Arama Motoru Gelistirme Dongusu: Siralamayi Ogrenme ve Bilgiye Erisimin Degerlendirilmesi

Retrieval Effectiveness and Learning to Rank

EMINE YILMAZ

Professor and Turing Fellow
University College London

Research Consultant
Microsoft Research

LEARNING TO RANK (LTR)

"… the task to automatically construct a ranking model using training data, such that the model can sort new objects according to their degrees of relevance, preference, or importance."

- Liu [2009]

LTR models represent a rankable item—e.g., a document—given some context—e.g., a user-issued query—as a numerical vector $\vec{x} \in \mathbb{R}^n$

The ranking model $f: \vec{x} \rightarrow \mathbb{R}$ is trained to map the vector to a real-valued score such that relevant items are scored higher.

Tie-Yan Liu. Learning to rank for information retrieval. Foundation and Trends in Information Retrieval, 2009.

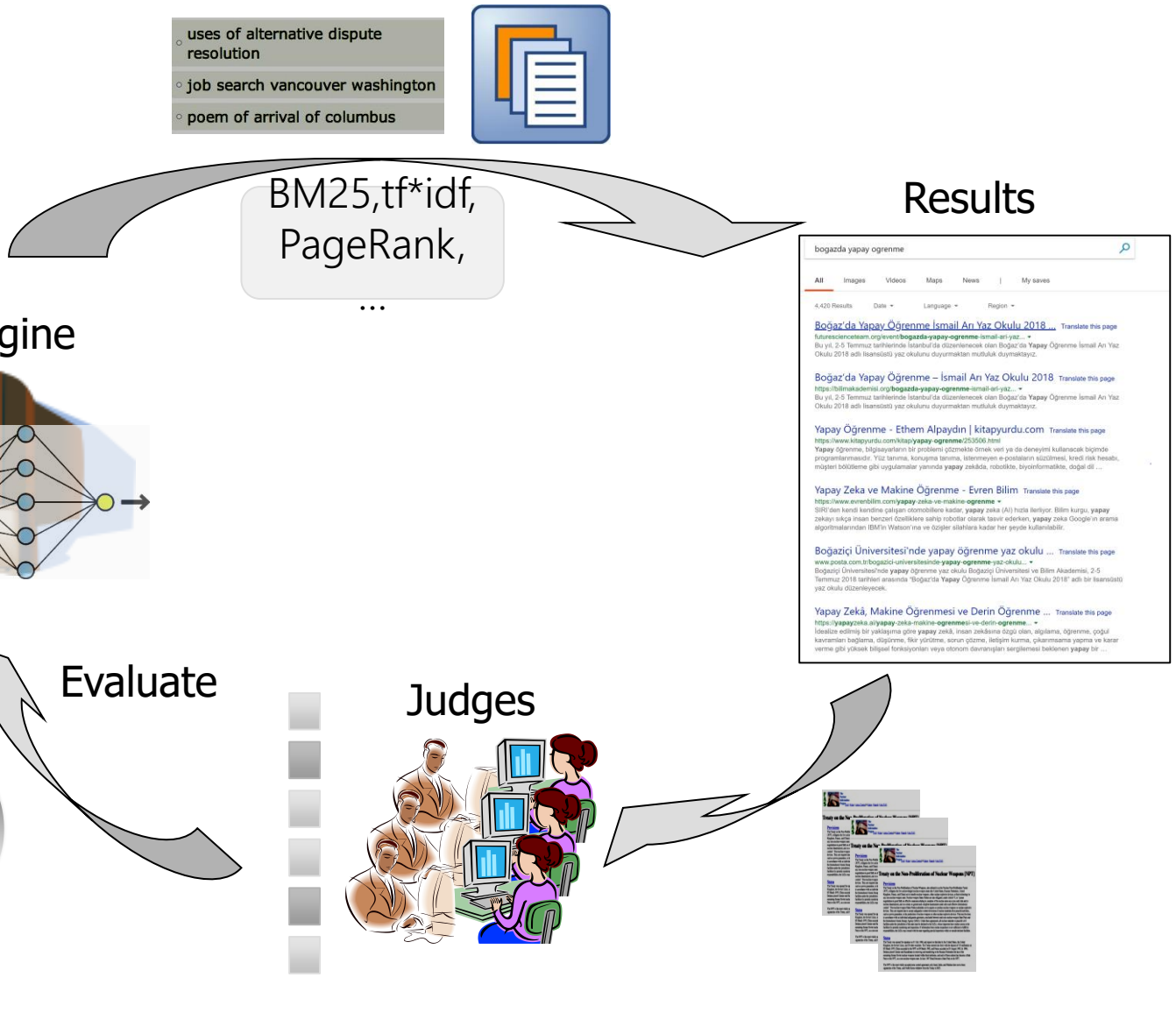
APPLICATIONS OF LEARNING TO RANK

- Search (Document Search, Entity Search, etc.)
- Recommender Systems (Collaborative Filtering)
- Question Answering
- Document Summarization
- Opinion Mining
- Machine Translation …

APPLICATIONS OF LEARNING TO RANK

- **Search (Document Search, Entity Search, etc.)**
- Recommender Systems (Collaborative Filtering)
- Question Answering
- Document Summarization
- Opinion Mining
- Machine Translation …

LEARNING TO RANK



FEATURES

Traditional learning to rank models employ hand-crafted features that encode insights about the problem

They can often be categorized as:

Query-independent or static features

e.g., incoming link count, page-rank score

Query-dependent or dynamic features

e.g., BM25

FEATURES

Traditional learning to rank models employ hand-crafted features that encode insights about the problem

More semantic representations in the recent years

They can often be categorized as:

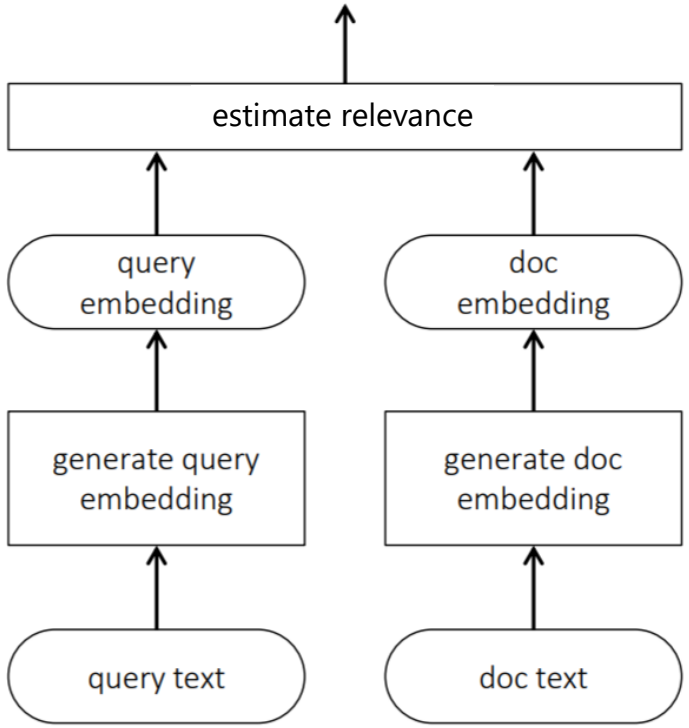
Query-independent or static features

e.g., incoming link count, page-rank score

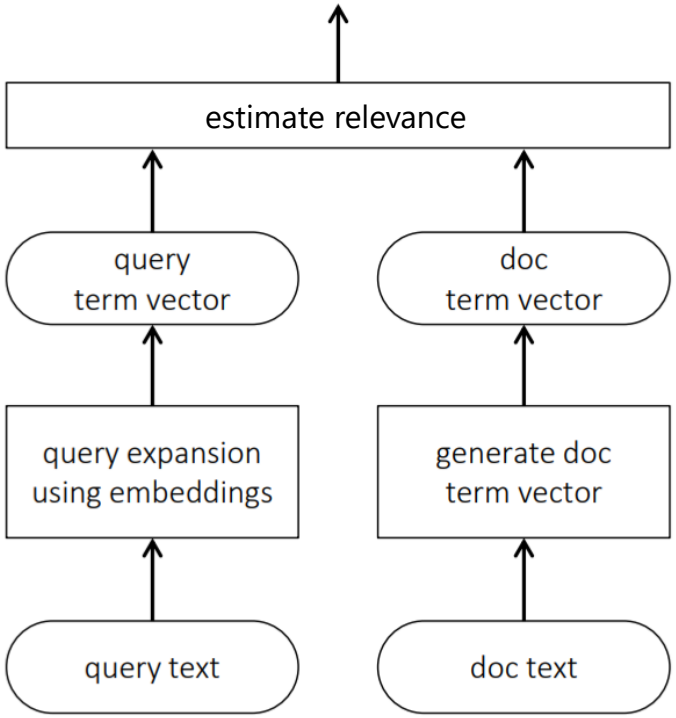
Query-dependent or dynamic features

e.g., BM25

TERM EMBEDDINGS FOR SEARCH



Compare query and document directly in the embedding space



Use embeddings to generate suitable query expansions

APPROACHES

Liu [2009] categorizes different LTR approaches based on training objectives:

Pointwise approach

Relevance label $y_{q,d}$ is a number—derived from binary or graded human judgments or implicit user feedback (e.g., CTR). Typically, a regression or classification model is trained to predict $y_{q,d}$ given $\vec{x}_{q,d}$.

Pairwise approach

Pairwise preference between documents for a query ($d_i \succ d_j$ w.r.t. q) as label. Reduces to binary classification to predict more relevant document.

Listwise approach (Modern Systems)

Directly optimize for rank-based metrics evaluating user satisfaction (more to be discussed later)

Tie-Yan Liu. Learning to rank for information retrieval. Foundation and Trends in Information Retrieval, 2009.

EVALUATION METRICS: PRECISION VS. RECALL

Retrieved list

1	R
2	N
3	R
4	N
5	N
6	R
7	N
8	N
9	N
10	R
⋮	⋮

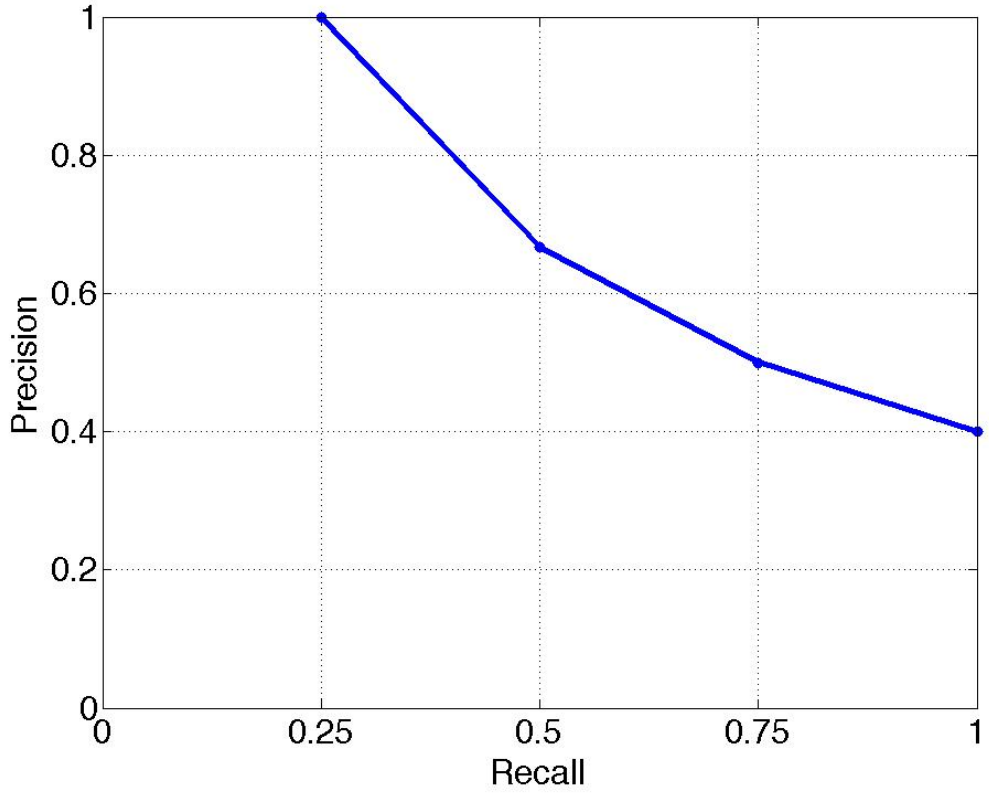
$$PC(k) = \frac{|\text{relevants up to rank k}|}{k}$$

$$Recall(k) = \frac{|\text{relevants up to rank k}|}{|\text{relevants in the query}|}$$

VISUALIZING RETRIEVAL PERFORMANCE: PRECISION-RECALL CURVES

List:

R
N
R
N
N
R
N
N
N
R

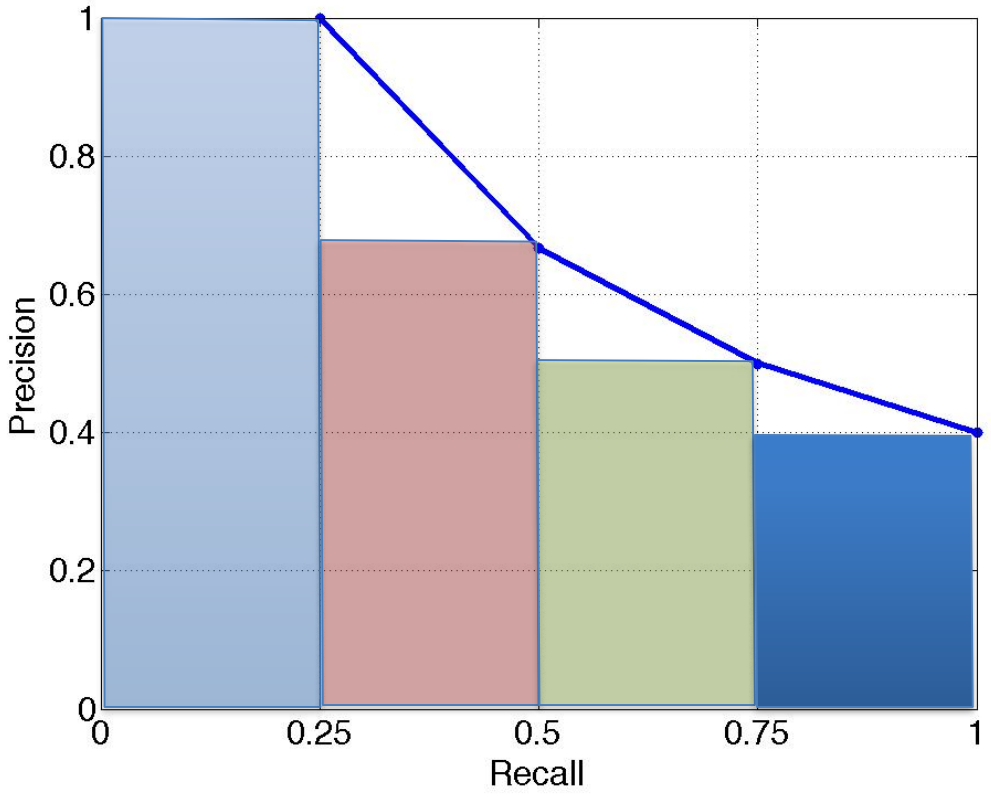


EVALUATION METRICS: AVERAGE PRECISION

List:

R
N
R
N
N
R
N
N
N
R

$\Leftrightarrow R$



EVALUATION METRICS: NDCG

- Some documents more relevant than others
  - User receives some gain from each document
- Discount gain based on rank

$$DCG = \sum_{r=1}^{N} G(r) \cdot D(r)$$

$$G(r) = rel(r), 2^{rel(r)}, \ldots$$

$$D(r) = \frac{1}{\log_b(r)}, \frac{1}{r}, \ldots$$

- Normalized discounted cumulative gain

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

EVALUATION METRICS

- Two categories
  - User-oriented metrics
    - *PC(k), NDCG(k)*
  - System-oriented metrics
    - *AP, NDCG*

APPROACHES

Pointwise approach

Relevance label $y_{q,d}$ is a number—derived from binary or graded human judgments or implicit user feedback (e.g., CTR). Typically, a regression or classification model is trained to predict $y_{q,d}$ given $\vec{x}_{q,d}$.

Pairwise approach

Pairwise preference between documents for a query ($d_i \succ d_j$ w.r.t. q) as label. Reduces to binary classification to predict more relevant document.

Listwise approach (Modern Systems)

Directly optimize for rank-based metric, such as NDCG—difficult because these metrics are often not differentiable w.r.t. model parameters.

Tie-Yan Liu. Learning to rank for information retrieval. Foundation and Trends in Information Retrieval, 2009.

PAIRWISE OBJECTIVES

RankNet loss

Pairwise loss function proposed by Burges et al. [2005]—an industry favourite [Burges, 2015]

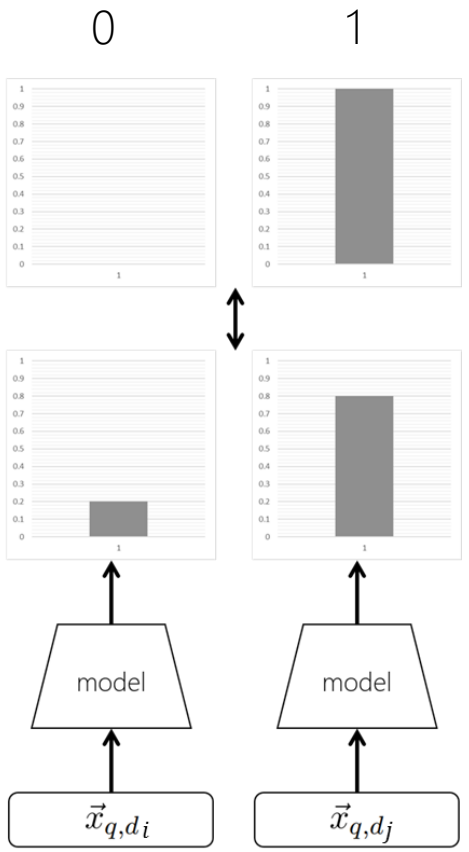
Predicted probabilities: $p_{ij} = p(s_i > s_j) \equiv \frac{1}{1+e^{-\gamma.(s_i - s_j)}}$

Desired probabilities: $\bar{p}_{ij} = 1$ and $\bar{p}_{ji} = 0$

Computing cross-entropy between p and $\bar{p}$

$$\mathcal{L}_{RankNet} = -\bar{p}_{ij}.log(p_{ij}) - \bar{p}_{ji}.log(p_{ji})$$

Use neural network as the model, and gradient descent as the algorithm, to optimize the cross-entropy loss.



Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In ICML, 2005.
Chris Burges. RankNet: A ranking retrospective. https://www.microsoft.com/en-us/research/blog/ranknet-a-ranking-retrospective/. 2015.

LISTWISE OBJECTIVES

Blue: relevant Gray: non-relevant

NDCG higher for left but pairwise errors less for right

Due to strong position-based discounting in IR measures, errors at higher ranks are much more problematic than at lower ranks

But listwise metrics are non-continuous and non-differentiable

[Burges, 2010]

Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. Learning, 2010.

LISTWISE OBJECTIVES: LAMBDARANK

Burges et al. [2010] make two observations:

1. To train a model we don't need the costs themselves, only the gradients (of the costs w.r.t model scores)
2. It is desired that the gradient be bigger for pairs of documents that produces a bigger impact in NDCG by swapping positions

LambdaRank loss

Multiply actual gradients with the change in NDCG by swapping the rank positions of the two documents

$$\lambda_{LambdaRank} = \lambda_{RankNet} \cdot |\Delta NDCG|$$

Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. Learning, 2010.

LISTWISE OBJECTIVES: LAMBDARANK

- Empirically shown to optimize the objective metric
- Most current learning to rank models based on different variations of the idea
- Winner of the Yahoo! Learning to Rank Challenge

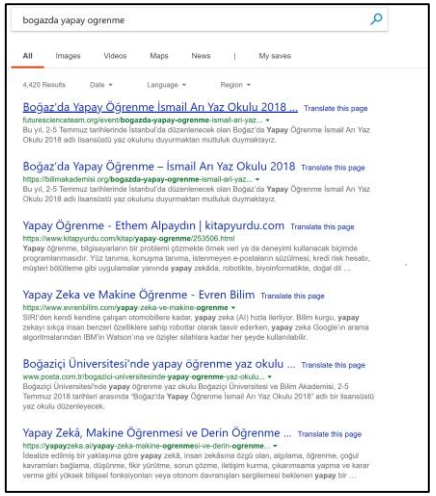
Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. Learning, 2010.

LEARNING TO RANK

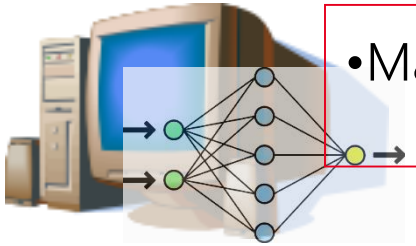
- uses of alternative dispute resolution
- job search vancouver washington
- poem of arrival of columbus

BM25,tf*idf, PageRank, …

Results

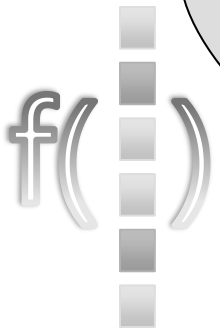


Search Engine

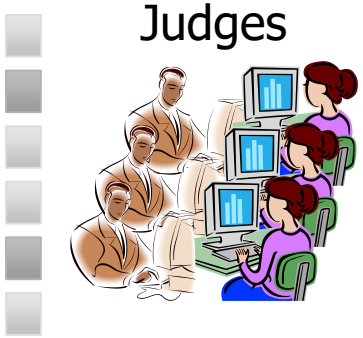


- Many different evaluation metrics
  - Which metric?

Evaluate



Judges



OPTIMIZING FOR A METRIC

- Empirical Risk Minimization
  - “X” evaluates user satisfaction
  - Optimize for “X”

- A common misconception!
  - Informative vs. uninformative metrics

TRAINING WITH MORE INFORMATION

- Train for a more informative metric
  - “X” : user satisfaction

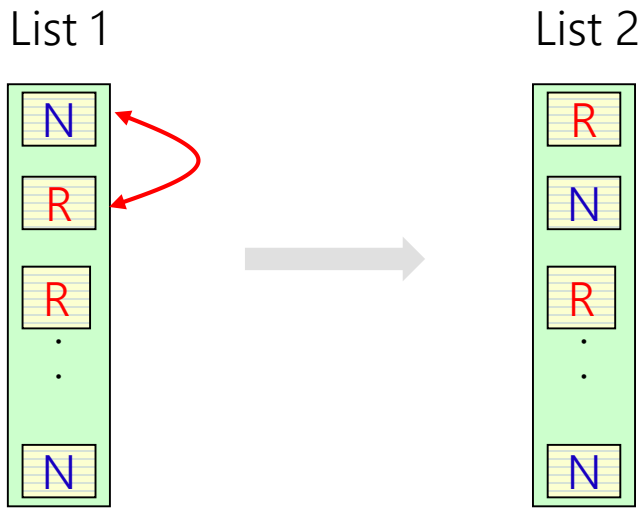
TRAINING WITH MORE INFORMATION

- Train for a more informative metric
  - “X” : user satisfaction
  - “Y”: more *informative* than “X”

TRAINING WITH MORE INFORMATION

- Train for a more informative metric
  - “X” : user satisfaction
  - “Y”: more *informative* than “X”
  - Train for “Y”!
    - Better test set “X” than training for “X”!

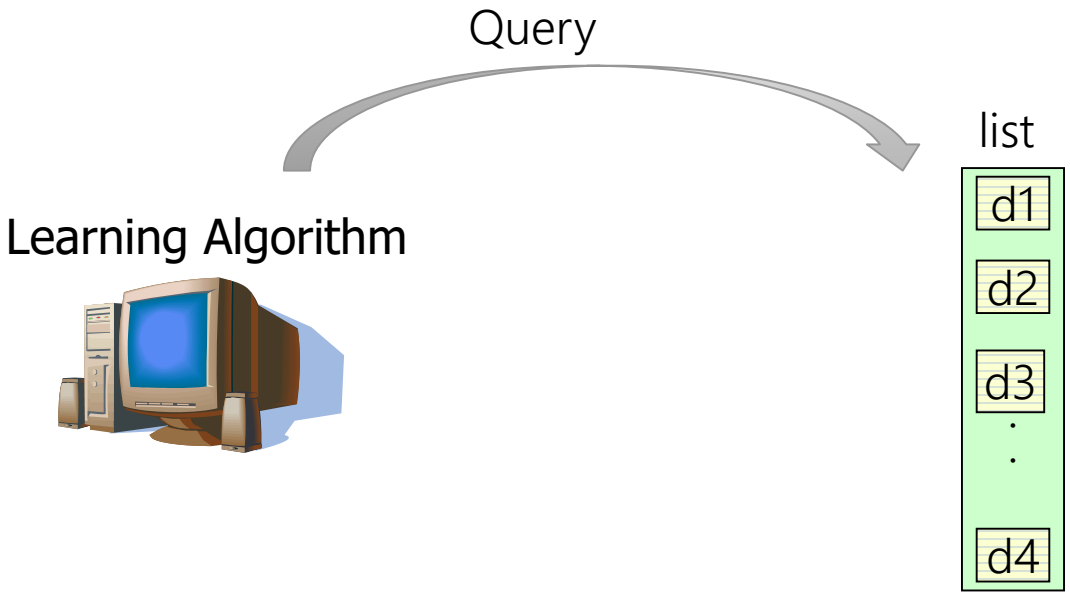
WHAT MAKES A METRIC MORE INFORMATIVE?



- Metrics respond to flips in the same way
- Some metrics may ignore some flips
  - Metrics sensitive to many flips more informative
- Metrics weight flips differently
  - Some metrics give too much weight to some flips

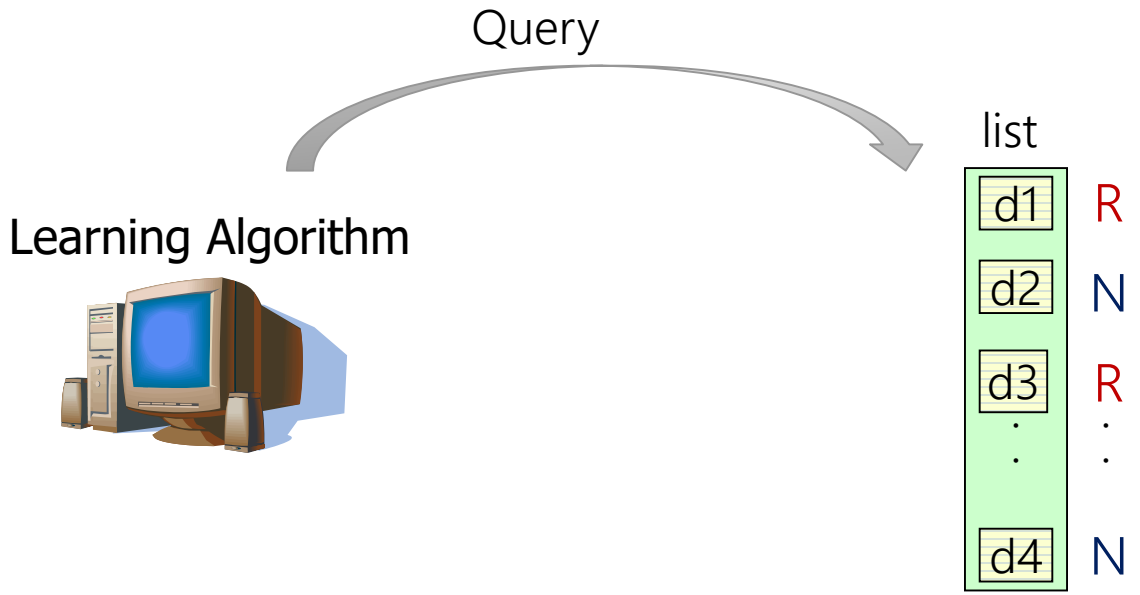
INFORMATIVENESS AND LEARNING TO RANK

- Evaluation metrics *summarize* the training data



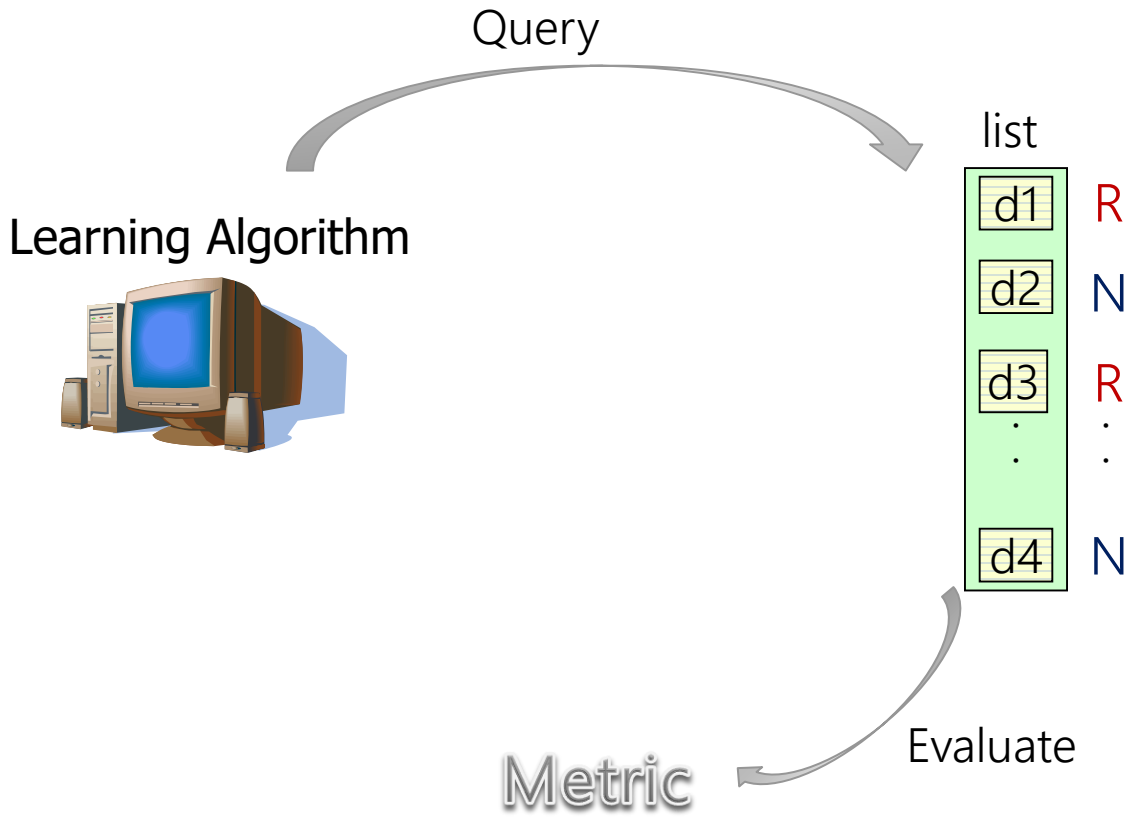
INFORMATIVENESS AND LEARNING TO RANK

- Evaluation metrics *summarize* the training data



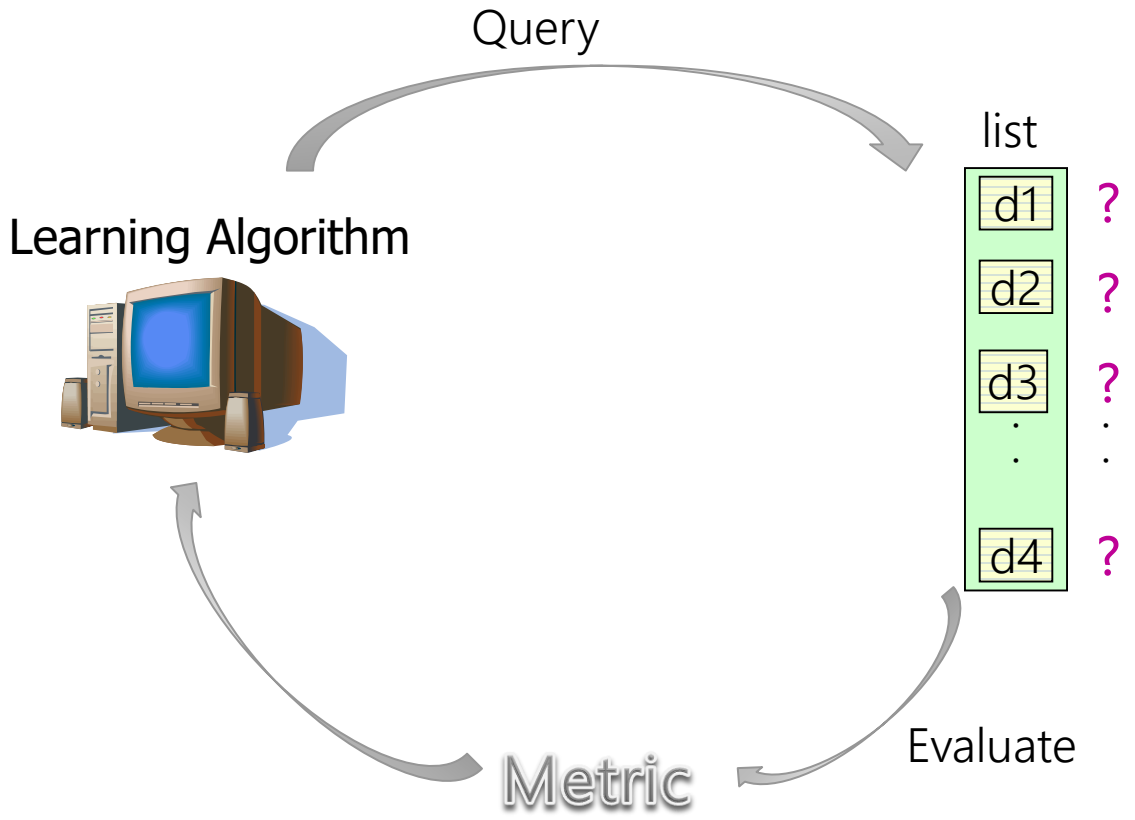
INFORMATIVENESS AND LEARNING TO RANK

- Evaluation metrics *summarize* the training data



INFORMATIVENESS AND LEARNING TO RANK

- Evaluation metrics *summarize* the training data



EVALUATION METRICS AND INFORMATIVENESS

- For the same part of the ranking
  - Some metrics insensitive to some flips at this part
- Two lists with identical PC(10) values

1	R
2	R
3	R
4	N
5	N
6	N
7	N
8	N
9	N
10	N

1	N
2	N
3	N
4	N
5	N
6	N
7	N
8	R
9	R
10	R

EVALUATION METRICS AND INFORMATIVENESS

- Some metrics ignore some parts of ranking
  - Two lists with identical PC(5) and NDCG(5) values

1	R
2	N
3	R
4	N
5	R
6	N
7	N
8	N
9	N
10	N

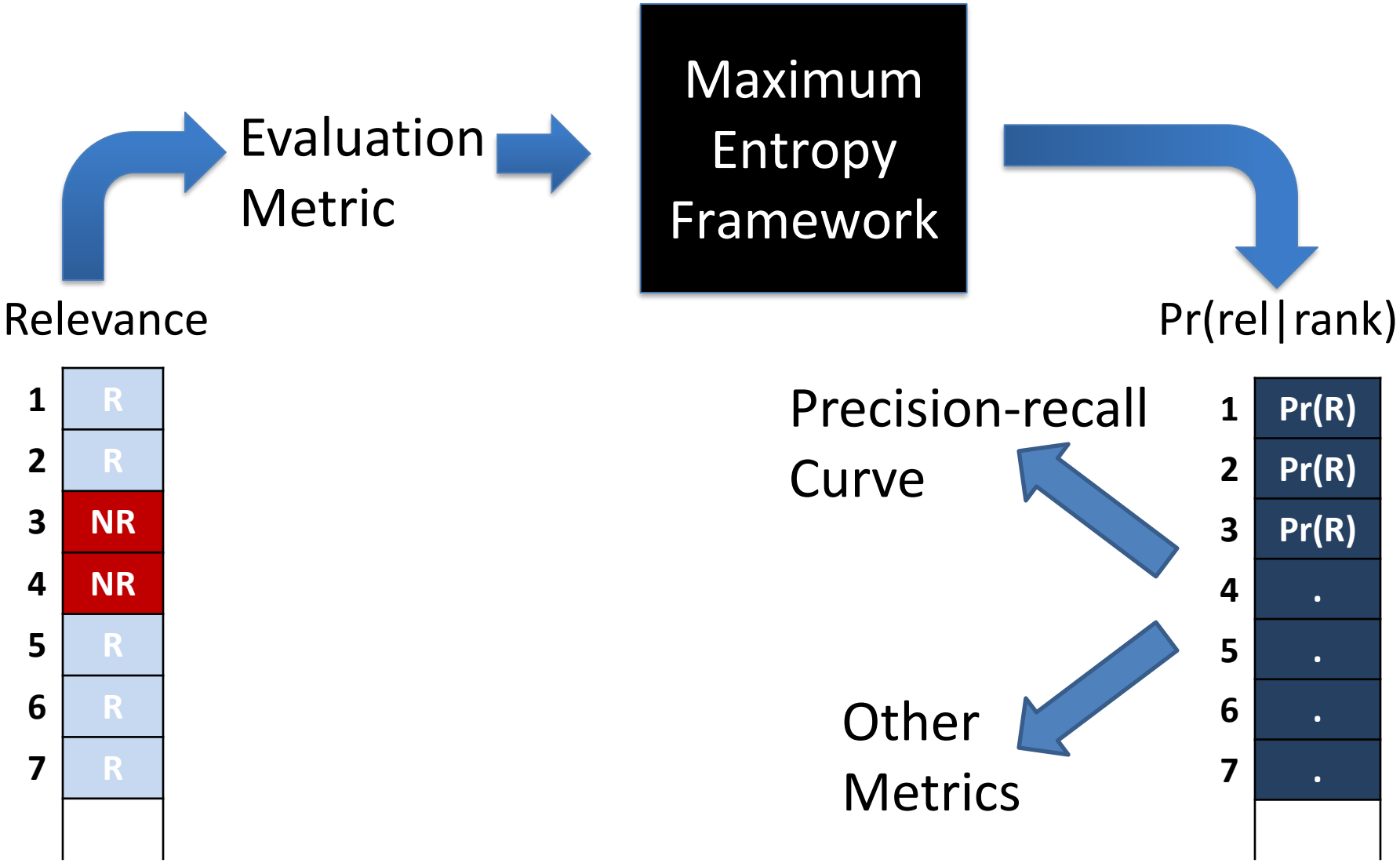
1	R
2	N
3	R
4	N
5	R
6	R
7	R
8	R
9	R
10	R

INFORMATIVENESS OF METRICS

[YILMAZ AND ROBERTSON, IRJ'10]

- Quality of a ranked list
  - Relevance of documents in the list
- How much does a metric reduce one's uncertainty in the underlying list?
  - Informative metrics: large reduction in uncertainty
  - Non-informative metrics: little or no reduction in uncertainty

INFORMATIVENESS OF METRICS

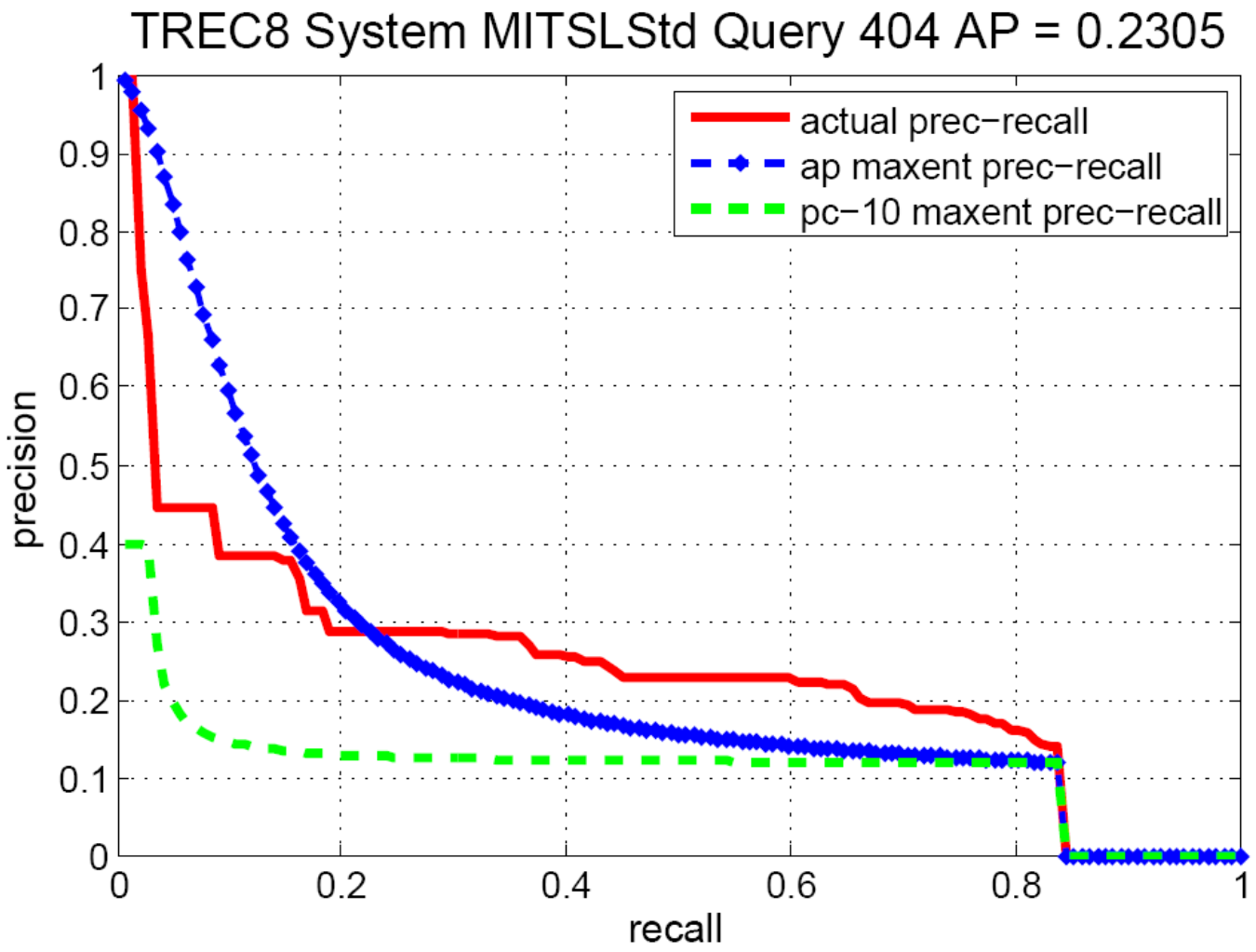


SETUP FOR AP METRIC

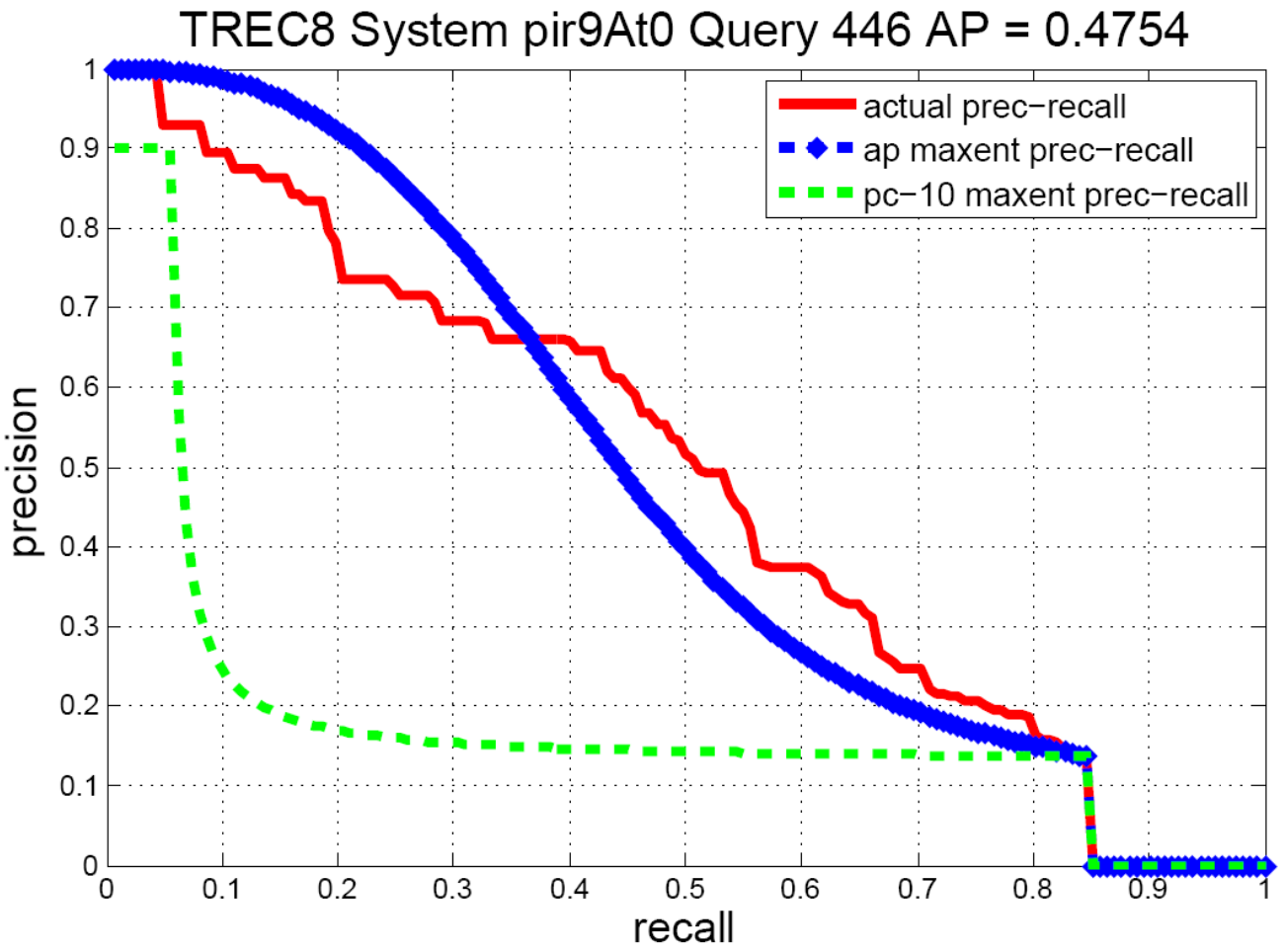
- Goal:
  - Given the average precision value (ap) of a list, infer probability of relevance of document at rank i

- Maximum entropy setup:
  - Maximize
    - $\sum_{i=1}^{N} H(p_i)$
  - Subject to
    - $E[AP] = \frac{1}{R} \sum_{i=1}^{N} \left(\frac{p_i}{i} \left(1 + \sum_{j=1}^{i-1} \right) \right) = ap$
    - $E[R] = \sum_{i=1}^{N} p_i = R$

INFORMATIVENESS OF METRICS



INFORMATIVENESS OF METRICS



WHICH EVALUATION METRIC?

- Which evaluation metric?
  - Informative Metrics: AP, NDCG
  - Less Informative Metrics: NDCG(10), PC(10)
    - NDCG(10) more informative than PC(10)

- AP vs. NDCG
  - AP more informative than NDCG regarding binary relevance of documents

INFORMATIVENESS AND LEARNING TO RANK

- Hypothesis:
  - Optimizing for a more informative metric “Y” gives better test set “X” than optimizing for “X” directly
- Learning algorithms
  - LambdaRank
  - SoftRank (Optimize for “smooth” versions of metrics)
- Evaluation Metrics
  - Informative Metrics: AP, NDCG
  - Less Informative Metrics: NDCG(10), PC(10)

TRAINING ON DIFFERENT METRICS: LAMBDARANK

Optim. Metric	TestMetric		
	AP	**PC(10)**	**NDCG(10)**
AP	**61.95**	**54.21***	61.29
PC(10)	59.99	52.73	61.81
NDCG	61.30	53.27*	**62.90***
NDCG(10)	60.82	52.77	62.37

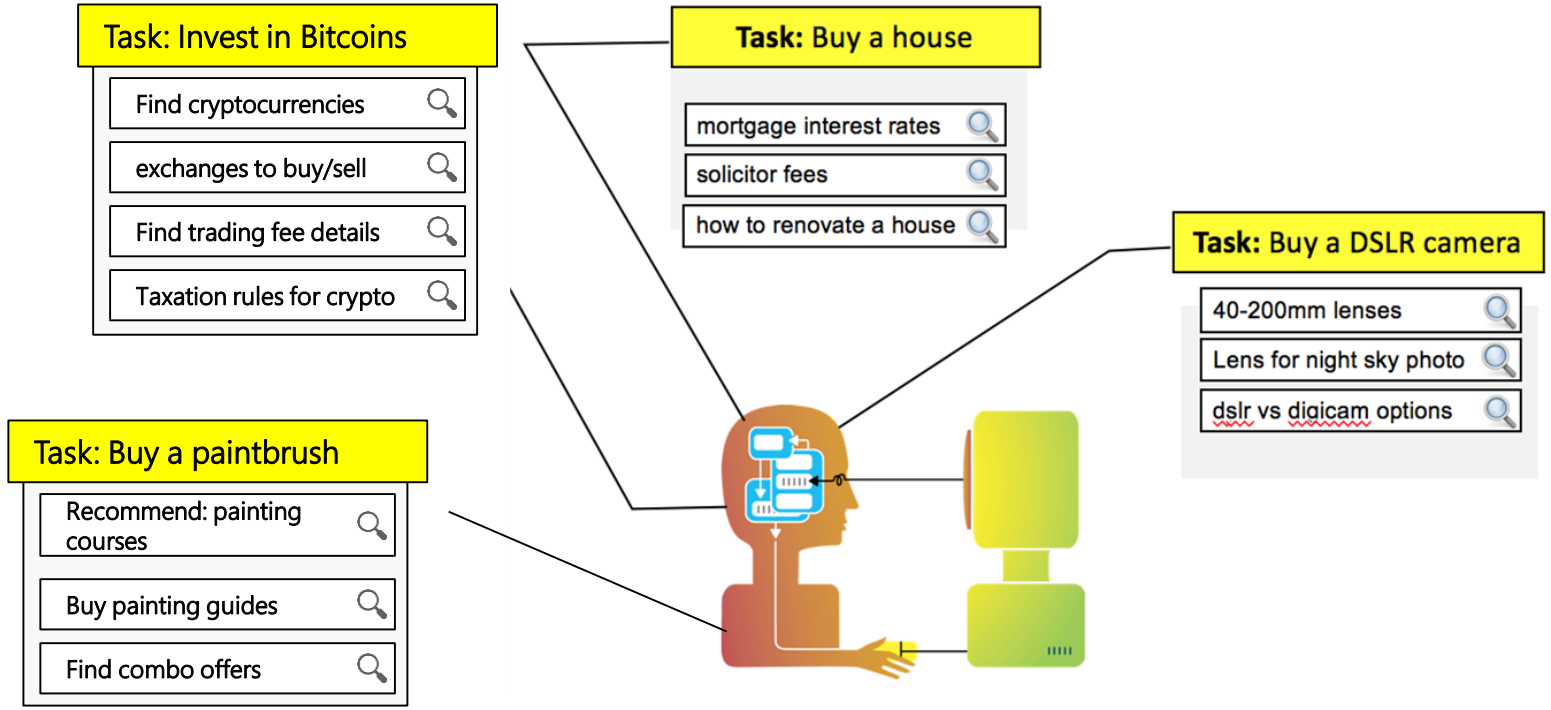
TRAINING ON DIFFERENT METRICS: SOFTRANK

Optim. Metric	TestMetric		
	AP	**PC(10)**	**NDCG(10)**
AP	62.91	54.96*	63.03*
PC(10)	62.28	54.44	62.24
NDCG	62.82	54.92*	62.98*
NDCG(10)	62.30	54.72*	62.41

SUMMARY

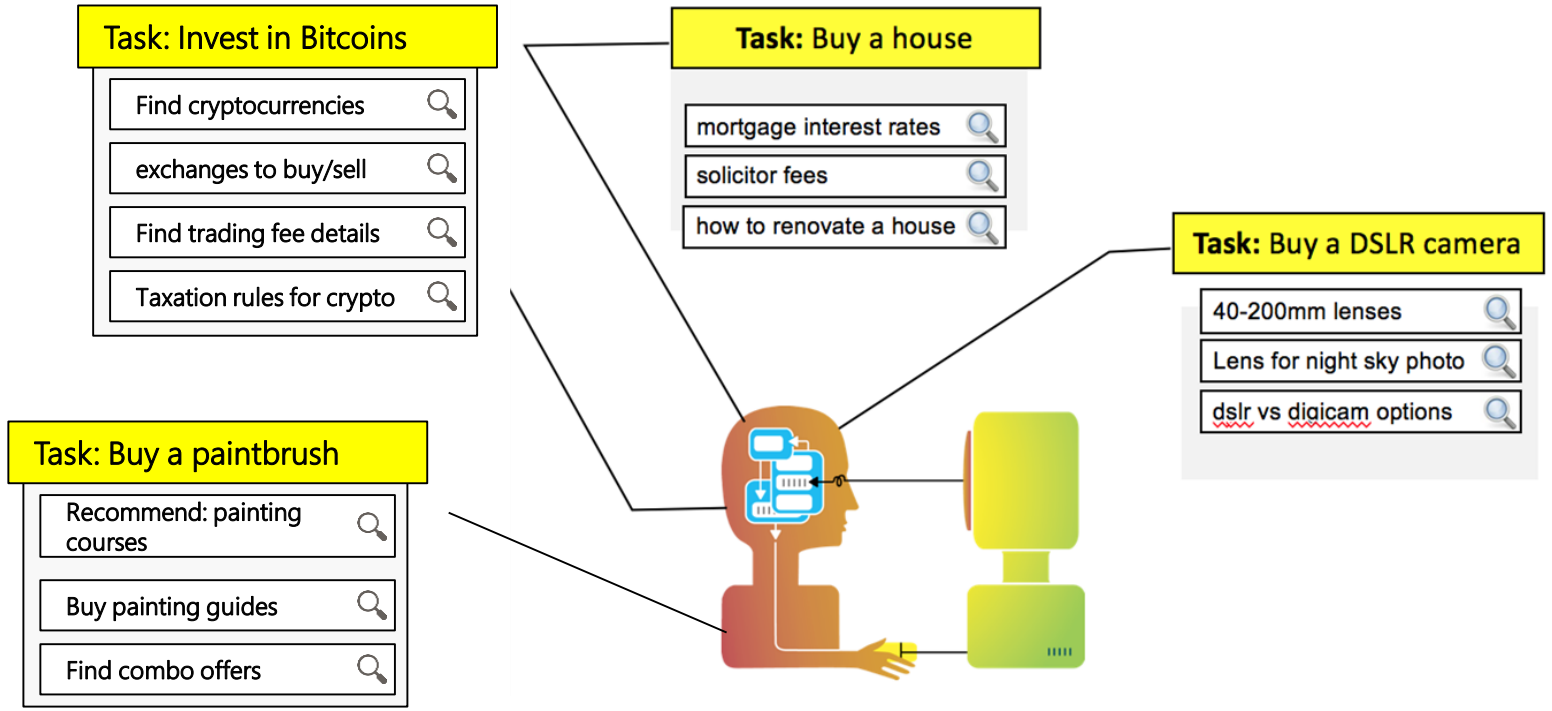
- Be careful about which metric you use!
- Optimize for informative metrics
  - Similar conclusions in classification
- Informative metric design
  - Graded Average Precision
  - What is the ultimate metric for learning to rank?

CURRENT RESEARCH: TASK BASED IR



- Users use online systems to achieve some real world tasks

CURRENT RESEARCH: TASK BASED IR



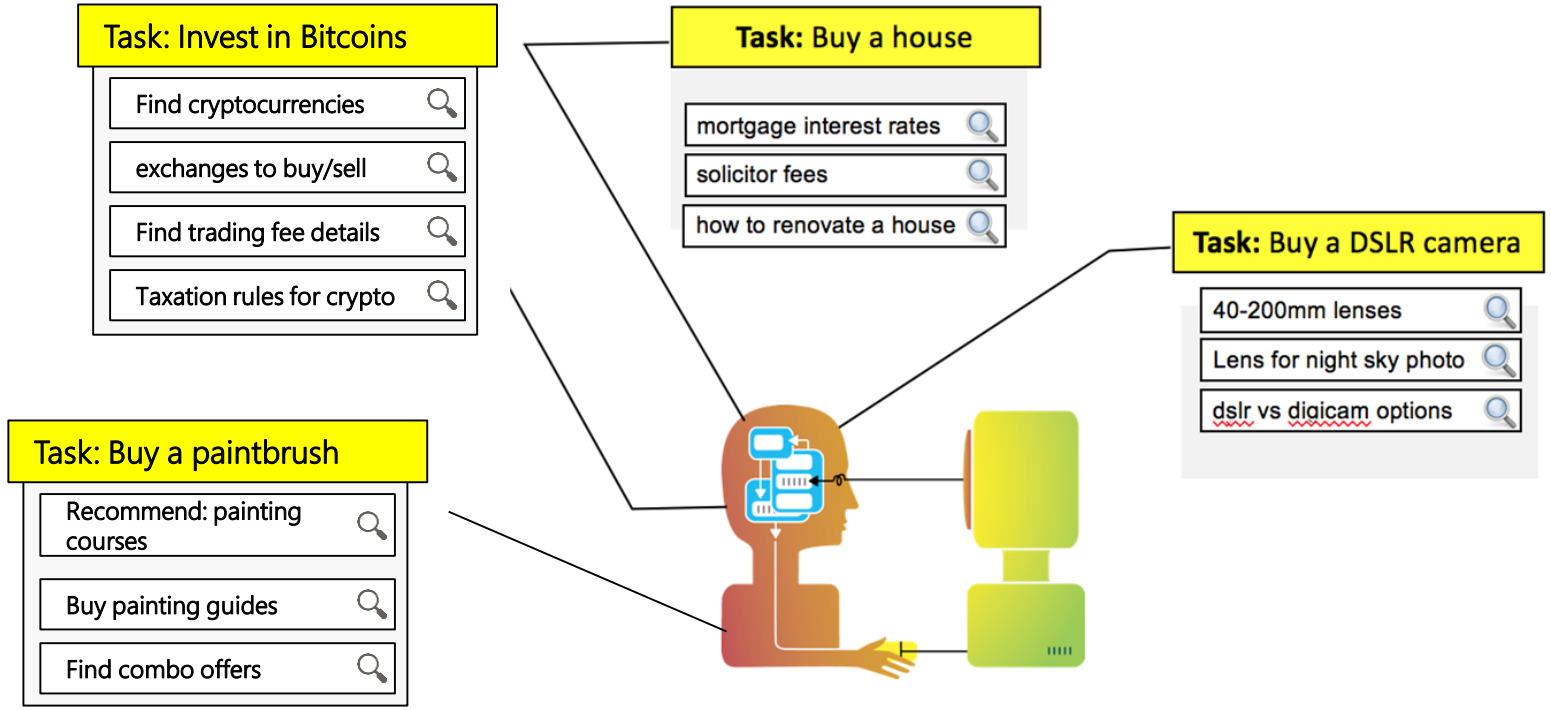
25
Page Views per task

44
Minutes per task

7
Queries per task

- Users use online systems to achieve some real world tasks
- Significant effort required using existing systems

CURRENT RESEARCH: TASK BASED IR



25
Page Views per task

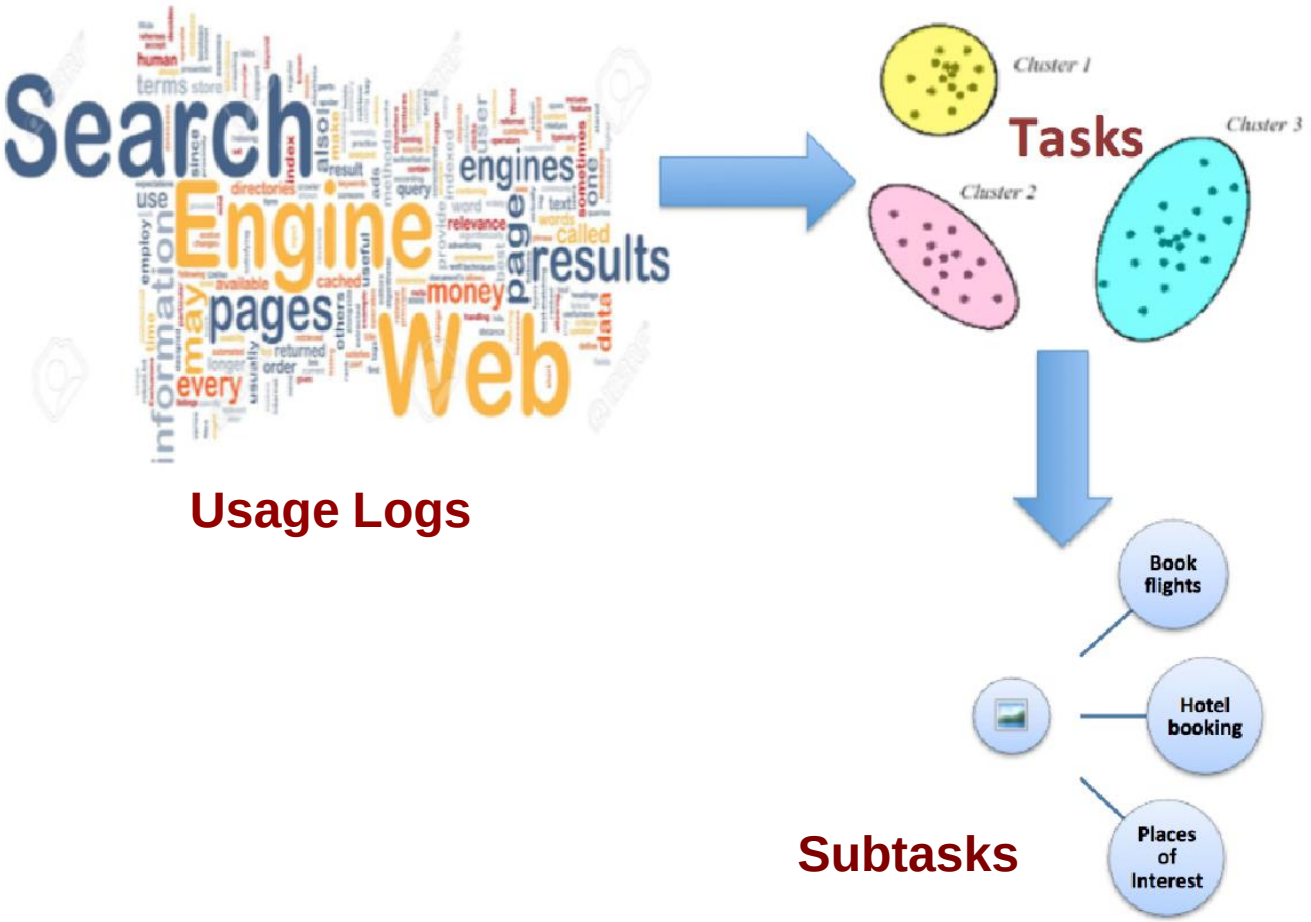
44
Minutes per task

7
Queries per task

- Devise next generation intelligent online services than can
  - **Go beyond the input from the user**
  - **Automatically detect the task the user trying to achieve**
  - **Provide the user with contextual task completion assistance**

RESEARCH CHALLENGES FOR TASK BASED IR SYSTEMS

- Task extraction/representation
- Design of task based retrieval interfaces
- Task based personalization
- Task based evaluation of retrieval systems



- Usage logs (questions asked, queries issued, pages viewed, etc.) contain information about tasks users use the online systems for

- Mine information from usage logs using machine learning techniques to infer the representations of tasks

- **Extracting Hierarchies of Search Tasks & Subtasks via a Bayesian Nonparametric Approach. R. Mehrotra and E. Yilmaz. In Proceedings of ACM SIGIR 2017.**
- **Deep Sequential Models for Task Satisfaction Prediction R. Mehrotra, E. Yilmaz et al. In Proceedings of ACM CIKM 2017.**
- **Task Embeddings: Learning Query Embeddings using Task Context. R. Mehrotra and E. Yilmaz. In Proceedings of ACM CIKM 2017.**

MORE ON CURRENT/FUTURE WORK

- Conversational IR system design and evaluation
- Stance detection (fake news detection)
- Understanding user behaviour across different devices
- AI for education
- Predicting cryptocurrency price change using resources from the web

Thank You!

Dr. Manisha Verma

Dr. Jiyin He

Bhaskar Mitra

Sahan Bulathwela

Dr. Rishabh Mehrotra

Dr. Shangsong Liang

Qiang Zhang

Andrew Burnie