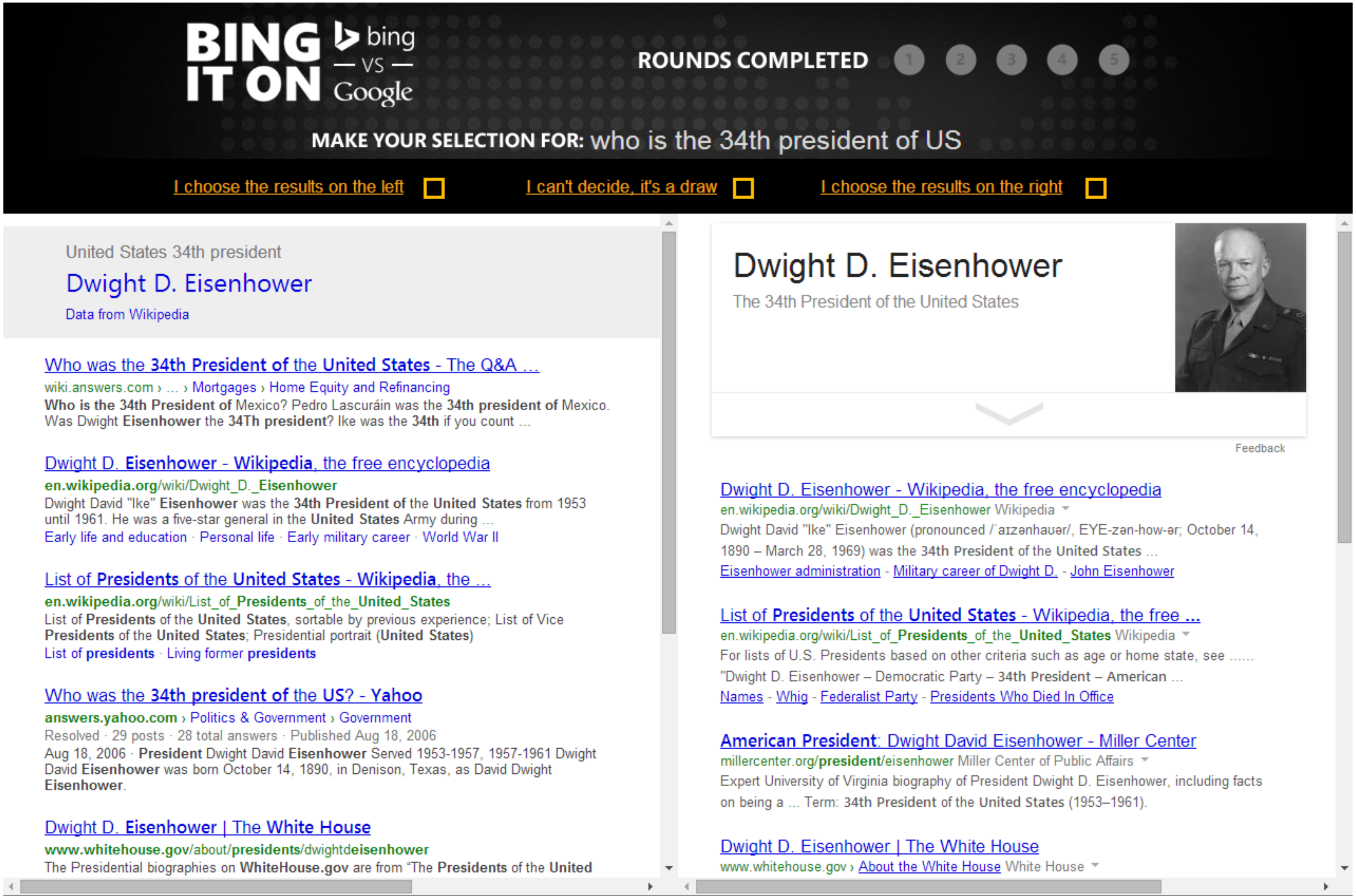
Retrieval Evaluation

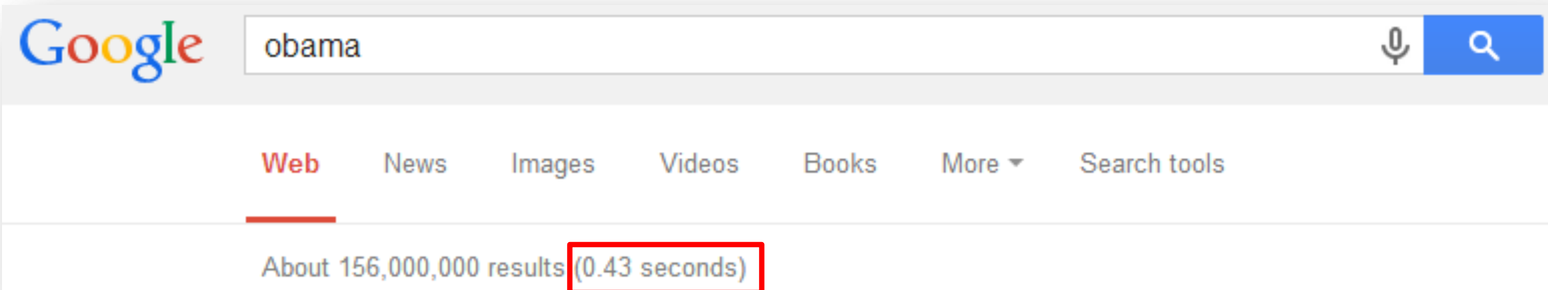
Slides borrowed from Hongning Wang with modifications

Bing v.s. Google?

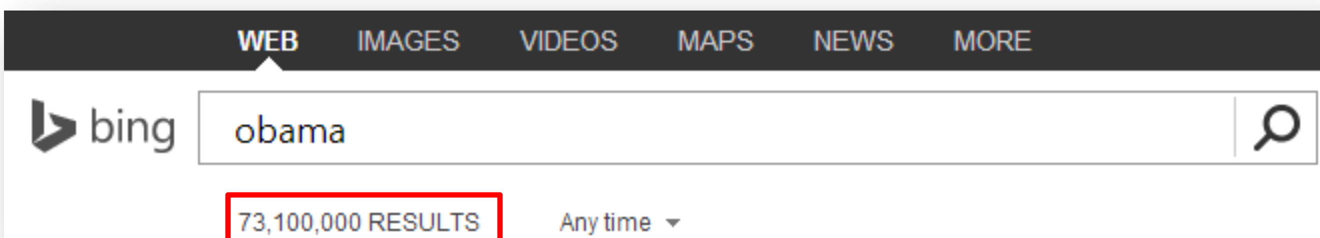


Which search engine do you prefer: Bing or Google?

- What are your judging criteria?
  - How fast does it response to your query?

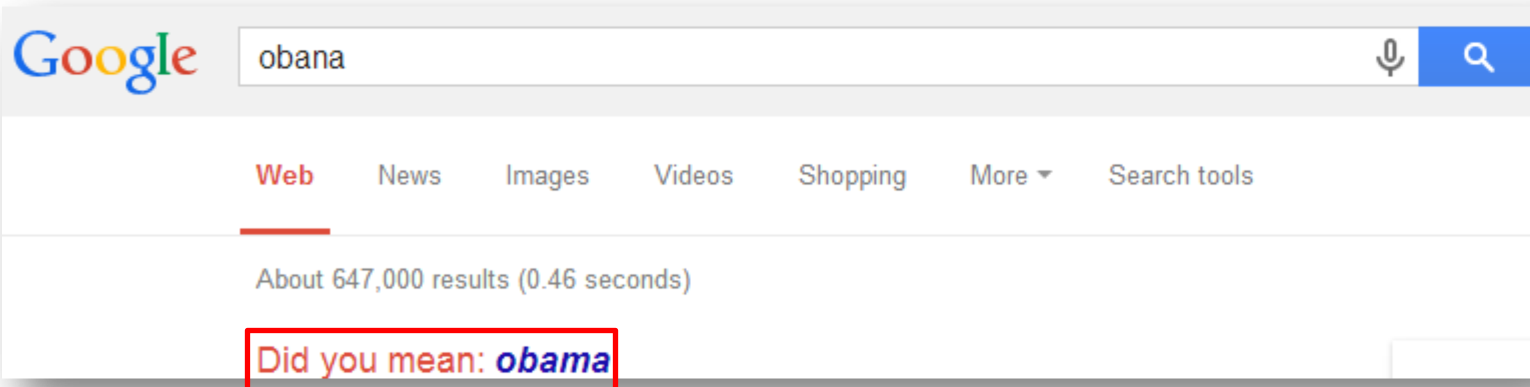


  - How many documents can it return?



Which search engine do you prefer: Bing or Google?

- What are your judging criteria?
  - Can it correct my spelling errors?



  - Can it suggest me good related queries?

Retrieval evaluation

- Aforementioned evaluation criteria are all good, but not essential
  - Goal of any IR system
    - Satisfying users' information need
  - Core quality measure criterion
    - *"how well a system meets the information needs of its users." – wiki*
    - Unfortunately vague and hard to execute

Quantify the IR quality measure

- Information need
  - *“an individual or group's desire to locate and obtain information to satisfy a conscious or unconscious need” – wiki*
  - Reflected by user <u>query</u>
  - Categorization of information need
    - Navigational
    - Informational
    - Transactional

Quantify the IR quality measure

- Satisfaction
  - *"the opinion of the user about a specific computer application, which they use" – wiki*
  - Reflected by
    - Increased result clicks
    - Repeated/increased visits
    - <u>Result relevance</u>

Classical IR evaluation

- Cranfield experiments
  - Pioneer work and foundation in IR evaluation
  - Basic hypothesis
    - Retrieved documents' relevance is a good proxy of a system's utility in satisfying users' information need
  - Procedure
    - 1,398 abstracts of aerodynamics journal articles
    - 225 queries
    - Exhaustive relevance judgments of all (query, document) pairs
    - Compare different indexing system over such collection

Classical IR evaluation

- Three key elements for IR evaluation
  1. A document collection
  2. A test suite of information needs, expressible as queries
  3. A set of relevance judgments, e.g., binary assessment of either *relevant* or *nonrelevant* for each query-document pair

Search relevance

- Users' information needs are translated into queries
- Relevance is judged with respect to the information need, **not** the query
  - E.g., Information need: "When should I renew my Texas driver's license?"

    Query: "Texas driver's license renewal"

    Judgment: whether a document contains the right answer, e.g., every 6 years; rather than if it literally contains those four words

Text REtrieval Conference (TREC)

- Large-scale evaluation of text retrieval methodologies
  - Since 1992, hosted by NIST
  - Standard benchmark for IR studies
  - A wide variety of evaluation collections
    - Web track
    - Question answering track
    - Cross-language track
    - Microblog track
    - And more…

Public benchmarks

TABLE 4.3 Common Test Corpora

Collection	*NDocs*	*NQrys*	*Size (MB)*	*Term/Doc*	*Q-D RelAss*
ADI	82	35			
AIT	2109	14	2	400	>10,000
CACM	3204	64	2	24.5	
CISI	1460	112	2	46.5	
Cranfield	1400	225	2	53.1	
LISA	5872	35	3		
Medline	1033	30	1		
NPL	11,429	93	3		
OSHMED	34,8566	106	400	250	16,140
Reuters	21,578	672	28	131	
TREC	740,000	200	2000	89-3543	» 100,000

Table from Manning Stanford CS276, Lecture 8

Evaluation metric

- To answer the questions
  - Is Google better than Bing?
  - Which ranking method is the most effective?
  - Shall we perform stemming or stopword removal?
- We need a quantifiable metric, by which we can compare different IR systems
  - As unranked retrieval sets
  - As ranked retrieval results

Evaluation of unranked retrieval sets

- In a Boolean retrieval system
  - Precision: fraction of retrieved documents that are relevant, i.e., p(relevant|retrieved)
  - Recall: fraction of relevant documents that are retrieved, i.e., p(retrieved|relevant)

	relevant	nonrelevant
retrieved	true positive (TP)	false positive (FP)
not retrieved	false negative (FN)	true negative (TN)

Precision:
$$P = \frac{TP}{TP + FP}$$

Recall: $R = \frac{TP}{TP + FN}$

Evaluation of unranked retrieval sets

- Precision and recall trade off against each other
  - Precision decreases as the number of retrieved documents increases (unless in perfect ranking), while recall keeps increasing
  - These two metrics emphasize different perspectives of an IR system
    - Precision: prefers systems retrieving fewer documents, but highly relevant
    - Recall: prefers systems retrieving more documents

Evaluation of unranked retrieval sets

- Summarizing precision and recall to a single value
  - In order to compare different systems
  - F-measure: weighted harmonic mean of precision and recall, α balances the trade-off

$$F = \frac{1}{\alpha \frac{1}{P} + (1-\alpha)\frac{1}{R}} \qquad \left(F_1 = \frac{2}{\frac{1}{P} + \frac{1}{R}} \right)$$

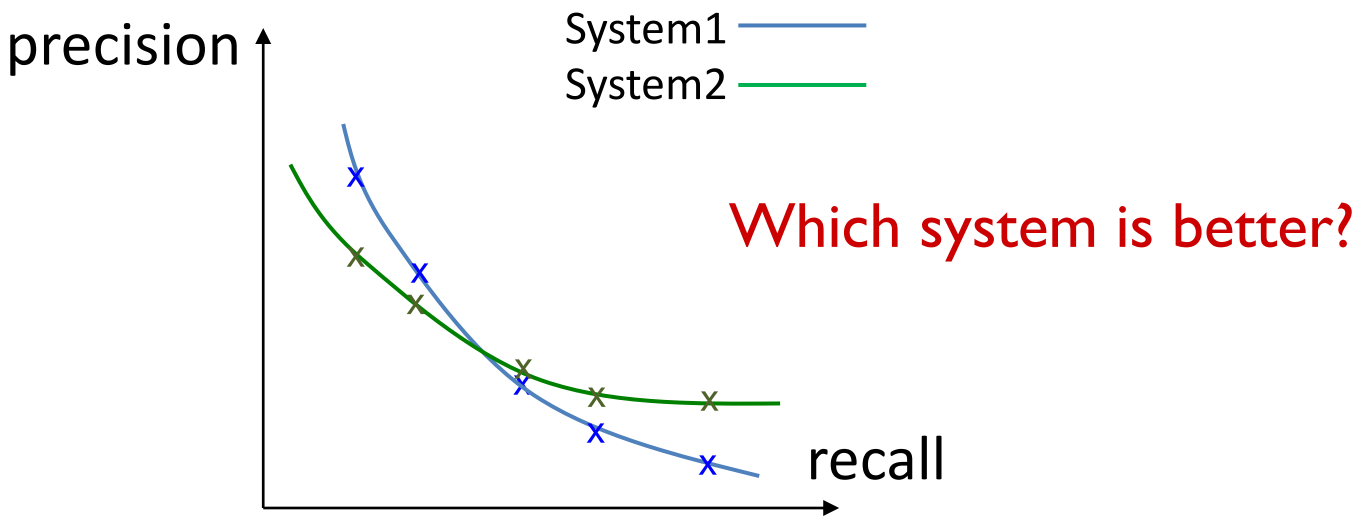
Equal weight between precision and recall

  - Why harmonic mean?
    - System1: P:0.53, R:0.36
    - System2: P:0.01, R:0.99

H	A
0.429	0.445
0.019	0.500

Evaluation of ranked retrieval results

- Ranked results are the core feature of an IR system
  - Precision, recall and F-measure are set-based measures, that cannot assess the ranking quality
  - Solution: evaluate precision at every recall point



Evaluation of ranked retrieval results

- Summarize the ranking performance with a single number
  - Binary relevance
    - Precision@K (P@K)
    - Mean Average Precision (MAP)
    - Mean Reciprocal Rank (MRR)
  - Multiple grades of relevance
    - Normalized Discounted Cumulative Gain (NDCG)

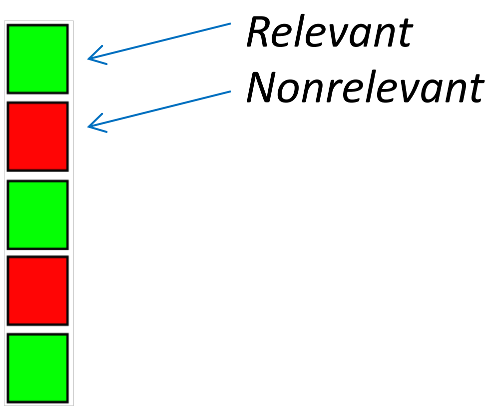
Precision@K

- Set a ranking position threshold K
- Ignores all documents ranked lower than K
- Compute precision in these top K retrieved documents
  - E.g.,

    P@3 of 2/3

    P@4 of 2/4

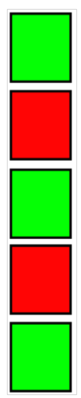
    P@5 of 3/5



- In a similar fashion we have Recall@K

Mean Average Precision

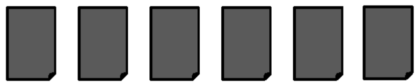
- Consider rank position of each <u>relevant</u> doc
  - E.g.,K_1, K_2, … K_R
- Compute P@K for each K_1, K_2, … K_R
- Average precision = average of those P@K
  - E.g.,



$$AvgPrec = \left(\frac{1}{1} + \frac{2}{3} + \frac{3}{5}\right)/3$$

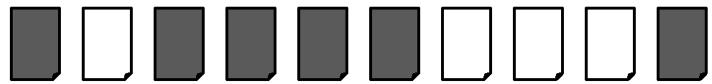
- MAP is the mean of Average Precision across multiple queries/rankings

AvgPrec is about one query



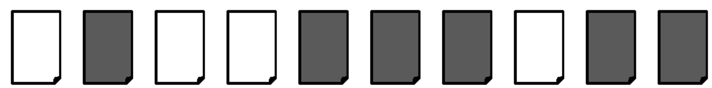
= the relevant documents

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

Figure from Manning Stanford CS276, Lecture 8

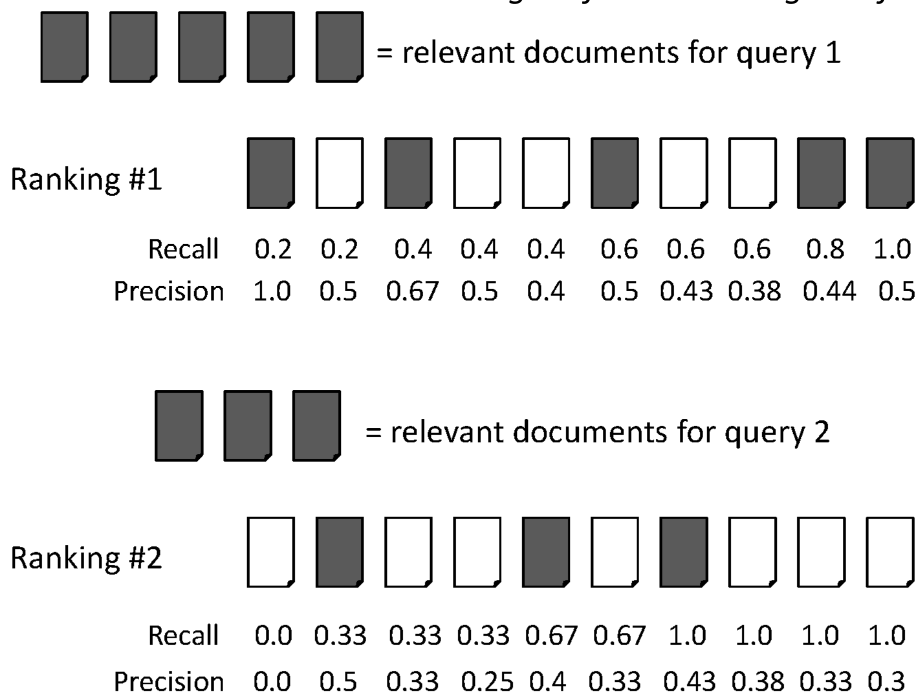
AvgPrec of the two rankings

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

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

MAP is about a system

Figure from Manning Stanford CS276, Lecture 8



Query 1, AvgPrec=(1.0+0.67+0.5+0.44+0.5)/5=0.62
Query 2, AvgPrec=(0.5+0.4+0.43)/3=0.44
MAP = (0.62+0.44)/2=0.53

MAP metric

- If a relevant document never gets retrieved, we assume the precision corresponding to that relevant document to be zero
- MAP is macro-averaging: each query counts equally
- MAP assumes users are interested in finding many relevant documents for each query
- MAP requires many relevance judgments in a text collection

Mean Reciprocal Rank

- Measure the effectiveness of the ranked results
  - Suppose users are only looking for one relevant document
    - looking for a fact
    - known-item search
    - navigational queries
    - query auto completion
- Search duration $\sim$ Rank of the answer
  - Measures a user's effort

Mean Reciprocal Rank

- Consider the rank position, K, of the first relevant document
- Reciprocal Rank = $\frac{1}{K}$
- MRR is the mean RR across multiple queries

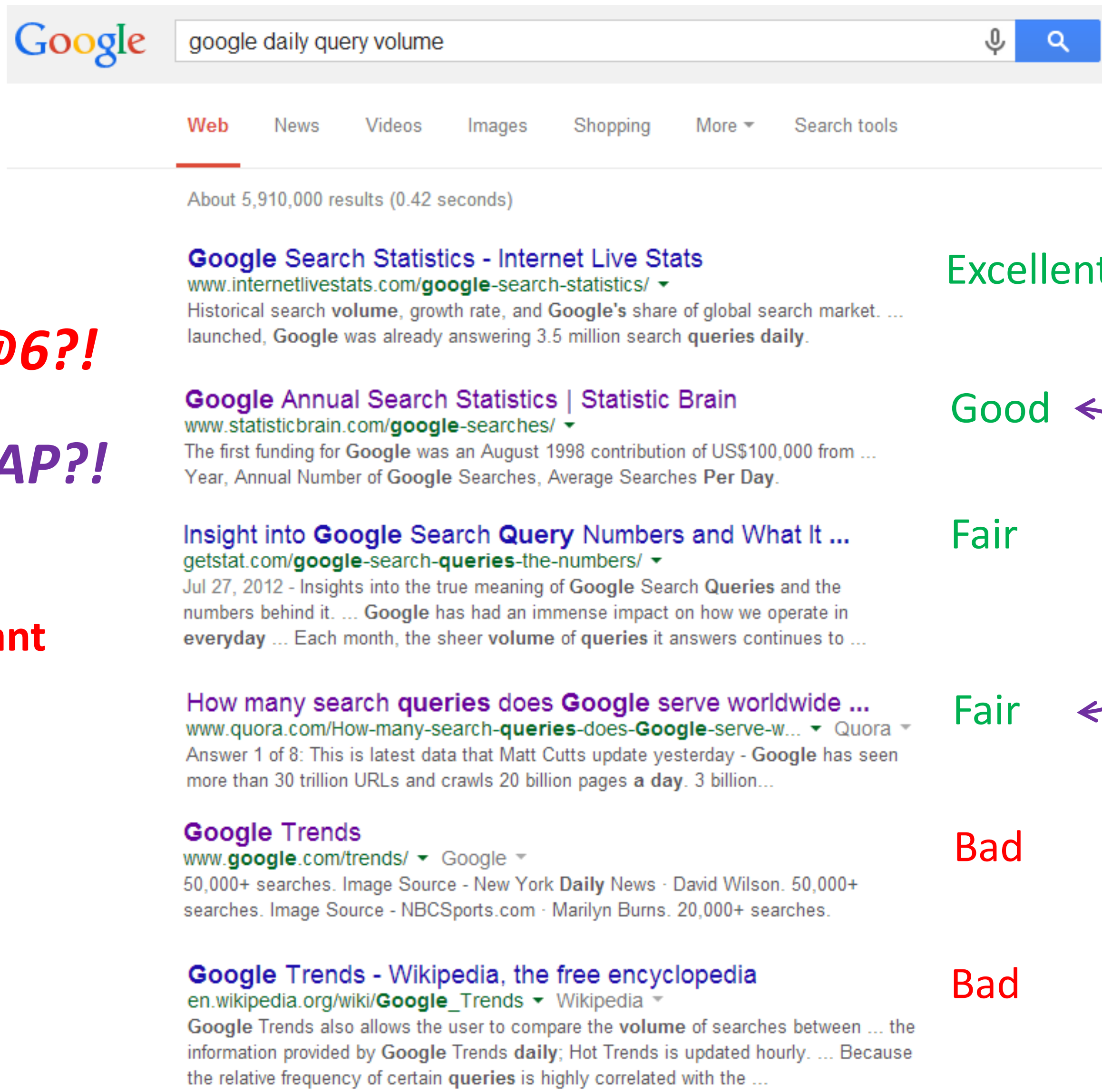
Beyond binary relevance

Same P@6?!

Same MAP?!

Relevant

Nonrelevant



Excellent

Good

Fair

Fair

Bad

Bad

Beyond binary relevance

- The level of documents' relevance quality with respect to a given query varies
  - Highly relevant documents are more useful than marginally relevant documents
  - The lower the ranked position of a relevant document is, the less useful it is for the user, since it is less likely to be examined
  - ***Discounted Cumulative Gain***

Discounted Cumulative Gain

- Uses graded relevance as a measure of usefulness, or gain, from examining a document
- Gain is accumulated starting at the top of the ranking and discounted at lower ranks
- Typical discount is 1/log (rank)
  – With base 2, the discount at rank 4 is 1/2, and at rank 8 it is 1/3

Discounted Cumulative Gain

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

Relevance label at position *i*

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

- Alternative formulation

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2(1+i)}$$

  – Standard metric in some web search companies
  – Emphasize on retrieving highly relevant documents

Normalized Discounted Cumulative Gain

- Normalization is useful for contrasting queries with varying numbers of relevant results
- Normalize DCG at rank n by the DCG value at rank n of the ideal ranking
  - The ideal ranking is achieved via ranking documents with their relevance labels

How about P@4, P@5, MAP and MRR?

NDCG - Example

5 documents: d_1, d_2, d_3, d_4, d_5

i	Ground Truth		Ranking Function$_1$		Ranking Function$_2$	
	Document Order	rel_i	Document Order	rel_i	Document Order	rel_i
1	d5	4	d3	2	d5	4
2	d4	3	d4	3	d3	2
3	d3	2	d2	1	d4	3
4	d2	1	d5	4	d1	0
5	d1	0	d1	0	d2	1

$$DCG_{GT} = \frac{2^4-1}{\log_2 2} + \frac{2^3-1}{\log_2 3} + \frac{2^2-1}{\log_2 4} + \frac{2^1-1}{\log_2 5} + \frac{2^0-1}{\log_2 6} = 21.35$$

$$DCG_{RF1} = \frac{2^2-1}{\log_2 2} + \frac{2^3-1}{\log_2 3} + \frac{2^1-1}{\log_2 4} + \frac{2^4-1}{\log_2 5} + \frac{2^0-1}{\log_2 6} = 14.38$$

$$DCG_{RF2} = \frac{2^4-1}{\log_2 2} + \frac{2^2-1}{\log_2 3} + \frac{2^3-1}{\log_2 4} + \frac{2^0-1}{\log_2 5} + \frac{2^1-1}{\log_2 6} = 20.78$$

Where do we get the relevance labels?

- Human annotation
  - Domain experts, who have better understanding of retrieval tasks
    - Scenario 1: annotator lists the information needs, formalizes into queries, and judges the returned documents
    - Scenario 2: given query and associated documents, annotator judges the relevance by inferring the underlying information need

Prepare annotation collection

- Human annotation is expensive and time consuming
  - Cannot afford exhaustive annotation of large corpus
  - Solution: pooling
    - Relevance is assessed over a subset of the collection that is formed from the top *k* documents returned by a number of different IR systems

Does pooling work?

- Judgments cannot possibly be exhaustive?
  - Relative rankings among the systems remain the same
- What about documents beyond top *k*?
  - Relative rankings among the systems remain the same

Rethink retrieval evaluation

- Goal of any IR system
  - Satisfying users' <u>information need</u>
- Core <u>quality</u> measure criterion
  - *"how well a system meets the information needs of its users." – wiki*

Challenge the assumptions in classical IR evaluations

- Assumption 1
  - Satisfaction = Result Relevance
- Assumption 2
  - Relevance = independent topical relevance
    - Documents are independently judged, and then ranked (that is how we get the ideal ranking)
- Assumption 3
  - Sequential browsing from top to bottom

What we have not considered

- The physical form of the output
  - User interface
- The effort, intellectual or physical, demanded of the user
  - User effort when using the system

What you should know

- Core criterion for IR evaluation
- Basic components in IR evaluation
- Classical IR metrics
- Pooling for preparing annotation collection

Today's reading

- Introduction to information retrieval
  - Chapter 8: Evaluation in information retrieval