IS2140 Information Storage and Retrieval

Unit 7: Relevance Feedback

Daqing He
School of Computing and Information
University of Pittsburgh

February 26, 2018

Classes in Next Several Weeks

- Oct 16: relevance feedback and query expansion
  - Assignment 4 Out
- Oct 22: user interaction and interactive information retrieval
  - Project Initial Presentation (Online)
- Oct 29: Web information retrieval
  - Assignment 3 Due
- Nov 5: Exam
- Nov 12: TA Session (Review assignments 1-3)
- Nov 19: Intelligent information retrieval
  - Assignment 4 Due
- Nov 26: Classification and Clustering

Term Project Proposal Presentation

- Each group create a presentation with 5 slides to highlight the project
  - Project name and team members
  - What is the search problem in the project?
  - What are the major inputs/outputs of the system?
  - What is your approach and plan?
    - Major steps of your approach
    - Current status, schedule until the end of the term
- The deadline is Oct 22
- Upload to the courseweb

Muddiest Points

- Language Model
  - On a multi-gigabyte collection, the frequencies of terms could range from 1 to billions. Do any of the calculations introduce problems when computing large ranges of values?
  - it was discussed that D(M_q || M_d) is preferred over D(M_d || M_q), but it sounded like the reasons for that are mostly historical. What are the technical or model trade-offs in preferring the first divergence over the second?
  - both Language Modeling and Vector Space Model are used for independent terms. Then I want to ask how to deal with dependent terms in documents. Should we always ignore dependent terms in a document like phrases? But what about the situation when processing social media documents, such as tags in which terms are almost dependent to each other?

Muddiest Points

- Smoothing
  - Based on my knowledge and understanding of this lecture, smoothing is a method to deal with the overfitting issue (try not so hard to fit the data) and the main idea of it is to give a relatively slow changes of the value. Like one of the examples on the lecture, we want to remove the steps of the curve so that to cut down the probabilities generated from the Maximum Likelihood Estimation (MLE). However, since we know that MLE is prone to overfitting, so we have other methods to deal with this issue in terms of the MLE, like we can add in penalty terms that punish parameters that are likely to be overfitted, or we can also stop the optimization process before the maximum is reached. So my question is: Can we use these methods in stead of smoothing? Or do we use the smoothing here for some other specific reasons?

Muddiest Points

- Evaluation
  - We say that in IR, most evaluations are comparative evaluation. However, measures like precision, recall and KAPPA score is actually objective evaluation since we can measure a single algorithm without comparing to the others.
  - When we focus on evaluating retrieval algorithms, do we need to consider the time consumption? If one retrieval algorithm for searching queries takes a long time but with a high accuracy , is it a good algorithm?

Muddiest Points

- Relevance Judgments and Ground truth
  - Can you explain the definition of ground truth?
  - As for relevance judgments, the standard approach is to mark the results as related or unrelated binary assessments. But the actual search results usually cannot be simply divided into two categories, they might be partially relevant to a query and have different levels of relevant. How does the IR evaluation deal with this problem.
  - Is there any possibilities that the ground truth judgments in the test collections become unreliable due to scientific progress or human error?
  - In slide 44, how to use judgments to evaluate a system that didn't participate in the evaluation?
  - Who made the ground truth in Cranfield methodology? what's the ground truth for a query?

Muddiest Points

- Kappa
  - How is Kappa measure used? I understand how to calculate it and what it represents but I'm not sure what to do with the value once I've calculated it. Do I simply exclude judges or query evaluations with large Kappa, or is there a way to incorporate this into the metrics of a system?
  - When we are in the situation of "tentative conclusion", which judge do we trust? Or we should find the third judge to restart the agreement process.
  - Does this measure also apply to large amount of judges?
  - Depending on different search purpose, does the size of test collections need to be changed? Since different evaluation methods have their own ways to choose test collections but the size is not specified. Is their a general standard for this?

Muddiest Points

- Evaluation Measures
  - How to evaluate our term project if we choose a dataset without queries and related documents? Does that mean we have to annotate by ourselves? If so, I think the method of know-item judgements is more suitable than the method of search-guided relevance assessment. How many queries are needed to be annotated?
  - Does Google prefer perfect precision or perfect recall?

Agenda

- Remaining Topics on Evaluation
- Relevance
- Relevance Feedback

Class Goals

- After this class, you should be able to
  - know the basic ideas of relevance feedback and query expansion
  - Know the formulas of relevance feedback in vector space model and probabilistic model

Set-Based Basic Measures

	Relevant	Not relevant
Retrieved	A	B
Not retrieved	C	D

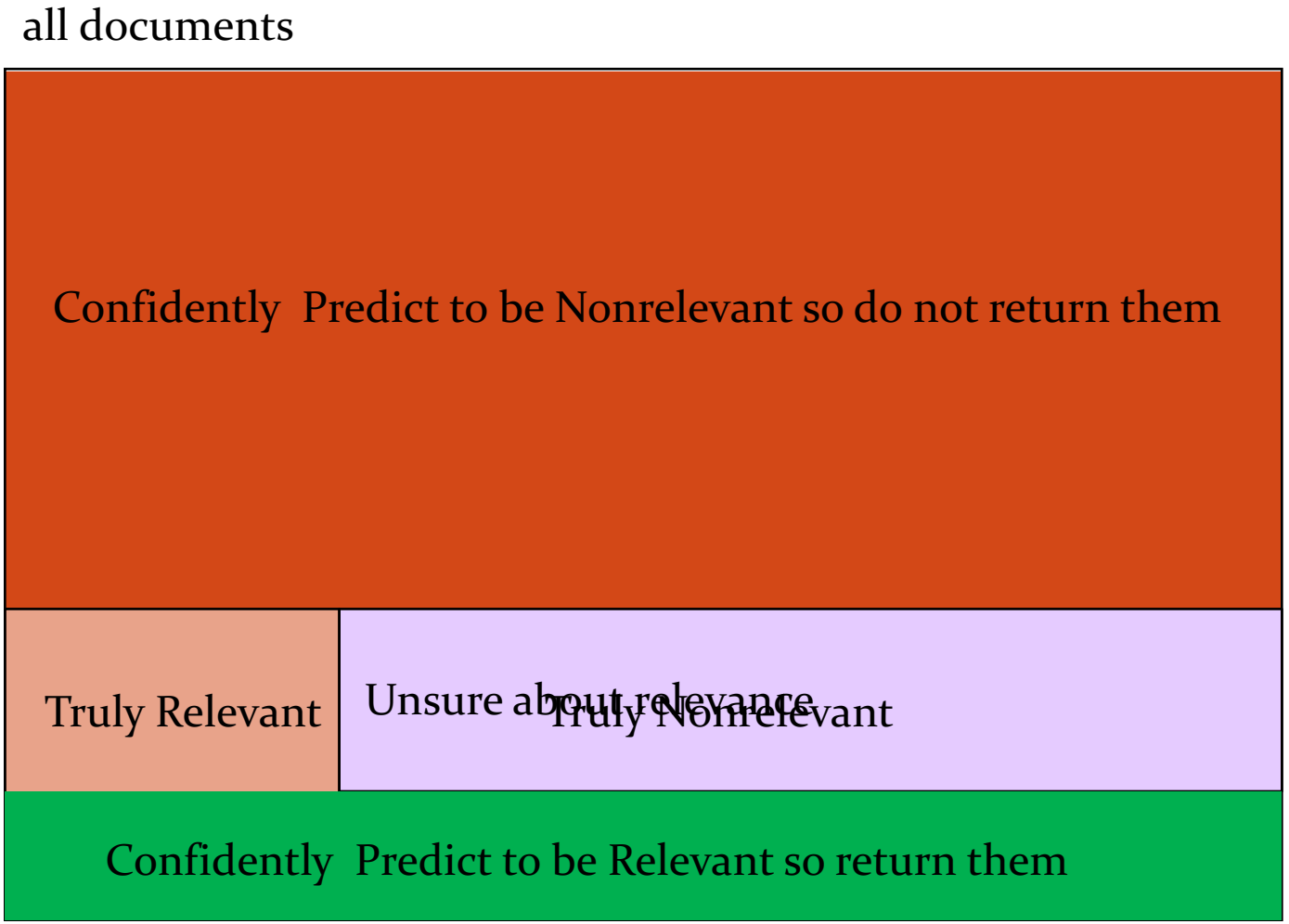
Collection size = A+B+C+D
Relevant = A+C
Retrieved = A+B

- **Precision** = A / (A+B)
- **Recall** = A / (A+C)
- **Miss** = C / (A+C)
- **False alarm** = B / (B+D)
- A: true positive, B: false positive, C: false negative, D: true negative

Precision and Recall

- Precision and Recall are two measures that should be viewed together
  - Precision is the bullseye or the needle in the haystack – you only try to get highly relevant hits (understanding that you may miss other relevant documents.)
  - Recall is the kitchen sink – you try to get all the relevant documents possible (understanding that you may get many non-relevant documents as well.)
- Problematic if consider only one of them
  - A perfect precision retrieval system
  - A prefect recall retrieval system

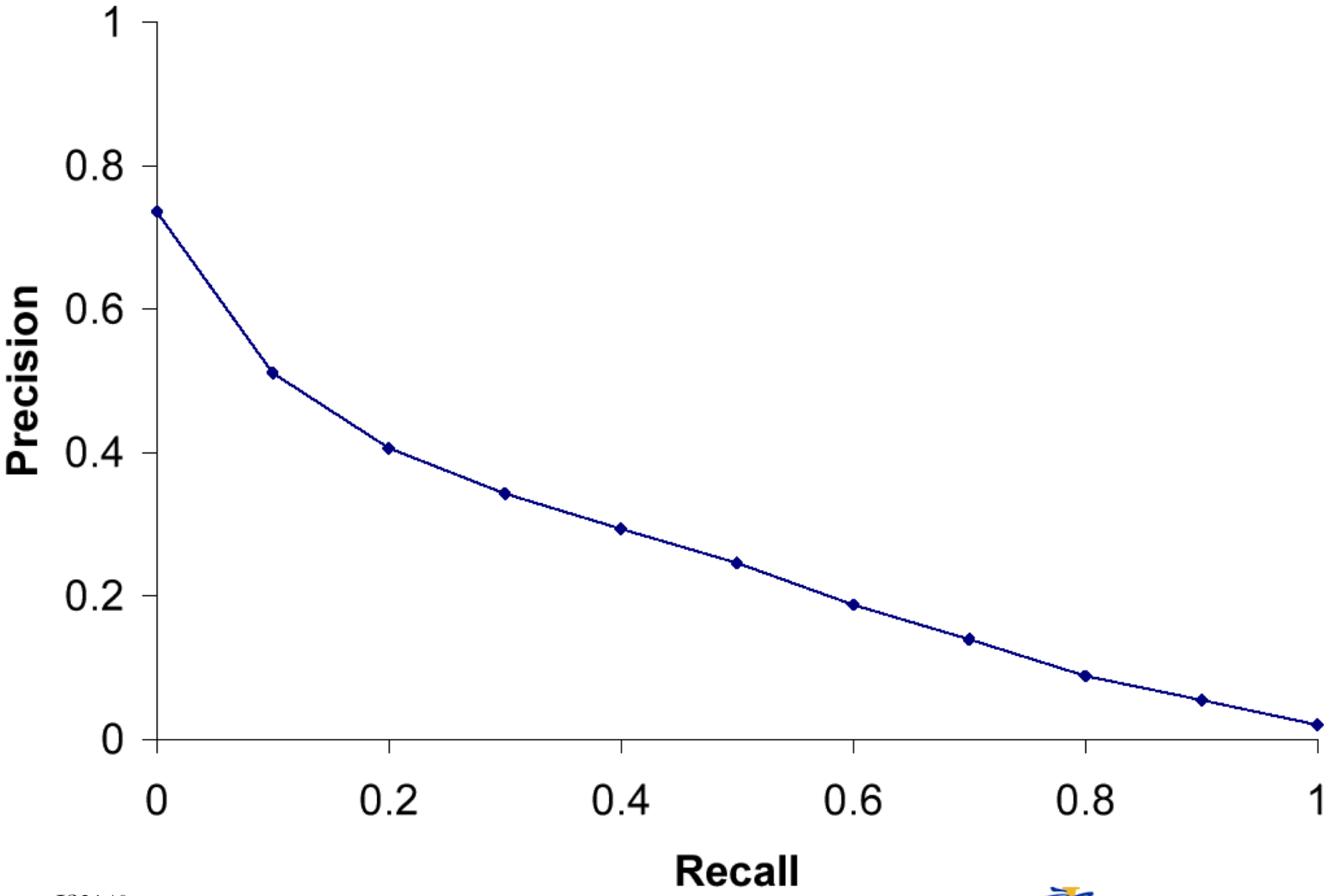
A View of Precision and Recall



Some Exercise on Precision and Recall

- A collection has 1000 documents, and there are 200 relevant documents to topic X. a search on topic X returns 100 documents, among which 80 are relevant. What is precision and what is recall? What is F1?

- A search engine for a given search topic returns 20 relevant documents and 30 nonrelevant documents, after examining the collection, the missed relevant not being returned are 40 documents, so what is the precision and recall of this search?



F-Measure

- Harmonic mean of recall and precision
- Beta controls relative importance of precision and recall
  - Beta = 1, precision and recall equally important, called F1
  - Beta > 1, recall is more important than precision
  - Beta < 1, precision is more important than recall
  - Why?

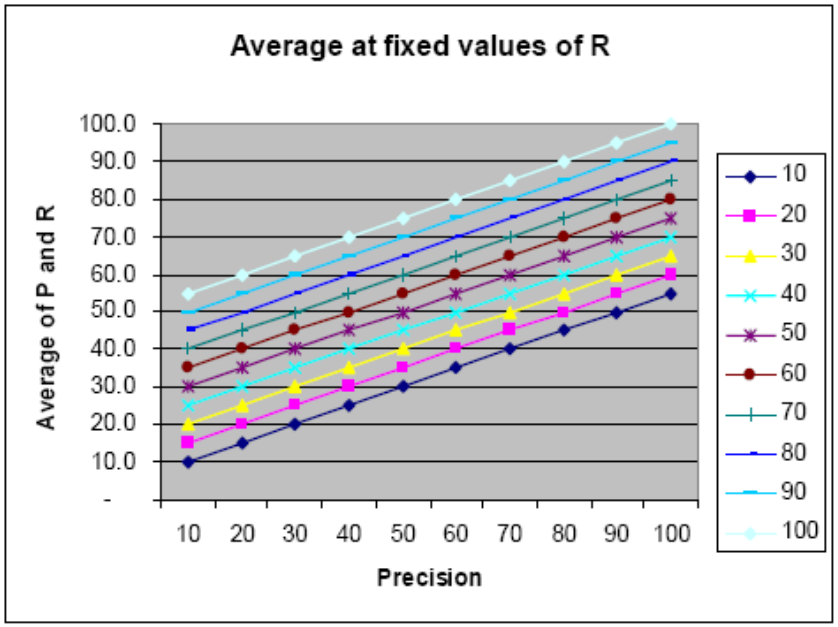
$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R} \quad \text{where} \quad \beta^2 = \frac{1-\alpha}{\alpha}$$

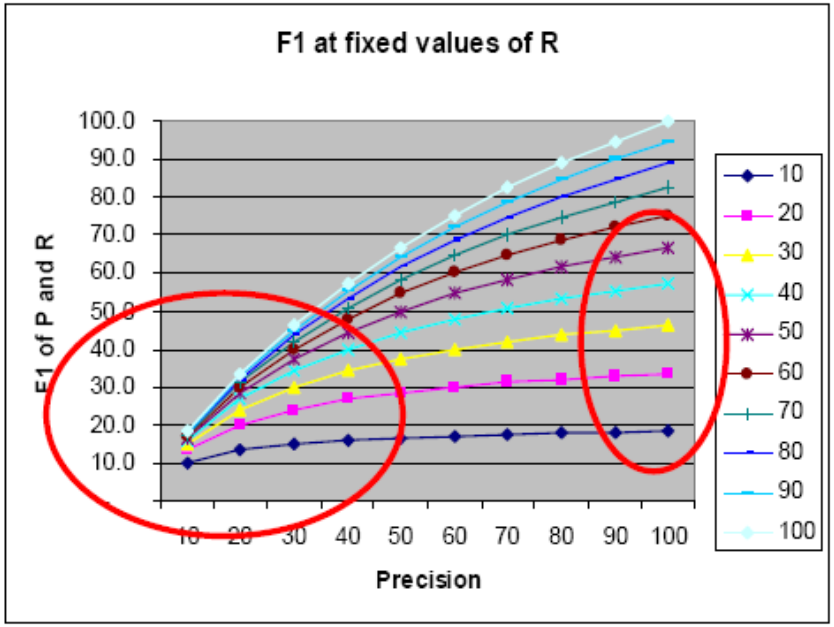
F measure as Harmonic Mean

- Harmonic mean of P and R
  - Inverse of average of their inverses

$$F_1 = \frac{2PR}{P+R} = \frac{1}{\frac{1}{2}(\frac{1}{R}+\frac{1}{P})}$$

- Heavily penalizes low values of P or R
  - Compared to standard average (P+R)/2

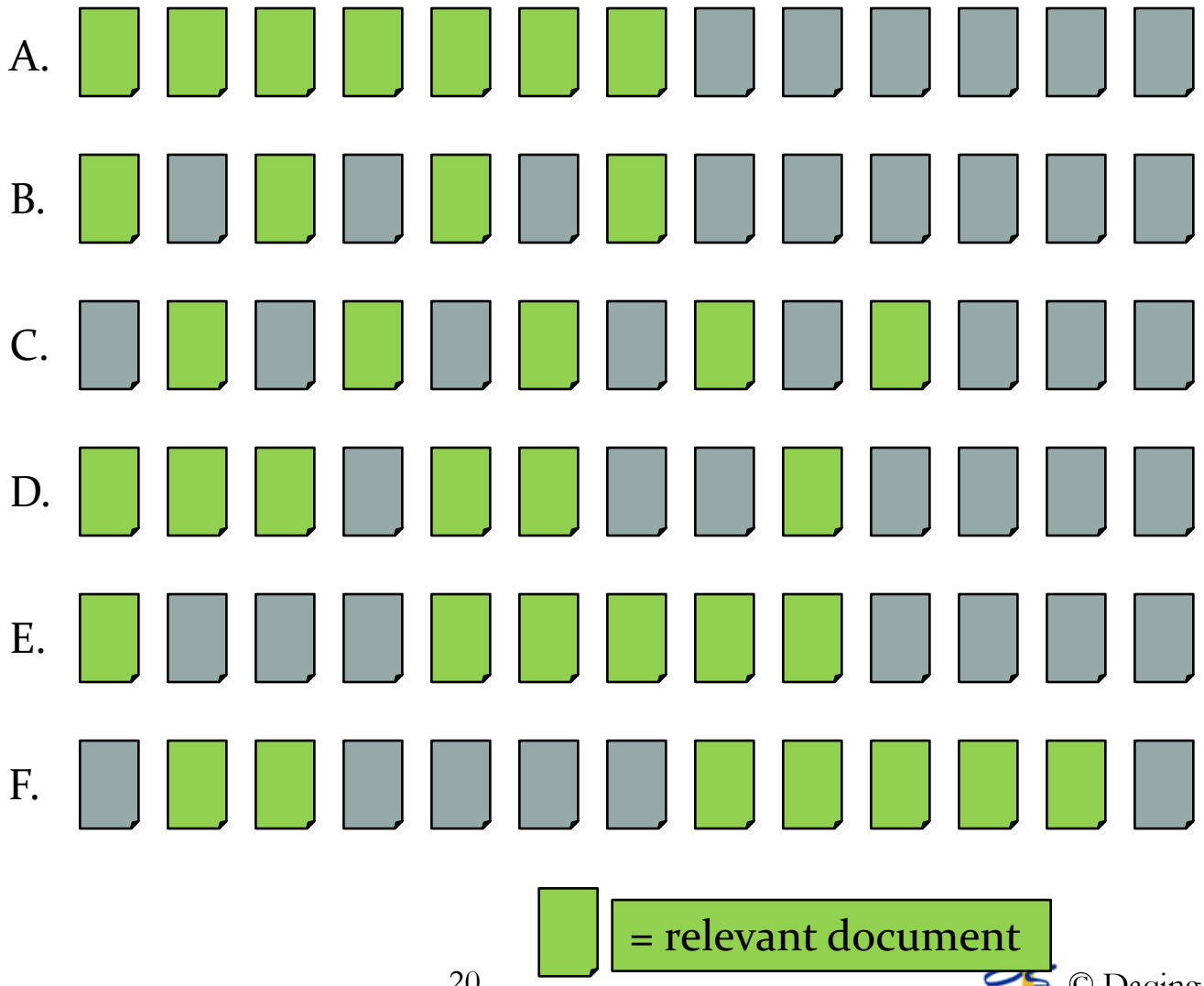




Measures for Ranked Retrieval

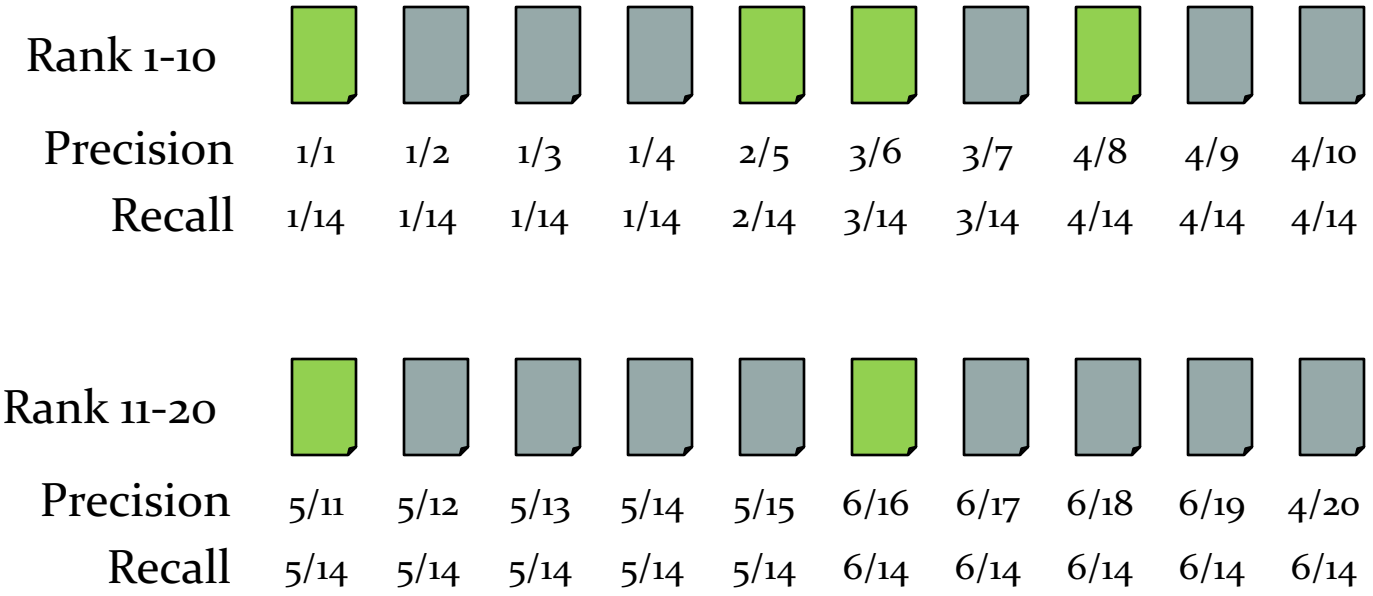
- Returned documents in relevant or not difference, but at the same time the ranking of each relevant document is also important
  - Instead of viewing the returned documents as a set
  - In theory, all documents in the collection are ranked
- Precision and Recall calculations
  - Compute the precision and recall value for each relevant document
  - Compute the precision at fixed recall points (e,g. P at 40% recall)
  - Compute the precision at fixed rank points (e.g., P at rank 20)

Which is the Best Rank Order?



Measuring Precision and Recall

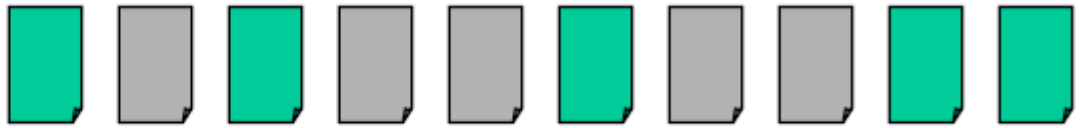
Assume there are a total of 14 relevant docs in a collection of 20 docs



Comparing two Ranked Lists

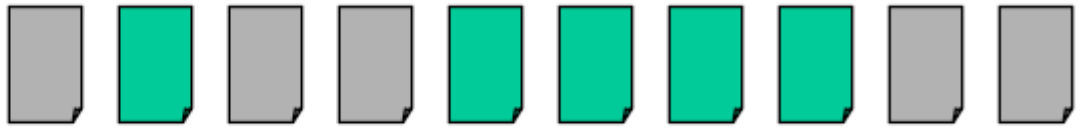
= All relevant documents

Ranking #1



Recall	0.2	0.2	0.4	0.4	0.4	0.6	0.6	0.6	0.8	1.0
Precis.	1.0	0.5	0.67	0.5	0.4	0.5	0.43	0.38	0.44	0.5

Ranking #2

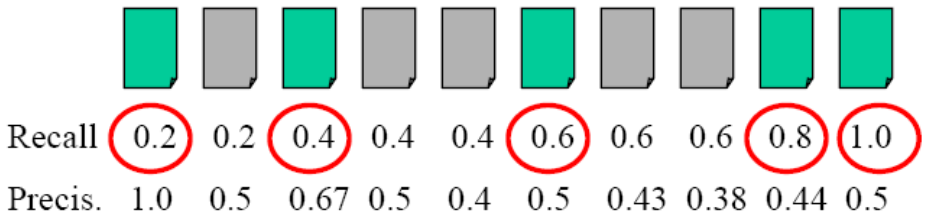


Recall	0.0	0.2	0.2	0.2	0.4	0.6	0.8	1.0	1.0	1.0
Precis.	0.0	0.5	0.33	0.25	0.4	0.5	0.57	0.63	0.55	0.5

Average Precision

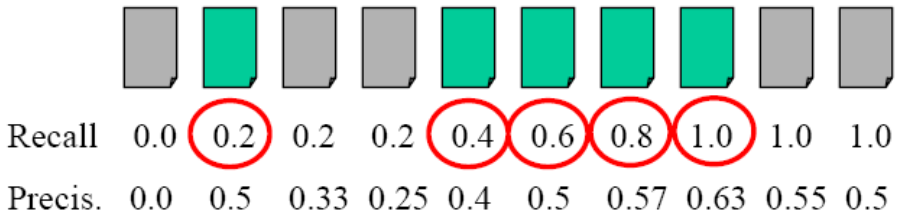
- Often want a single value to indicate the effectiveness in ranked retrieval
- Commonly used measure is average precision
  - Average precisions when recall increases (when meet each relevant document)

Ranking #1



AvgPrec = (1+0.67+0.5+0.44+0.5)/5 = 0.622

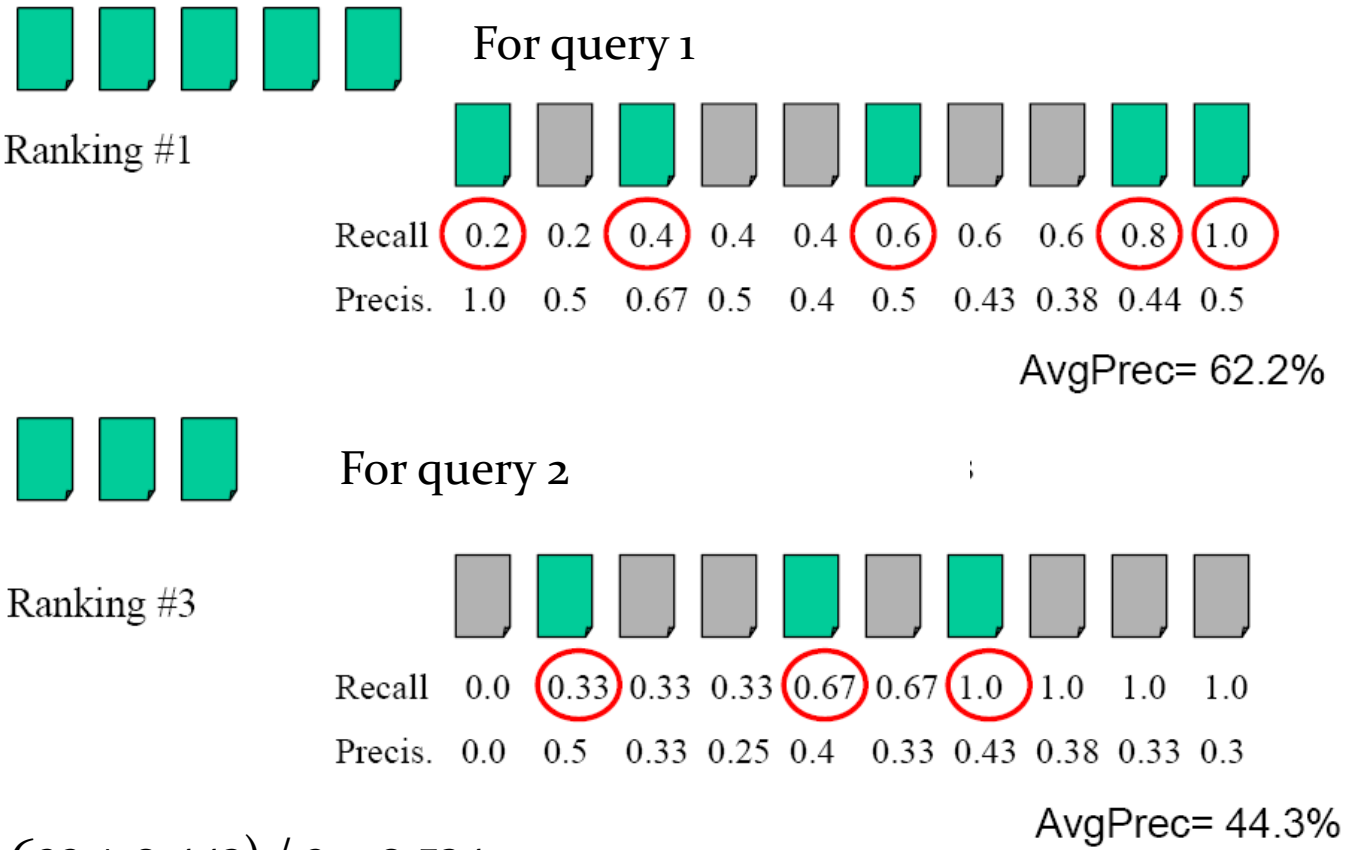
Ranking #2



AvgPrec = (0.5+0.4+0.5+0.57+0.63)/5 = 0.52

Mean Average Precision (MAP)

- Mean over the average precision of multiple queries.



MAP = (0.622 + 0.443) / 2 = 0.534

Mean Average Precision (MAP) - II

- The formula for MAP

$$\mathrm{MAP}(Q) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m_j}\sum_{k=1}^{m_j}\mathrm{Precision}(R_k)$$

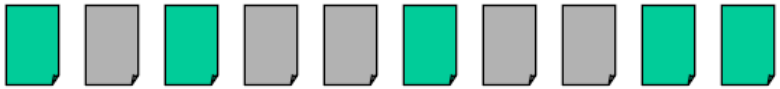
the set of relevant documents for an information need $q_j \in Q$ is $\{d_1, \ldots d_{m_j}\}$ and R_k is the set of ranked retrieval results from the top result until you get to document d_k, then

- Question: what happens if not all relevant documents are returned
  - Because average precision assumes that there is a precision value for every recall point
- Solutions:
  - Put the unretrieved documents at the bottom of the extension
  - Essentially, assume that the precision value for the unretreived docs is 0

Therefore

- If we know that there are totally 10 relevant documents, what is the average precision then?

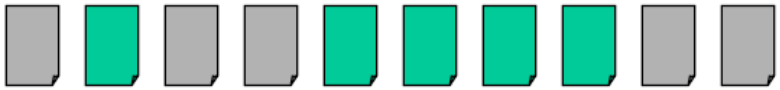
Ranking #1



AvgPrec = ?

Precis. 1.0 0.5 0.67 0.5 0.4 0.5 0.43 0.38 0.44 0.5

Ranking #2



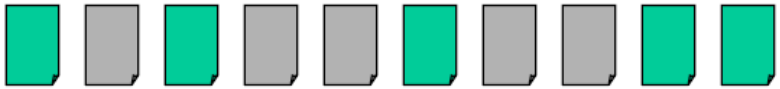
AvgPrec = ?

Precis. 0.0 0.5 0.33 0.25 0.4 0.5 0.57 0.63 0.55 0.5

Therefore

- If we know that there are totally 10 relevant documents, what is the average precision then?

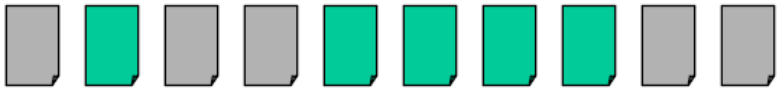
Ranking #1



Precis. 1.0 0.5 0.67 0.5 0.4 0.5 0.43 0.38 0.44 0.5

AvgPrec = (1+0.67+0.5+0.44+0.5)/10 = 3.11/10

Ranking #2



Precis. 0.0 0.5 0.33 0.25 0.4 0.5 0.57 0.63 0.55 0.5

AvgPrec = (0.5+0.4+0.5+0.57+0.63)/10 = 2.6/10

Measures for Ranked Retrieval

- Returned documents in relevant or not difference, but at the same time the ranking of each relevant document is also important
- Precision and Recall calculations
  - Compute the precision and recall value for each relevant document
  - Compute the precision at fixed recall points (e,g. P at 40% recall)
  - Compute the precision at fixed rank points (e.g., P at rank 20)
- Fdifferent queries, different relevant documents in the collection. Precision at fixed rank points cannot perfectly describe the search engine's retrieval ability.
  - e.g., for first query, 5 relevant documents in total; for second query, 10 relevant documents in total. P at rank 20 is unfair.

R- Precision

- Precision at the R-th position in the ranking of results for a query that we know has R relevant documents in the collection

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	

R = # of relevant docs = 6

R-Precision = 4/6 = 0.67

Mean Reciprocal Rank (MRR)

- MRR is the mean of Reciprocal Rank of a set of queries
  - Reciprocal Rank is the reciprocal of the first relevant document's rank in the ranked list
  - E.g.
    - The first relevant document is at rank 5
    - RR = 1/5 = 0.2
- Which type of searches is MRR a good measure?

Cumulative Gain

- Cumulative Gain (CG) is the sum of the graded relevance values of all results in a search result list. The CG at a particular rank position p is defined as:

$$\mathrm{CG_p} = \sum_{i=1}^{p} rel_i$$

where rel_i is the graded relevance of the result at position i.

- E.g. For ranked documents $D_1, D_2, D_3, D_4, D_5, D_6$ with relevance scores as 3,2,3,0,1,2. CG is

$$\mathrm{CG_p} = \sum_{i=1}^{p} rel_i = 3 + 2 + 3 + 0 + 1 + 2 = 11$$

Discounted Cumulative Gain

- CG is not really sensitive to the ranking
- But we know ranked list follow two assumptions:
  - Highly relevant documents are more useful when appearing earlier in a search engine result list (have higher ranks)
  - Highly relevant documents are more useful than marginally relevant documents, which are in turn more useful than irrelevant documents.
- Therefore, we would use Discounted Cumulative Gain (DCG)

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

DCG example

- For ranked documents $D_1, D_2, D_3, D_4, D_5, D_6$ with relevance scores as 3,2,3,0,1,2.

i	Reli	$\frac{2^{rel_i}-1}{log_2(1+i)}$
1	3	(2^3-1)/log(1+1) =7
2	2	(2^2-1)/log(1+2)= 1.89
3	3	(2^3-1)/log(1+3) = 3.5
4	0	(2^0-1)/log(1+4)= 0
5	1	(2^1-1)/log(1+5) = 0.39
6	2	(2^2-1)/log(1+6) = 1.07

DCG_6 = 7+1.89 + 3.5 + 0 + 0.39+ 1.07 = 13.85

Normalized DCG

- DCG has problems in aggregating among different topics
- Better normalized DCG to a fixed range of values
- Ideal (Perfect) ranking: order documents to be the monotonically decreasing sort of the relevance judgment scores
  - E.g. for all documents to be ranked $D_1, D_2, D_3, D_4, D_5, D_6$ with relevance scores as 3,2,3,0,1,2, and ideal ranking would be 3,3,2,2,1,0
    - i.e., one idea ranking is D1,D3,D2,D6,D5,D4, but there could be more idea ranking
- For an ideal ranking, we can have DCG too, it is called IDCG
  - $IDCG_6 =$

IDCG

- For a topic with all relevant documents D_1, D_3, D_2, D_6, D_5 with relevance scores as $3, 3, 2, 2, 1$

i	Reli	$\frac{2^{rel_i}-1}{log_2(1+i)}$
1	3	$(2^3\text{-}1)/\log(1\text{+}1) = 7$
2	3	$(2^3\text{-}1)/\log(1\text{+}2) = 4.42$
3	2	$(2^2\text{-}1)/\log(1\text{+}3) = 1.5$
4	2	$(2^2\text{-}1)/\log(1\text{+}4) = 1.29$
5	1	$(2^1\text{-}1)/\log(1\text{+}5) = 0.39$

$IDCG_6$ =
= (7+
4.42+
1.5+
1.29+
0.39) = 14.6

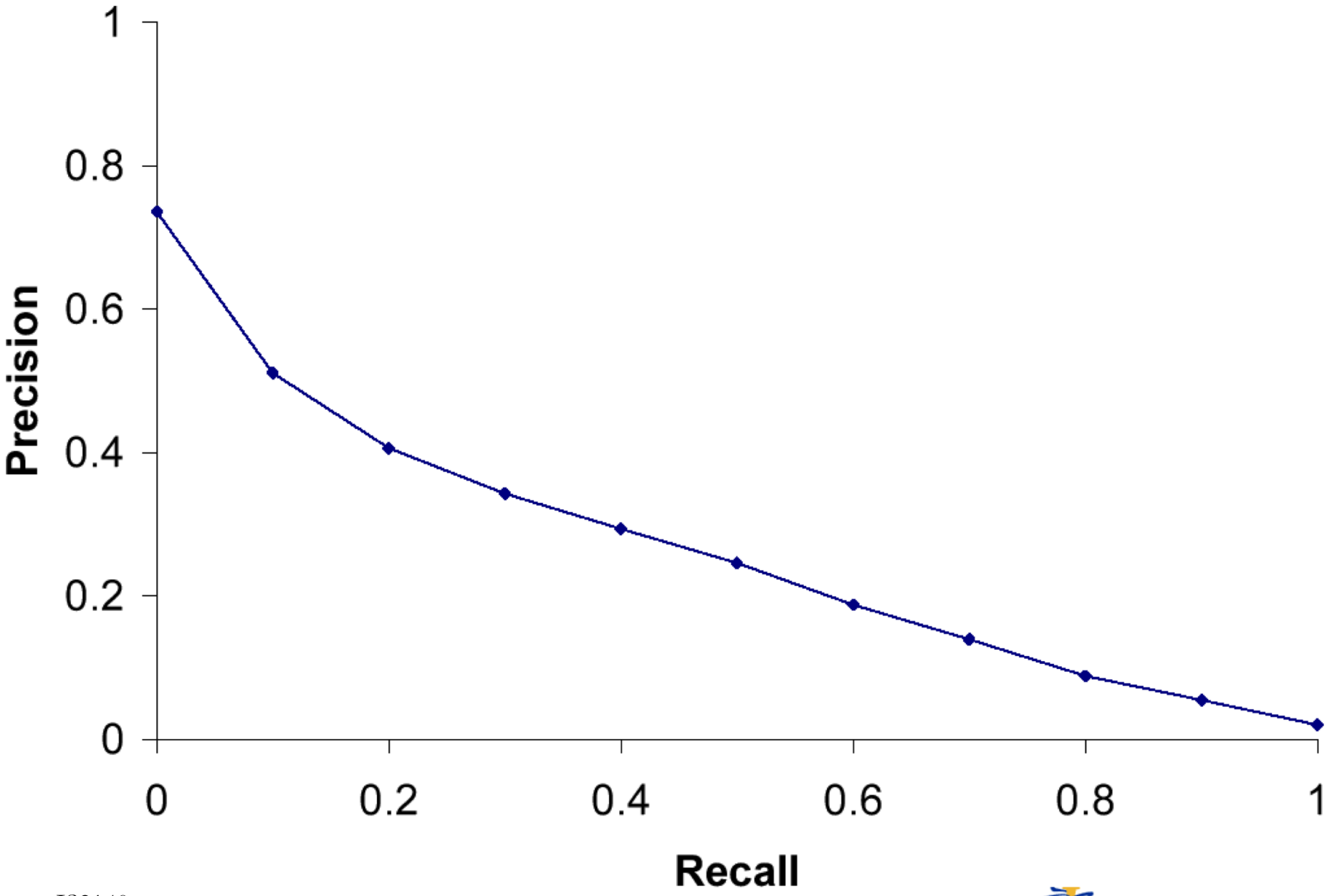
NDCG

- Normalized DCG therefore is

$$\mathrm{nDCG_p} = \frac{DCG_p}{IDCGp}$$

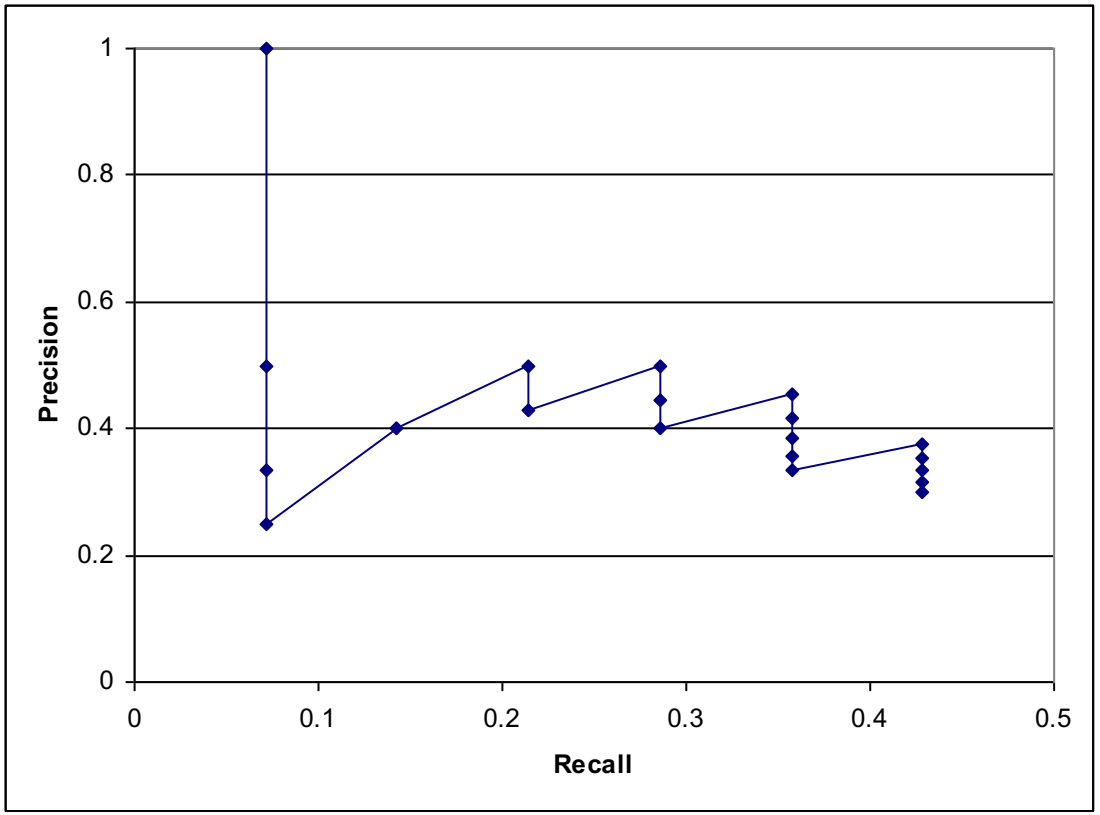
- IIR use the following formula where Zk is the normalization factor

$$\mathrm{NDCG}(Q, k) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} Z_k \sum_{m=1}^{k} \frac{2^{R(j,m)} - 1}{\log(1+m)},$$



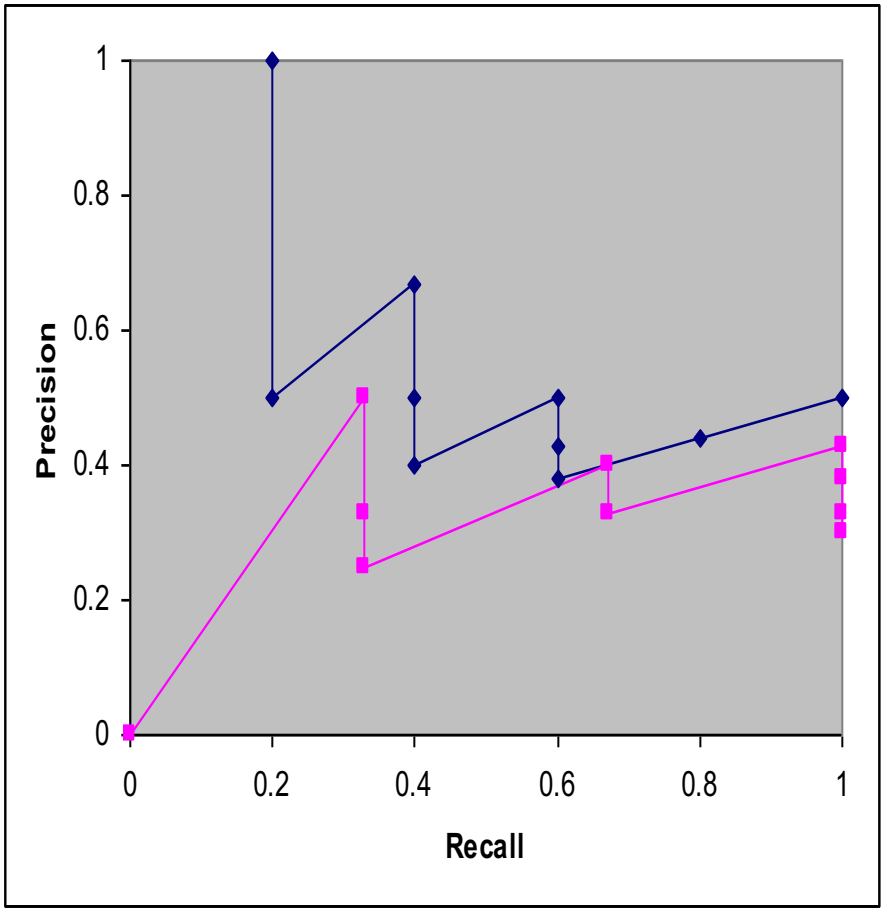
Precision and Recall Graph

- Average precision only gives a value, good summary, but
  - Sometimes, want to see the precision/recall tradeoff
  - Plot each (recall, precision) point on a graph

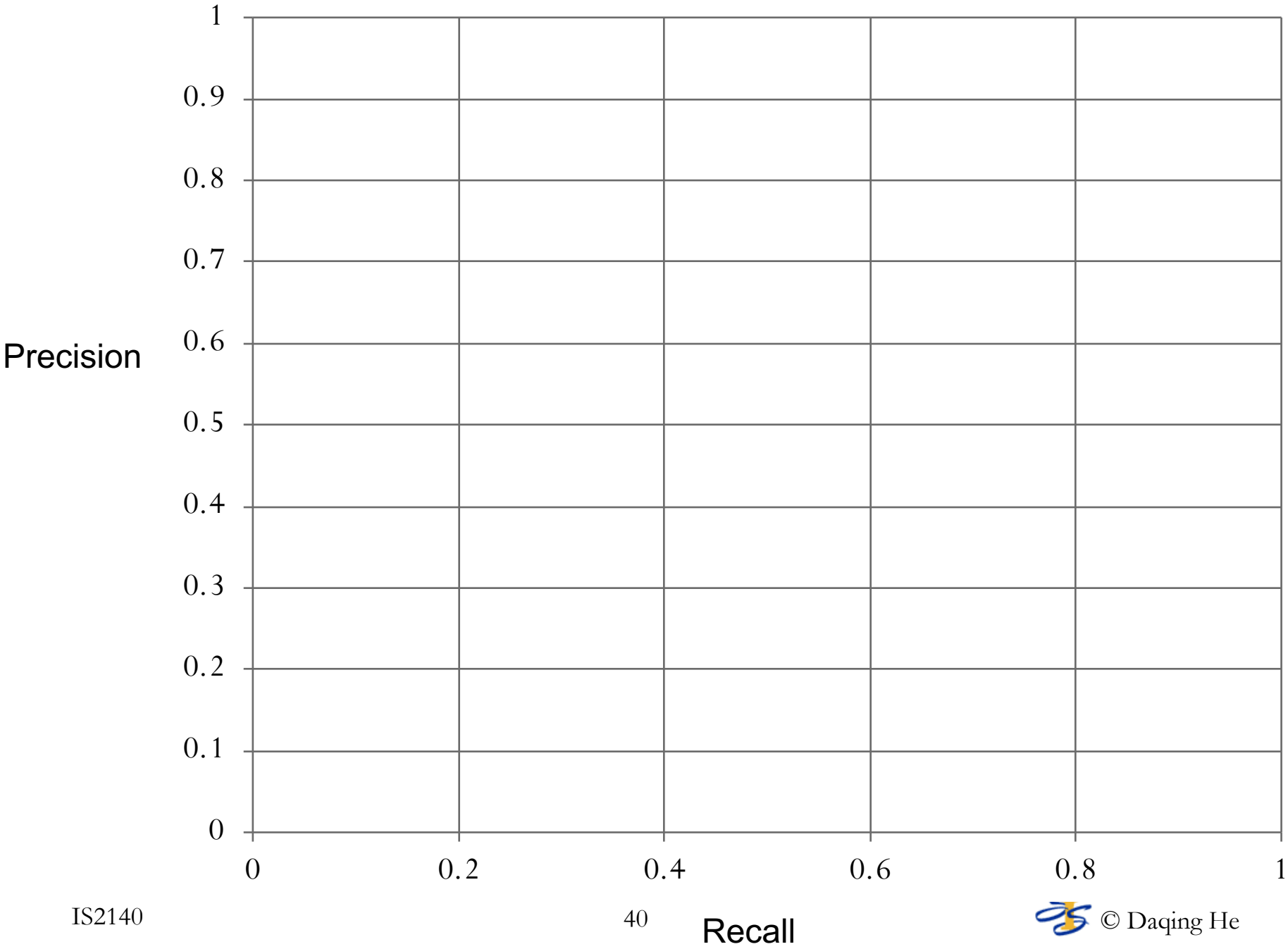


Averaging over Graph

- Recall/Precision graph has the odd sawtooth shape when plot directly
  - Why?
- Problems
  - How do you compare performance across queries?
  - How to obtain precision value when recall is at, say, 30%
  - Can the sawtooth shape describe what's going on with the retrieval result?

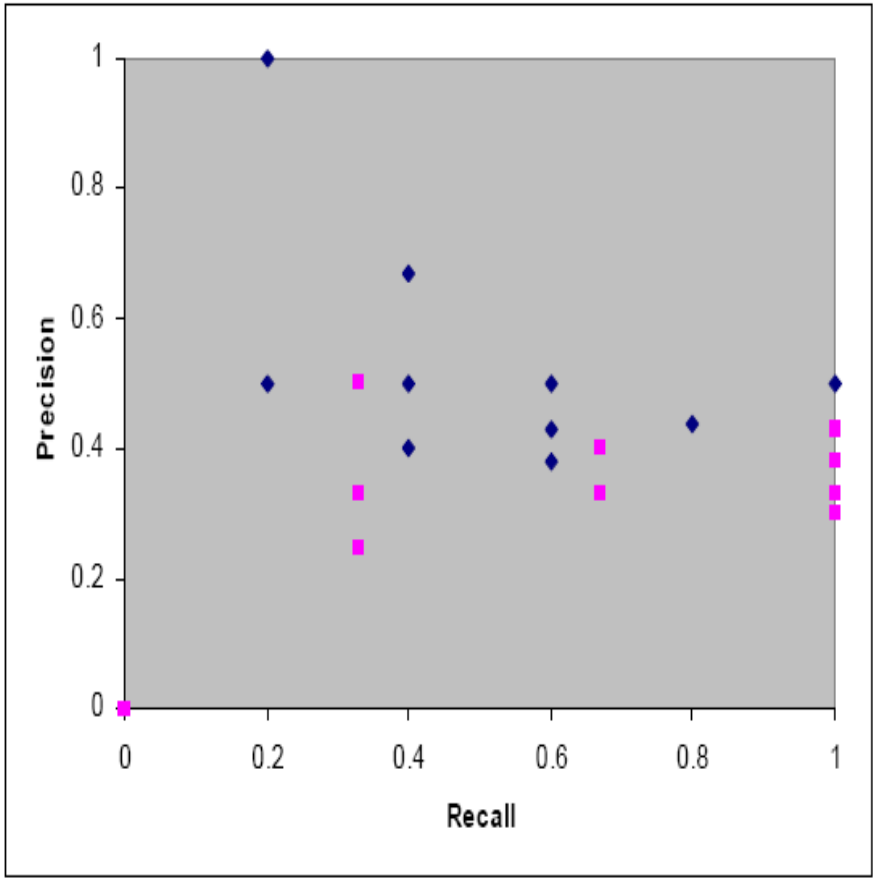


Solution: Interpolation!, but how



Averaging over Graph

- Recall/Precision graph has the odd sawtooth shape when plot directly
  - Why?
- Problems
  - How do you compare performance across queries?
  - How to obtain precision value when recall is at, say, 30%
  - Can the sawtooth shape describe what's going on with the retrieval result?



Solution: Interpolation!, but how

Interpolation

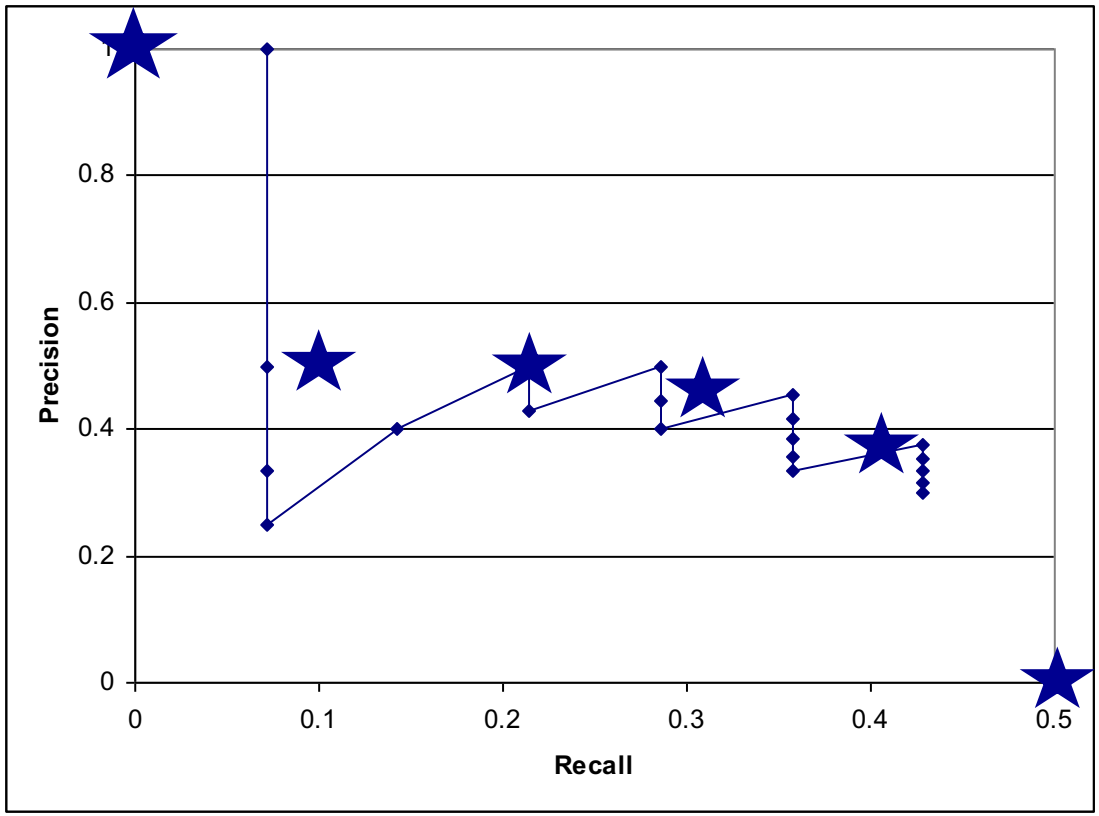
- It is an empirical fact that *on average* as recall increases, precision decreases
  - Verified time and time again
  - *On average*
- Seems reasonable to aim for an interpolation that makes function monotonically decreasing
- One approach:

$$P(R) = \max\{P' \ : \ R' \geq R \wedge (R', P') \in S\}$$

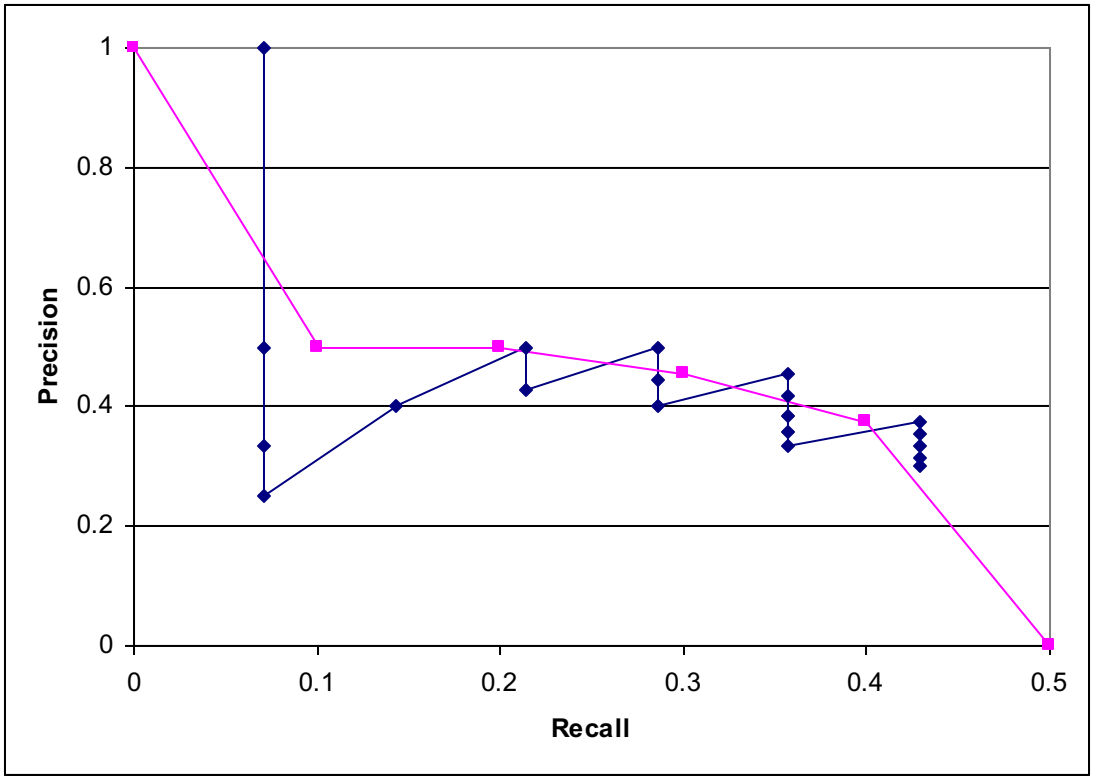
where S is the set of observed (R,P) points

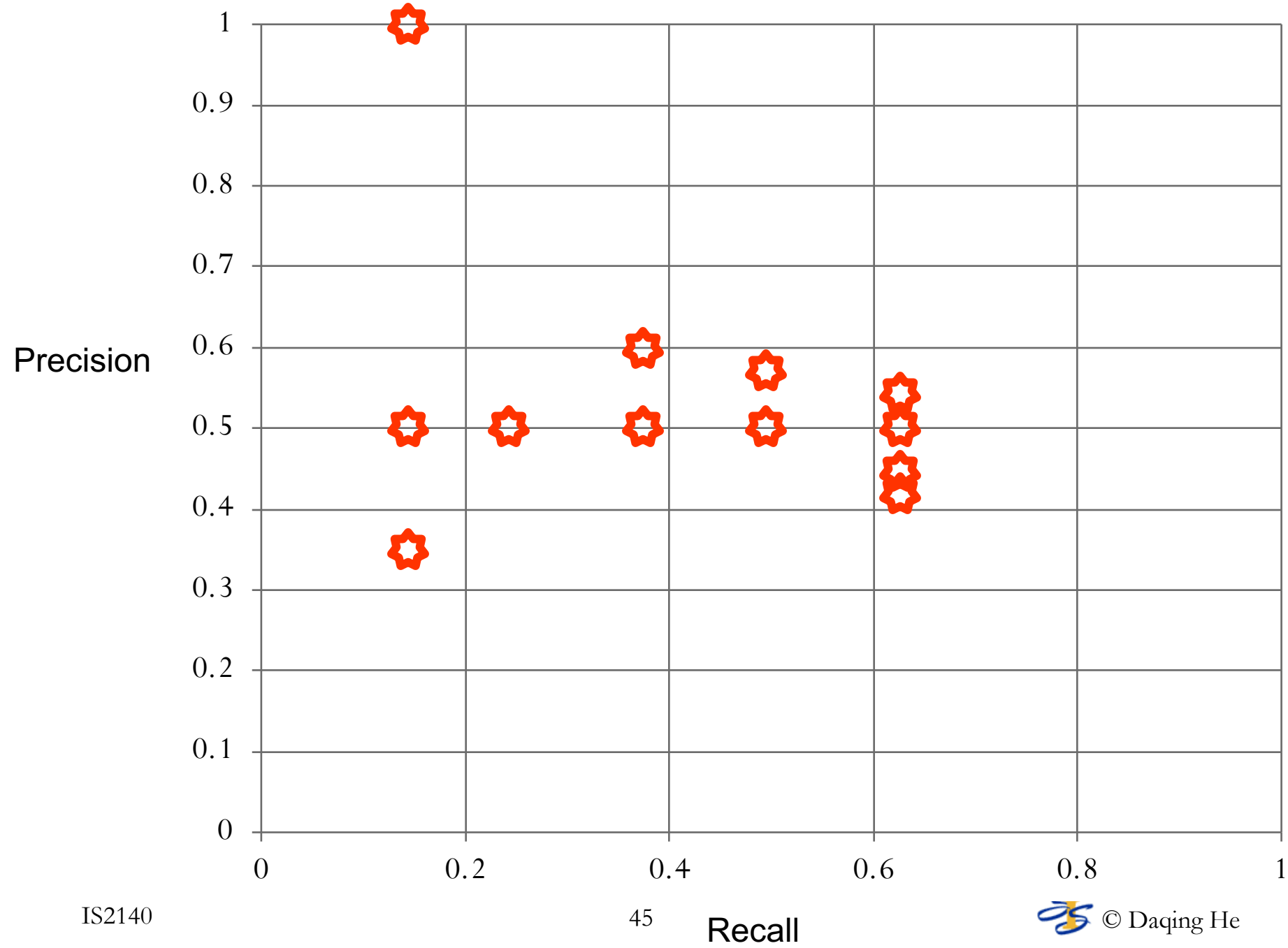
Interpolation

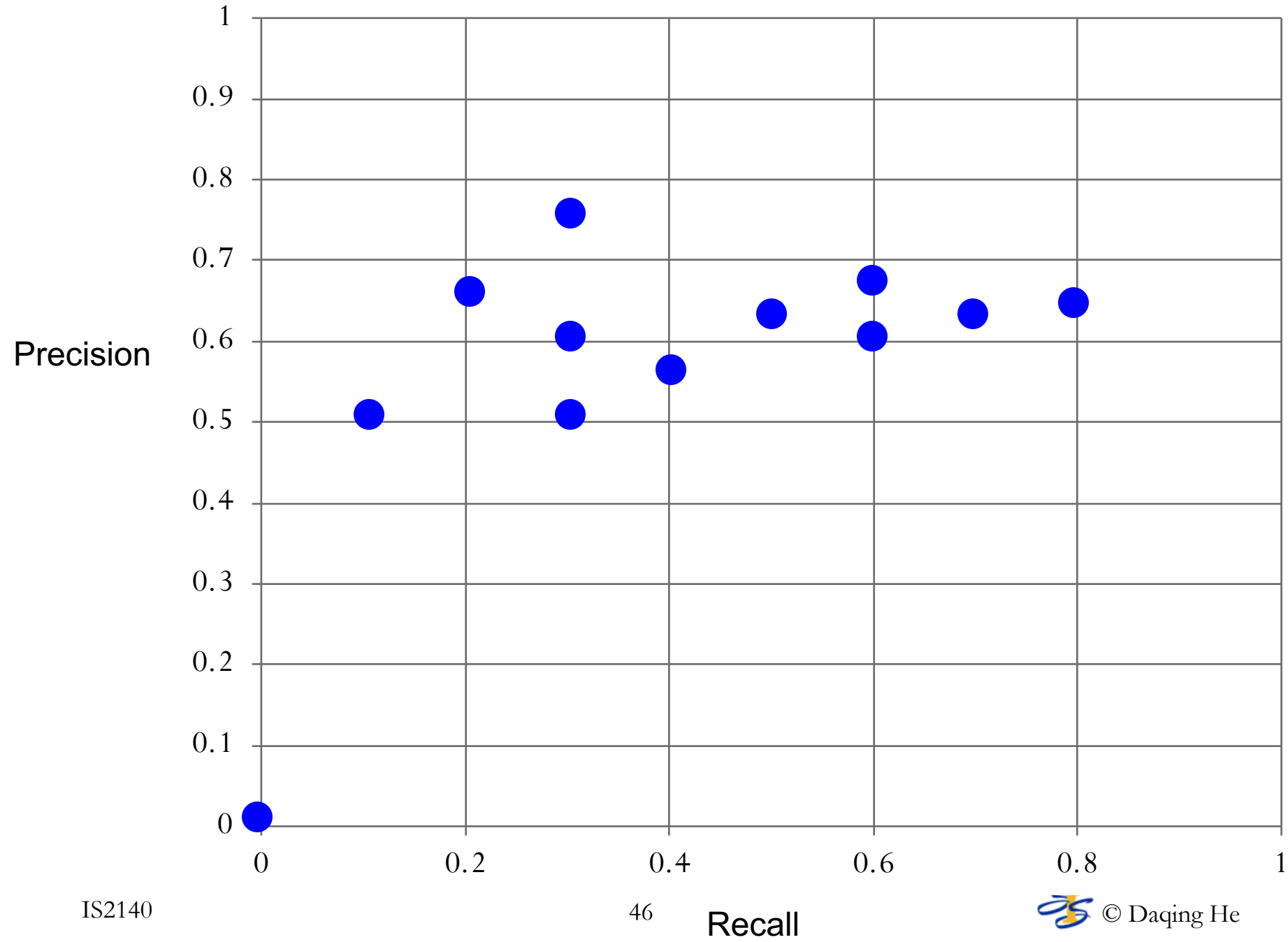
- How to perform interpolation?



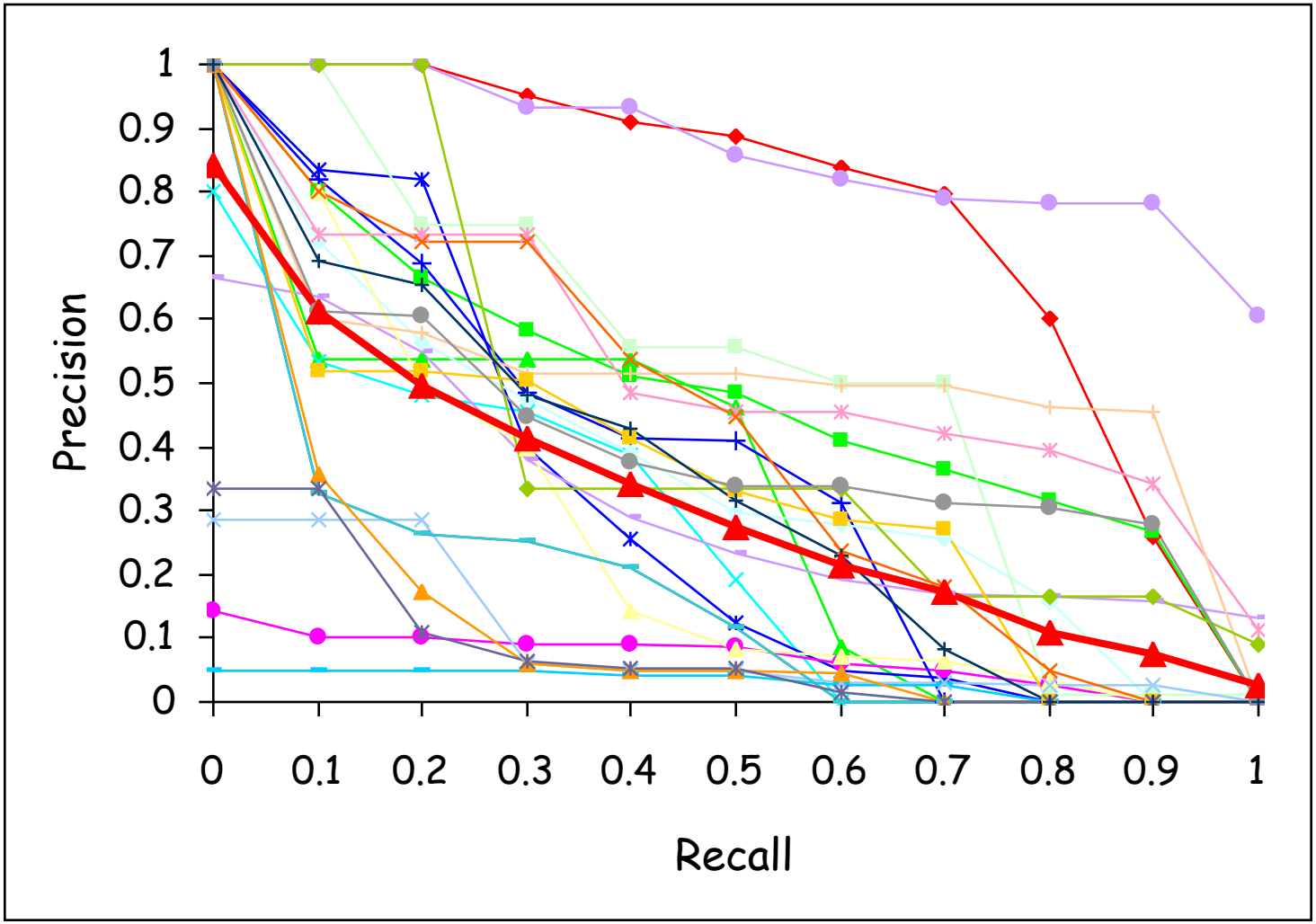
Result of Interpolation







What to do with the curves?



Adapted from a presentation by Ellen Voorhees at the University of Maryland, March 29, 1999

trec_eval

- trec_eval is the standard evaluation software for ad-hoc retrieval
  - Written by Chris Buckley while at Cornell
  - http://trec.nist.gov/trec_eval/
- What trec_eval reports:
  - Retrieved: Number of documents retrieved by your program
  - Relevant: The number of relevant documents in the database
  - Rel_ret: The number of relevant documents your program found
  - Interpolated Recall at 11 Recall points
  - Average precision over all relevant documents
  - Precision at ranks 5, 10, 15, 20, 30, 100, 200, 500, and 1000
  - R-Precision: Precision at rank r, where r is the number of relevant
  - And others
- Statistics provided on a by-query or by-query-set basis

Why significance tests?

- System A and B identical on all but one query:
  - Is it just a lucky query for System A?
  - Need A to beat B frequently to believe it is really better
  - Need as many queries as possible

Empirical research suggests 25 is minimum needed
TREC tracks generally aim for at least 50 queries

- System A beats system B on every query:
  - But only does so by 0.00001%
  - Does that mean much?
- Significance tests consider those issues

Averages Can Deceive

Experiment 1

Query	System A	System B
1	0.20	0.40
2	0.21	0.31
3	0.22	0.42
4	0.19	0.25
5	0.17	0.27
6	0.20	0.30
7	0.21	0.22
Average	0.20	0.31

Experiment 2

Query	System A	System B
1	0.20	0.20
2	0.21	0.21
3	0.22	0.22
4	0.19	0.96
5	0.17	0.17
6	0.20	0.20
7	0.21	0.21
Average	0.20	0.31

How Much is Enough?

- Measuring improvement
  - Achieve a meaningful improvement

Guideline: 0.05 is noticeable, 0.1 makes a difference (in MAP)

  - Achieve reliable improvement on "typical" queries

- Sign test or Wilcoxon signed-ranks test are typical

Wilcoxon signed rank test for paired samples

  - Do not require that data be normally distributed
  - Sign test answers how often
  - Wilcoxon answers how much
  - Sign test is crudest but most convincing

Evaluating Ranked Retrieval

Test collection: topics, documents, relevance judgments

Topic T1 for System A

Rank	Doc#	Score	Rel?	Prec.
1	FR05	0.97	R	1.00
2	FR03	0.91	R	1.00
3	FR02	0.88		
4	FR10	0.82		
5	FR07	0.80	R	0.60
6	FR04	0.77		
7	FR06	0.63	R	0.57
8	FR08	0.62		
9	FR09	0.55		
10	FR01	0.51	R	0.50
Avg. Prec. (AP):				0.73

System	A	B	Wilcoxon Test	
Topic	AP	AP	A-B	Signed Rank
T1	0.73	0.50	+0.23	+9
T2	0.45	0.38	+0.07	+3.5
T3	0.56	0.36	+0.20	+8
T4	0.00	0.09	-0.09	-5
T5	0.13	0.10	+0.03	+1
T6	1.00	0.83	+0.17	+7
T7	0.24	0.28	-0.04	-2
T8	0.47	0.20	+0.27	+10
T9	0.53	0.41	+0.12	+6
T10	0.23	0.30	-0.07	-3.5
MAP:	0.43	0.35		W+ = 44.5 W- = 10.5

If min(W+, W-) < 8 ➔ difference is not significant (two-tailed, p=0.05)

Relevance

Relevance

- Relevance is the basis for evaluating IR
  - Both precision and recall depends on "relevance"
- However, Relevance is difficult to define precisely
  - No universal accepted definition
    - Except all agree that it is hard to define relevance
- The full set of relevant documents is never known
  - It is impractical to judge every document in the collection…
  - so it is very likely that some relevant documents were missed

What is relevance?

Relevance is the	*measure* *degree* *dimension* *estimate* *appraisal* *relation*	of a	*correspondence* *utility* *connection* *satisfaction* *fit* *bearing* *matching*
existing between a	document article textual form reference information provided fact	and a	query request information used point of view information need statement
as determined by	*person* *judge* *user* *requester* *Information specialist*		

Does this help?

Tefko Saracevic. (1975) Relevance: A Review of and a Framework for Thinking on the Notion in Information Science. Journal of the American Society for Information Science, 26(6), 321-343;

Relevance - II

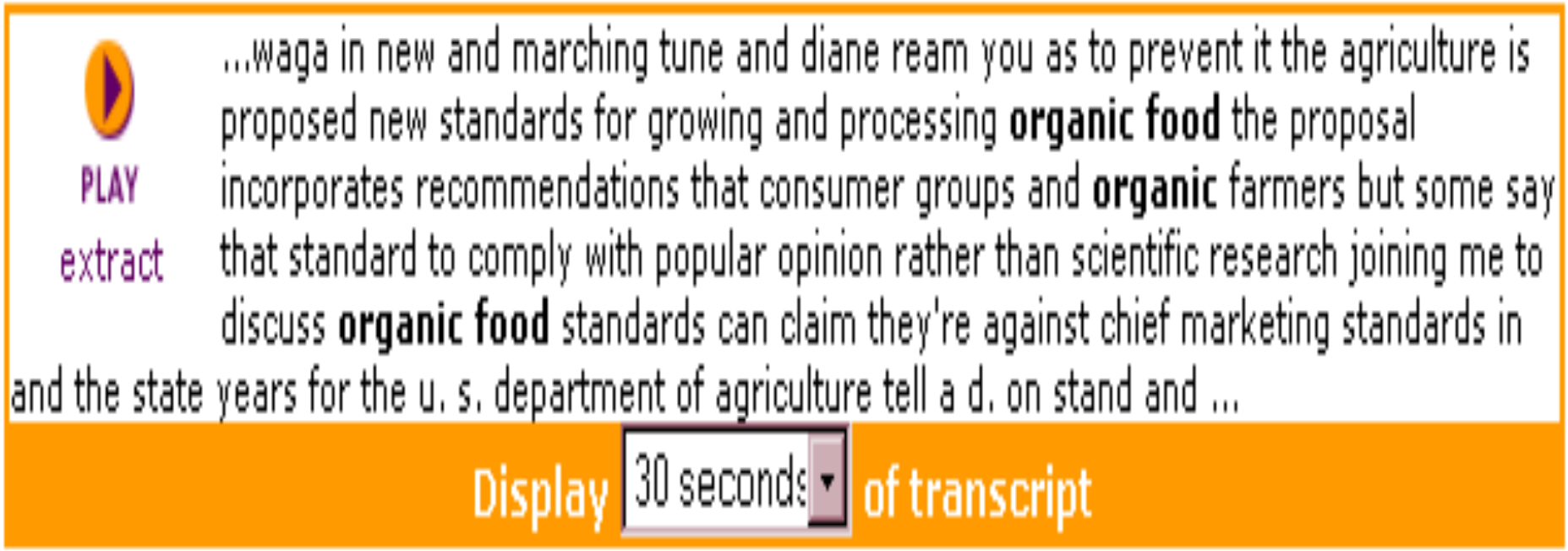
- A relevant document is one that a person judges as useful in the context of a specific information need
  - Who does the judging?
  - How does the person define "useful"?
  - People aren't consistent
  - A person's judgment depends upon more than document and query
    - E.g., what the person knew before running the query
- Two related concepts
  - Topical relevance
  - Utility

Topical Relevance

- Saracevic's "system view" of relevance
  - Concern the "aboutness"
- Assume that relevance is
  - solely a property of the internal mechanism of the retrieval system
  - is related the content of the document, so it is objective
  - is the result of the match between a query and the document representation
  - virtually ignoring the role of user

An Example

- The query is "organic food"



Topical Relevance - II

- Limitations
  - Assume relevance to a query = relevance to a need
  - Assume relevance to be objective (about the content), not subjective (related to users, to other documents)
  - Assume relevance is static, not change
- But is Useful for evaluating IR system and algorithms
  - No user around, so cheap to run experiments
  - Relevance of docs is static, so can run experiments many times
  - Relevance of documents only related to the content and the mechanisms inside IR system, so provide basis for comparing effectiveness of different retrieval systems
  - Topical relevance is an important factor in users' relevance judgment

Utility

- Cooper's catch-all concept
  - Not only include topical relevance
  - But also quality, novelty, importance, credibility and many other features of the documents to the user's need.
- Not concentrate on "aboutness", but on "usefulness"
  - Some documents could not be on "aboutness", but still be "useful"
  - Some document could be on "aboutness", but still not be "useful"

Relevance Judgment

- Barry's 23 categories of relevance criteria
  - Not for topical relevance, but for pertinence and utility
  - Organized into 7 broad groups
    - Criteria to the information content
    - Criteria to users' previous experience and knowledge
    - Criteria to user's beliefs and preferences
    - Criteria to other information in the environment
    - Criteria to the sources of documents
    - Criteria to the documents as a physical entity, and
    - Criteria to the user's situation

Carol L. Barry: User-Defined Relevance Criteria: An Exploratory Study. JASIS 45(3): 149-159 (1994)

Barry's Study

- Total 444 responses made by 18 respondents examining 242 documents
  - Top 5 criteria by percentage of total responses
    - Depth/scope, 14%
    - content novelty, 12%
    - subjective accuracy/validity, 10%
    - tangibility, 7%
    - Recency 6%
  - Top 5 criteria by the number of respondents mentioning
    - Depth/scope, 16
    - subjective accuracy/validity, 13
    - consensus within the field, availability/environment, 11
    - content novelty, background/experience 10
    - Affectiveness, expernal verification, 9

Barry's Study

- Mentions of groups of criterion categories
  - Top 5 mentioned categories
    - Criteria for information content 35%,
    - criteria for user's background/experience, 22%,
    - criteria for user's belief and preferences 16%,
    - criteria to other information 15%,
    - criteria to source 7%
  - Top 5 mentioned categories by # respondents
    - Criteria for information content 18,
    - criteria to other information 18,
    - criteria for user's background/experience 15,
    - criteria for user's belief and preferences 14,
    - criteria to source 11

Relevance Feedback

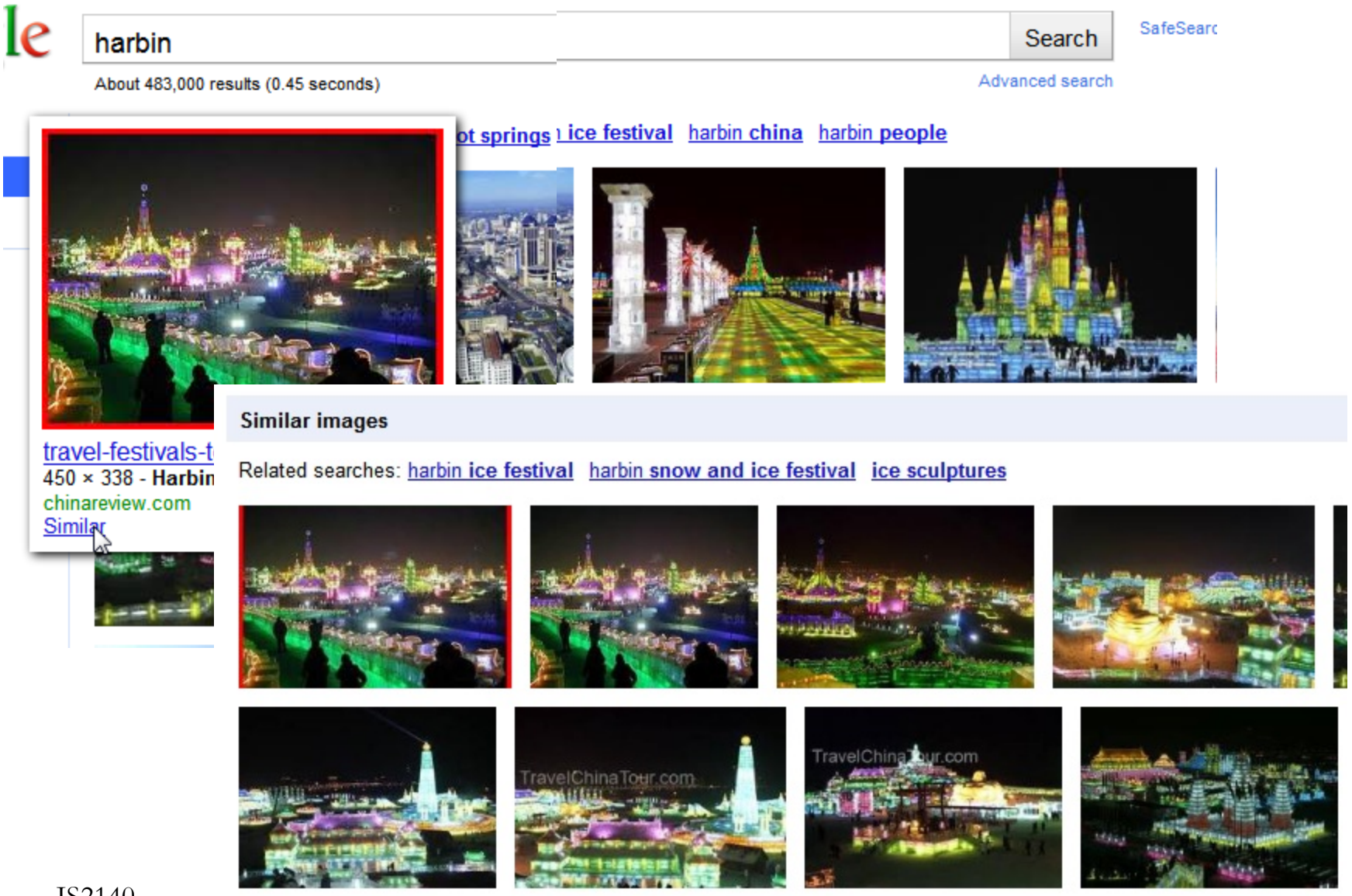
Basic Settings

- Search System heavily reply on queries for finding relevant docs
- But a query only approximates user's information need
  - User initial query is often short and poor approximation
  - People can improve query when seeing relevant and non-relevant docs
    - Adding and removing terms
    - Adjust terms for the same concepts
    - Reweight terms
    - Change query structures
- Question: can better query be **automatically** generated by analyzing relevant/non-relevant docs? – called **relevance feedback**

Procedure of Relevance Feedback

- Basic procedure
  - The user issues a (short, simple) query
  - The system returns an initial set of retrieval results
  - The user marks some returned documents as relevant or not relevant
  - The system computes a better representation of the information need based on the user feedback
  - The system displays a revised set of retrieval results
- The procedure can go through several iterations
- Also can capture user's evolving information needs
- Often seen as an extension of "query-by-example"

Relevance Feedback Example



Relevance Feedback: Example 2

(a) Query: New space satellite applications

(b)

\+ 1. 0.539, 08/13/91, NASA Hasn't Scrapped Imaging Spectrometer

\+ 2. 0.533, 07/09/91, NASA Scratches Environment Gear From Satellite Plan

3. 0.528, 04/04/90, Science Panel Backs NASA Satellite Plan, But Urges Launches of Smaller Probes

4. 0.526, 09/09/91, A NASA Satellite Project Accomplishes Incredible Feat: Staying Within Budget

5. 0.525, 07/24/90, Scientist Who Exposed Global Warming Proposes Satellites for Climate Research

6. 0.524, 08/22/90, Report Provides Support for the Critics Of Using Big Satellites to Study Climate

7. 0.516, 04/13/87, Arianespace Receives Satellite Launch Pact From Telesat Canada

\+ 8. 0.509, 12/02/87, Telecommunications Tale of Two Companies

Relevance Feedback: Example 2

- Term extracted from relevant docs for expanding the query

(c) 2.074 new 15.106 space
30.816 satellite 5.660 application
5.991 nasa 5.196 eos
4.196 launch 3.972 aster
3.516 instrument 3.446 arianespace
3.004 bundespost 2.806 ss
2.790 rocket 2.053 scientist
2.003 broadcast 1.172 earth
0.836 oil 0.646 measure

- So expanded query is (a) + (c)

Relevance Feedback: Example 2

- New and better search results
  - Docs with * are identified relevant docs

(d)

* 1. 0.513, 07/09/91, NASA Scratches Environment Gear From Satellite Plan

* 2. 0.500, 08/13/91, NASA Hasn't Scrapped Imaging Spectrometer

3. 0.493, 08/07/89, When the Pentagon Launches a Secret Satellite, Space Sleuths Do Some Spy Work of Their Own

4. 0.493, 07/31/89, NASA Uses 'Warm' Superconductors For Fast Circuit

* 5. 0.492, 12/02/87, Telecommunications Tale of Two Companies

6. 0.491, 07/09/91, Soviets May Adapt Parts of SS-20 Missile For Commercial Use

7. 0.490, 07/12/88, Gaping Gap: Pentagon Lags in Race To Match the Soviets In Rocket Launchers

8. 0.490, 06/14/90, Rescue of Satellite By Space Agency To Cost $90 Million

Assumptions of Relevance Feedback

- Assumptions:
  - You may not know what you're looking for, but you'll know when you see it
  - Relevant docs and non-relevant can be identified by the user
  - Relevant docs would use similar terms, and non-relevant docs would not use the terms
- Other assumptions:
  - Users like to provide such feedback,
  - we can obtain reliable feedbacks from users

Types of Relevance Feedback

- Interactive relevance feedback: feedback information obtained from the user
  - Explicit relevance feedback
    - users explicitly mark relevant and irrelevant documents in the search results
    - We have shown the examples
  - Implicit relevance feedback
    - system attempts to infer user intentions based on observable behavior
- Blind relevance feedback or pseudo relevance feedback
  - feedback in absence of any evidence, explicit or otherwise
  - System assumes that the top ranked documents as relevant docs

Implicit Feedback

- Users are often reluctant to provide relevance judgments
  - Take time to do the judgments
  - User may not know which documents are relevant
- Question: can we gather feedback information without asking the user to do explicit judgments?
- Idea: infer feedbacks from observed user behaviors

Any problem of this?

Observable Behavior

Behavior Category	Minimum Scope: Segment	Object	Class
Examine	View Listen Scroll Find Query	Select	Browse
Retain	Print	Bookmark Save Delete Purchase Email	Subscribe
Reference	Copy-and-paste Quote	Forward Reply Link Cite	
Annotate	Mark up	Rate Publish	Organize
Create	Type Edit	Author	

Kelly, D. and Teevan, J. 2003. Implicit feedback for inferring user preference: a bibliography. *SIGIR Forum* 37.2 (Sep. 2003), 18-28

Pseudo Relevance Feedback

- Also called blind relevance feedback
  - Avoid obtain user explicit or implicit feedback information
  - Use top returned N docs in the initial result for RF
  - Assume all of them are relevant

Any problem of this?

Does Pseudo RF Work?

- A study by Jimmy Lin
- Retrieval engine: Indri
- Test collection: TREC, topics 301-450
- Procedure:
  - Used topic description as query to generate initial hit list
  - Selected top 20 terms from top 20 hits using *tf.idf*
  - Added these terms to the original query

Pseudo RF Example

Number: 303
Title: Hubble Telescope Achievements

Description:
Identify positive accomplishments of the Hubble telescope since it was launched in 1991.

Narrative:
Documents are relevant that show the Hubble telescope has produced new data, better quality data than previously available, data that has increased human knowledge of the universe, or data that has led to disproving previously existing theories or hypotheses. Documents limited to the shortcomings of the telescope would be irrelevant. Details of repairs or modifications to the telescope without reference to positive achievements would not be relevant.

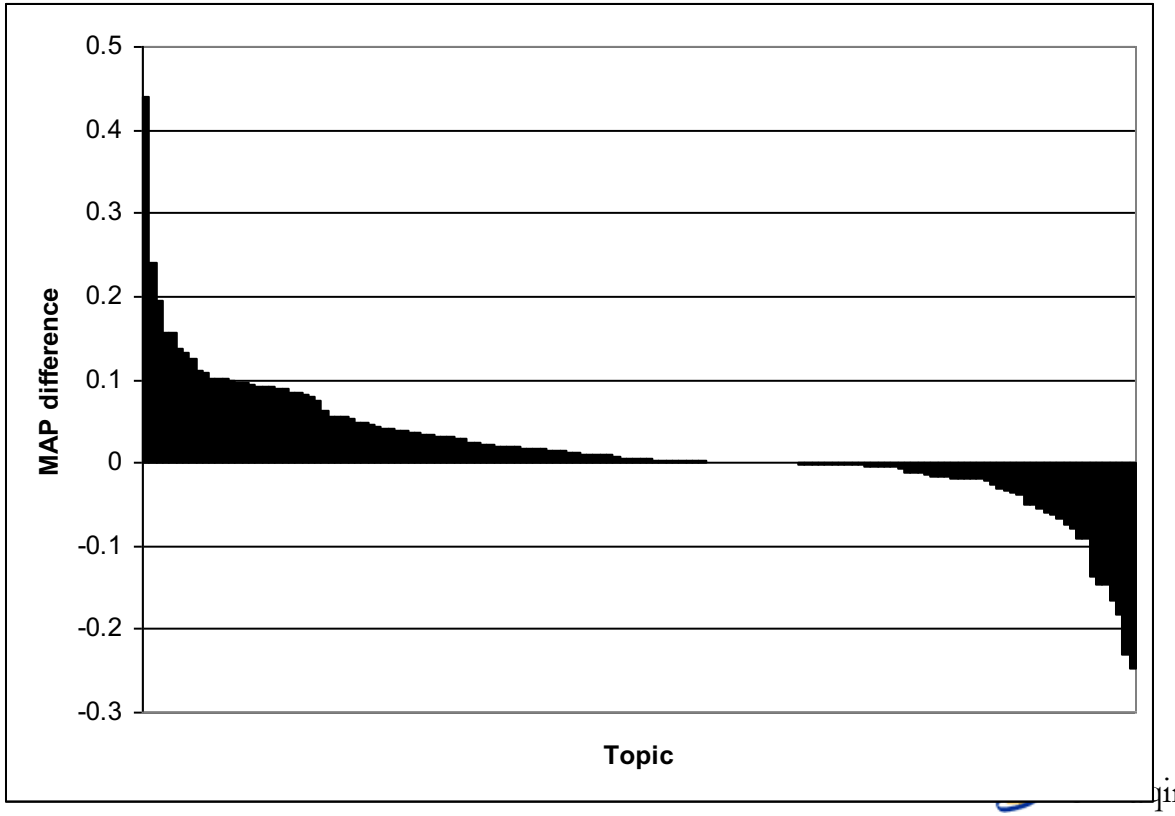
Terms added

telescope	1041.33984032195
hubble	573.896477205696
space	354.090789112131
nasa	346.475671454331
ultraviolet	242.588034029191
shuttle	230.448255669841
mirror	184.794966339329
telescopes	155.290920607708
earth	148.865466409231
discovery	146.718067628756
orbit	142.597040178043
flaw	141.832019493907
scientists	132.384677410089
launch	116.322861618261
stars	116.205713485691
universe	114.705686405825
mirrors	113.677943638299
light	113.59717006967
optical	106.198288687586
species	103.555123536418

Experiment Results

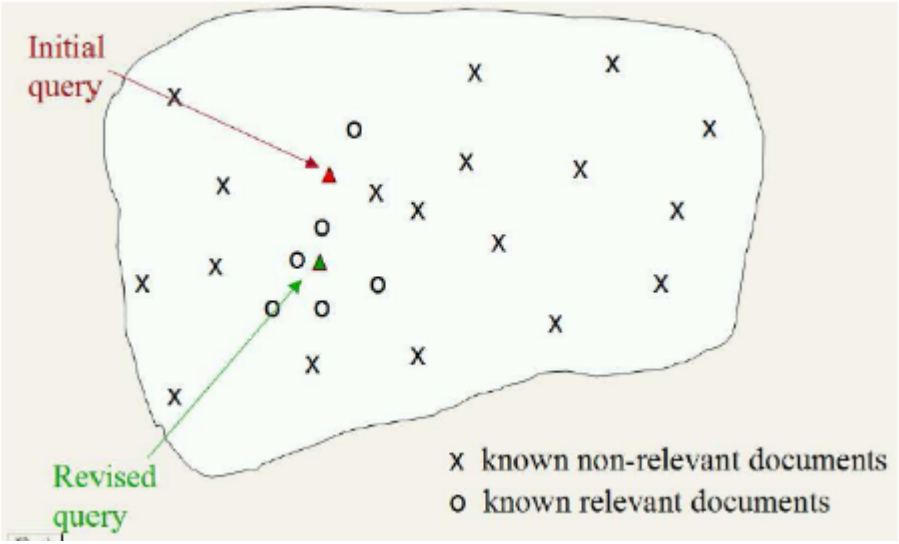
	MAP	R-Precision
No feedback	0.1591	0.2022
With feedback	0.1806 (+13.5%)	0.2222 (+9.9%)

Blind relevance feedback doesn't always help!



How to use Relevance Feedback?

- Assume that there is an optimal query
  - relevance feedback helps to bring user's query closer to the optimal one



- How?
  - Term reweighting:
    - boost weights of terms from re
  - Query expansion:
    - Add terms from relevant documents to the query

Relevance Feedback in Vector Space Model

- Rocchio Algorithm $\vec{q}_m = \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}}\vec{d}_j$
  - Often alpha = 1, Beta = 0.75, Gamma = 0.15
- Ide modifications
  - Ide: Beta and Gamma are 1, no normalization on Dr and Dnr

$$\vec{q}_m = \alpha\,\vec{q} + \beta\sum_{\forall \vec{d}_j \in D_r}\vec{d}_j - \gamma\sum_{\forall \vec{d}_j \in D_n}\vec{d}_j$$

  - Ide dec-hi: Beta and Gamma are 1, no normalization on Dr and Dnr, negative result only consider the highest ranked non-relevant doc

$$\vec{q}_m = \alpha\,\vec{q} + \beta\sum_{\forall \vec{d}_j \in D_r}\vec{d}_j - \gamma\ max_{non\text{-}relevant}(\vec{d}_j)$$

    - Ide dec-hi is the most effective when there are a few relevant docs

Rocchio in Example

$$\text{query vector} = \alpha \cdot \text{original query vector} + \beta \cdot \text{positive feedback vector} - \gamma \cdot \text{negative feedback vector}$$

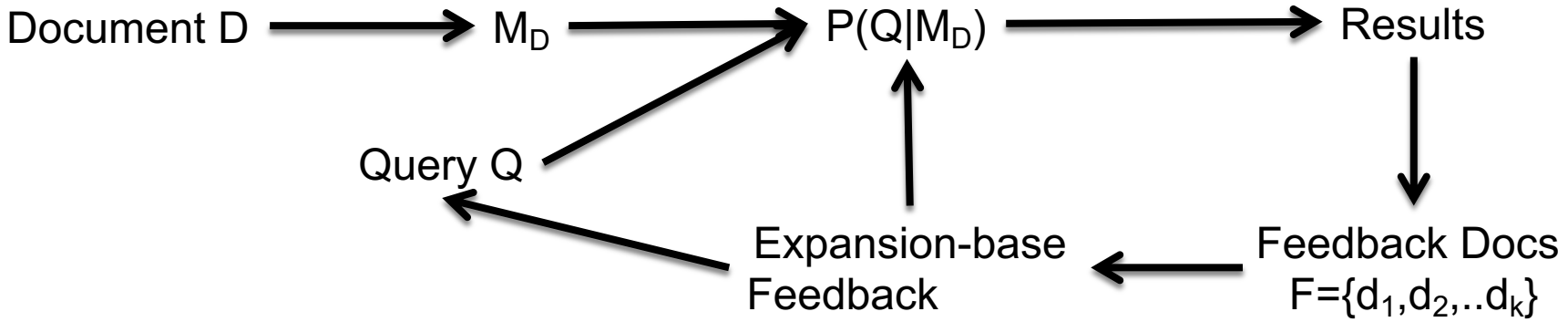
Typically, $\gamma < \beta$

Original query	0	4	0	8	0	0	$\alpha = 1.0$	0	4	0	8	0	0
Positive Feedback	8	4	8	0	0	2	$\beta = 0.5$	4	2	4	0	0	1
Negative feedback	0	0	4	4	0	1	$\gamma = 0.25$	0	0	1	1	0	.25
New query								4	6	3	7	0	.75

(+) (-)

Relevance Feedback in Language Models

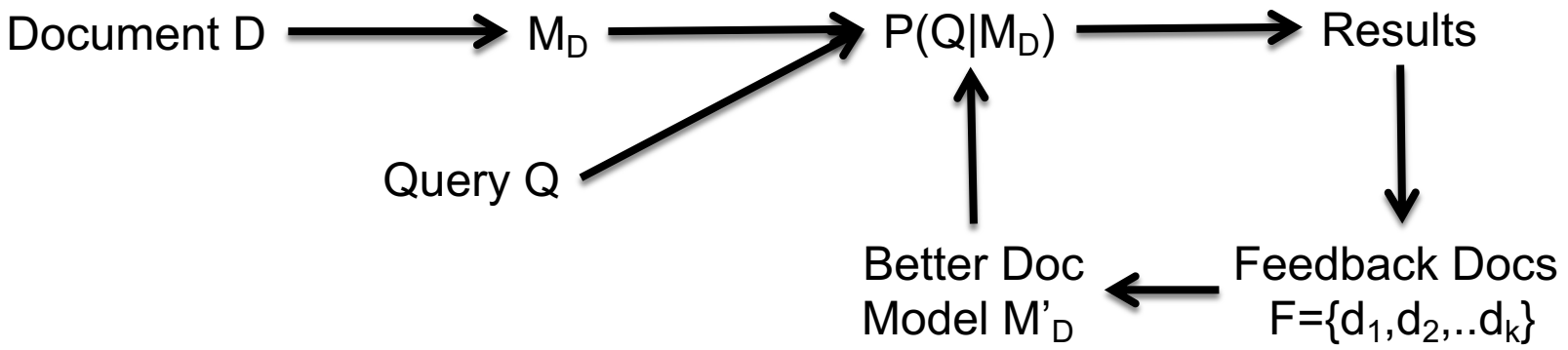
- Expansion-based Feedback in Query likelihood LMs



- Select new query terms from the feedback documents
  - Such as using TF-IDF or BM25 to select top weight terms
  - So the new terms are $\{nq_1, nq_2, \ldots, nq_n\}$
- Apply new query terms into new query
  - Combine with original query $Q' = \{q_1, q_2, .. q_m, nq_1, nq_2, \ldots, nq_n\}$, or
  - Simply viewed as the new query $Q' = \{nq_1, nq_2, \ldots, nq_n\}$

Relevance Feedback in Language Models

- Model-Interpolation Feedback in Query likelihood LMs



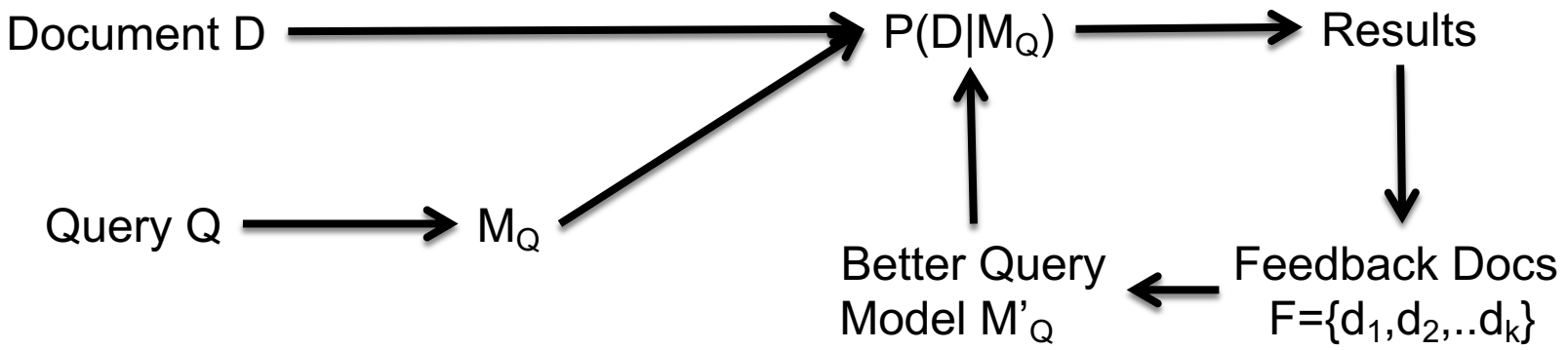
- Maybe hard to generate a query model, but can interpolate a better document model

$$P(Q|M'_D)=\alpha P(Q|M_D)+(1-\alpha)P(Q|F)$$

- We still can use JM smoothing or Dirochlet Prior smoothing for $P(Q|M_D)$

Relevance Feedback in Language Models

- Model-Interpolation Feedback in Document likelihood LMs



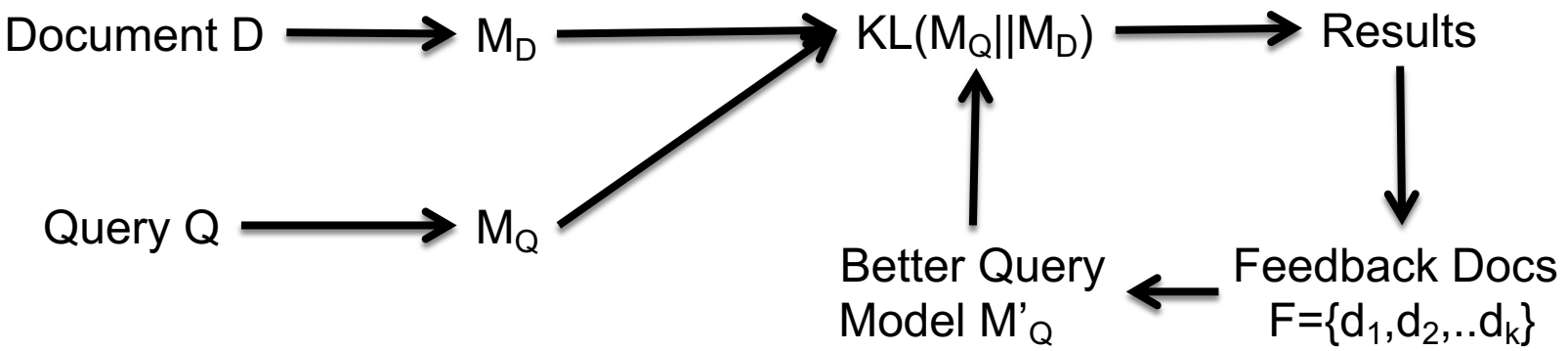
- Based on the feedback documents, we can estimate a better query model through interpolation

$$P(D|M'_Q)=\alpha P(D|M_Q)+(1-\alpha)P(D|F)$$

- We still can use JM smoothing or Dirochlet Prior smoothing for $P(D|M_Q)$

Relevance Feedback in Language Models

- Model-Interpolation Feedback in Model Comparison LMs



- Based on the feedback documents, we can estimate a better query model through interpolation

$$M'_Q=\alpha M_Q+(1-\alpha)M_F$$

- We still can use JM smoothing or Dirochlet Prior smoothing for both M_Q, and sometimes for M_F