collected in the assembled pool
  - Documents not in the pool were considered to be irrelevant

Popular Collections

- TREC: http://trec.nist.gov/
- CLEF: http://www.clef-initiative.eu
- NTCIR: http://research.nii.ac.jp/ntcir/index-en.html
- FIRE: http://fire.irsi.res.in/fire/static/resources
- Note that these web sites host the publications, current meeting information, and also where to get the test collections for use outside of the evaluations

Questions?

kychen@mail.ntust.edu.tw