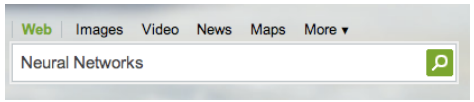
Document Selection Methodologies for Efficient and Effective Learning-to-Rank

Javed Aslam, Evangelos Kanoulas,
Virgil Pavlu, Stefan Savev, Emine Yilmaz

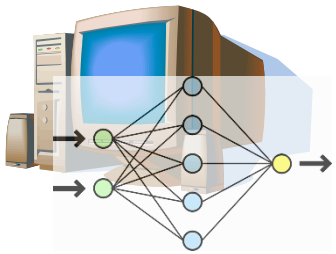
Search Engines

User's Request

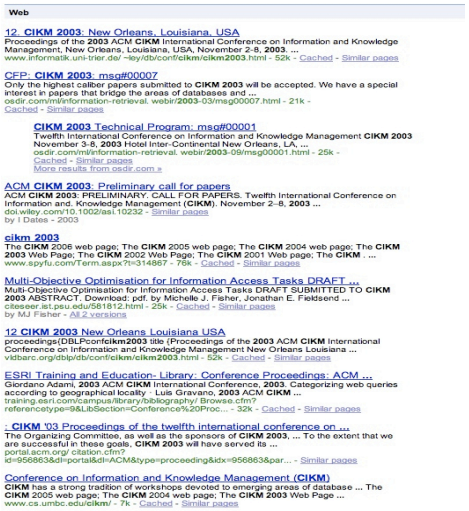


BM25,tf*idf,
PageRank, …

Search Engine



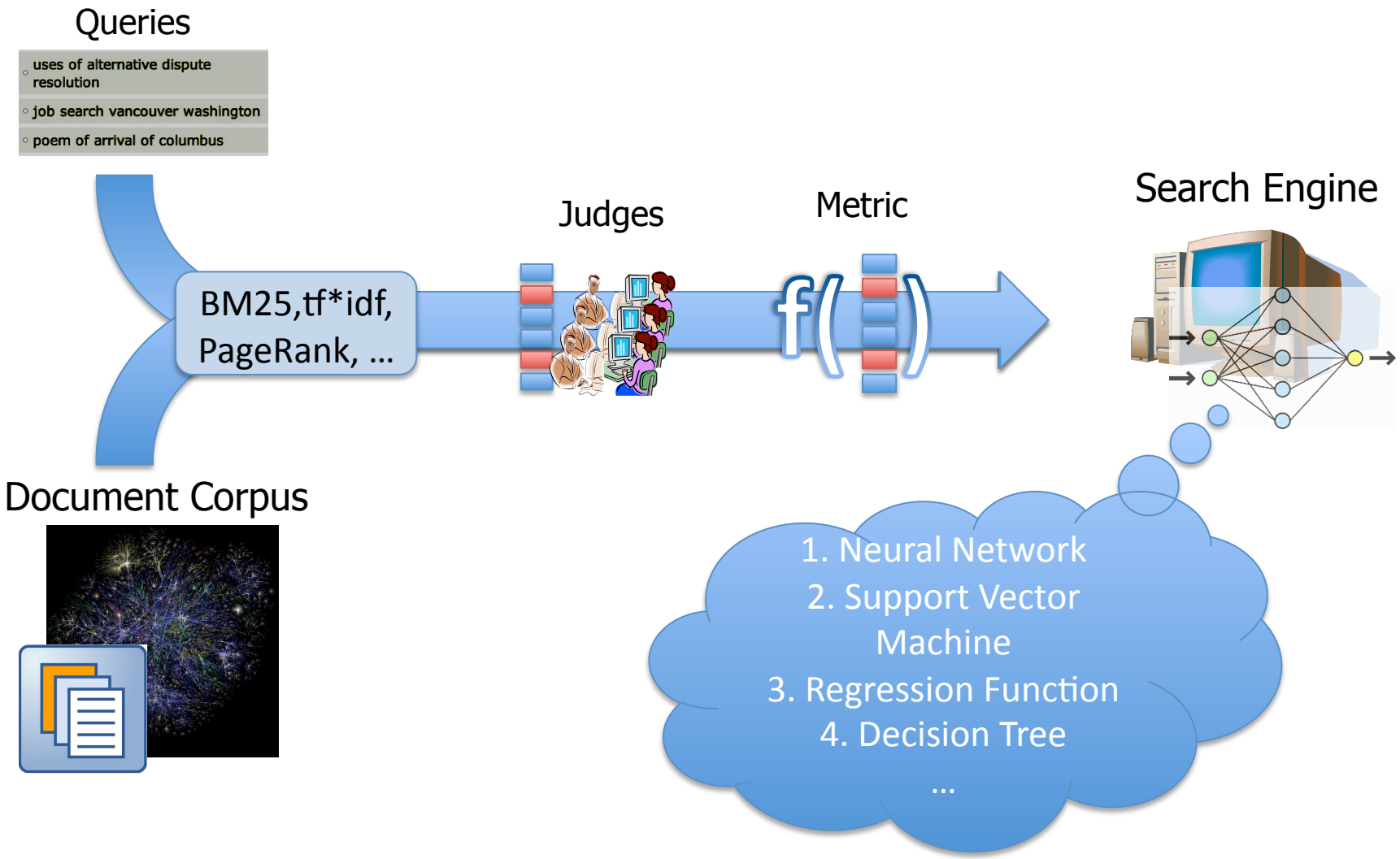
Results



Document Corpus

Hundreds of
features

Training Search Engines



Training Data Sets

- Data Collections
  - Billions of documents
  - Thousands of queries
- Ideal, in theory; infeasible, in practice…
  - Extract features from all query-document pairs
  - Judge each document with respect to each query
    - Extensive human effort
  - Train over all query-document pairs

Training Data Sets

- Train the ranking function over a subset of the complete collection
- Few queries with many document judged vs. many queries with few documents judged
  - Better to train over many queries with few judged documents [Yilmaz and Robertson '09]
- How should we select document?

Training Data Sets

- Machine Learning (Active Learning)
  - Iterative process
  - Tightly coupled with the learning algorithm
- IR Evaluation
  - Many test collections already available
  - Efficient and effective techniques to construct test collections
    - Intelligent way of selecting documents
    - Inferences of effectiveness metrics

Duality between LTR and Evaluation

- This work: Explore duality between Evaluation and Learning-to-Rank
  - Employ techniques used for efficient and effective test collection construction to construct training collections

Duality between LTR and Evaluation

- Can test collection construction methodologies be used to construct training collections?
- If yes, which one of these methodologies is better?
- What makes a training set better than the other?

Methodology

- Depth-100 pool (as the complete collection)
- Select subsets of documents from the depth-100 pool
  - Using different document selection methodologies
- Train over the different training sets
  - Using a number of learning-to-rank algorithms
- Test the performance of the resulting ranking functions
  - Five fold cross validation

Data Sets

- Data from TREC 6,7 and 8
  - Document corpus : TREC Discs 4 and 5
  - Queries : 150 queries; ad-hoc tracks
  - Relevance judgments : depth-100 pools

- Features from each query-document pair
  - 22 features; subset of LETOR features (BM25, Language Models, TF-IDF, …)

Document Selection Methodologies

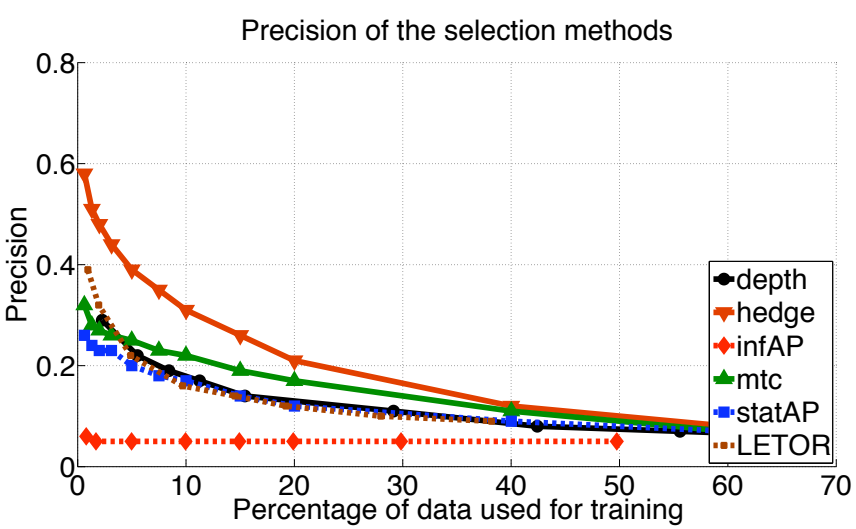
- Select subsets of documents
  - Subset size varying from 6% to 60%

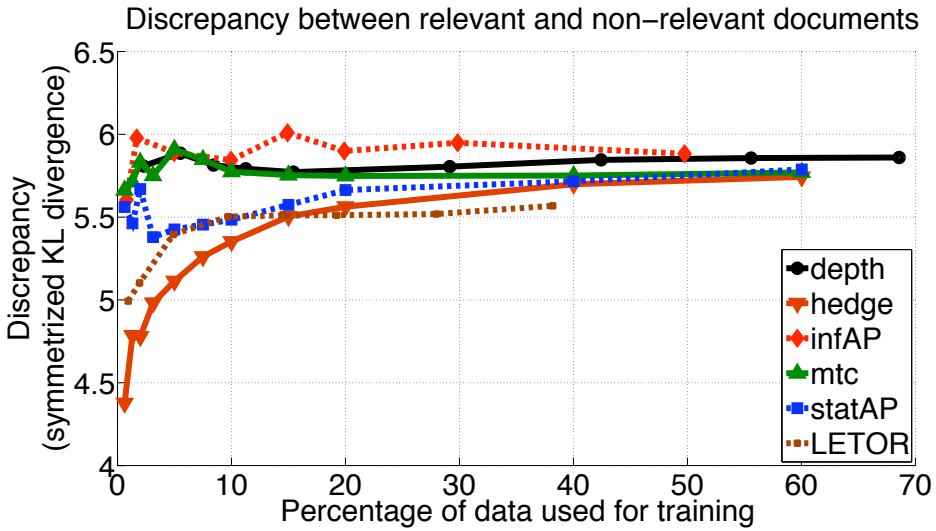
1. Depth-k pooling

2. InfAP (uniform random sampling)
3. StatAP (stratified random sampling)
4. MTC (greedy on-line algorithm)

5. LETOR (top-k by BM25; current practice)
6. Hedge (greedy on-line algorithm)

Document Selection Methodologies





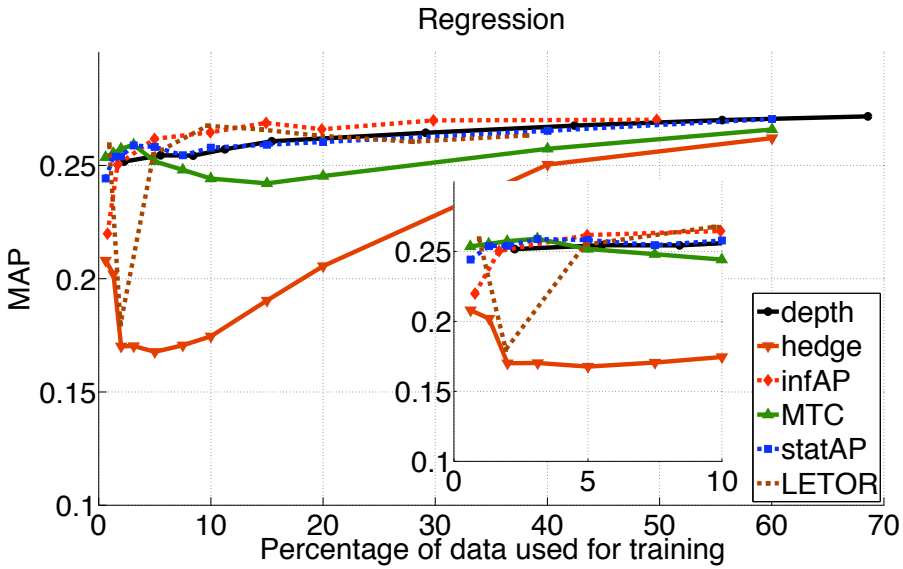
- Precision : fraction of selected documents that are relevant
- Discrepancy : symmetrized KL divergence between documents' language models

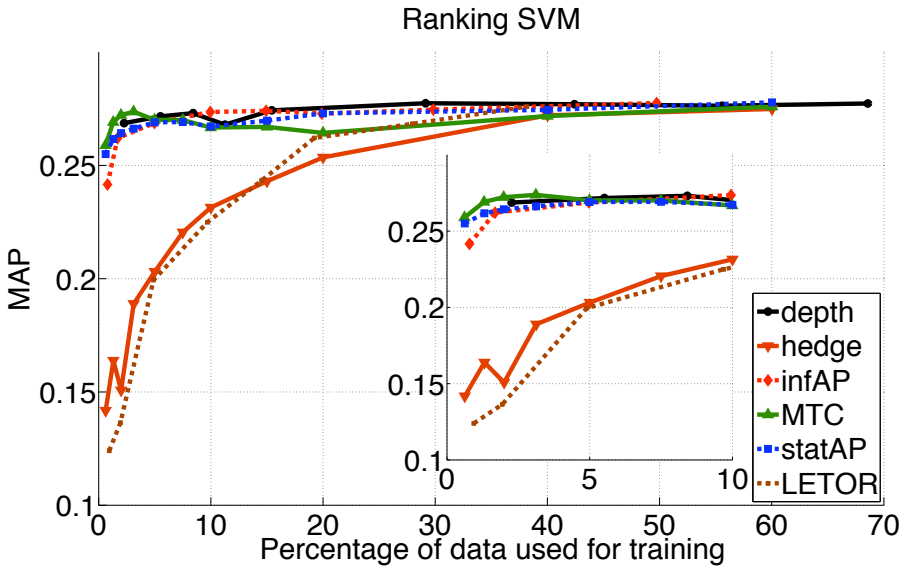
LTR Algorithms

- Train over the different data sets

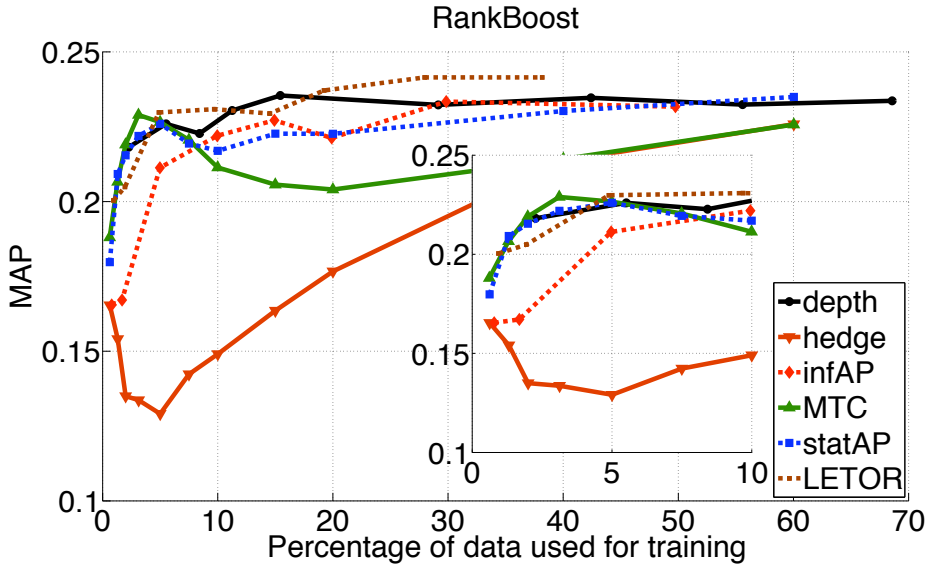
1. Regression (classification error)
2. Ranking SVM (AUC)
3. RankBoost (pairwise preferences)
4. RankNet (probability of correct order)
5. LambdaRank (nDCG)

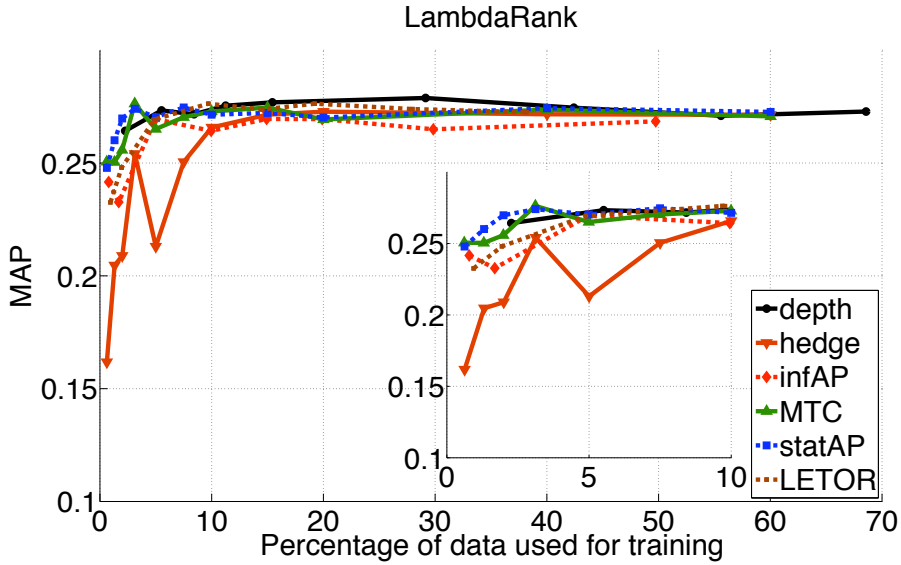
Results (1)


Regression
MAP
Percentage of data used for training
depth
hedge
infAP
MTC
statAP
LETOR

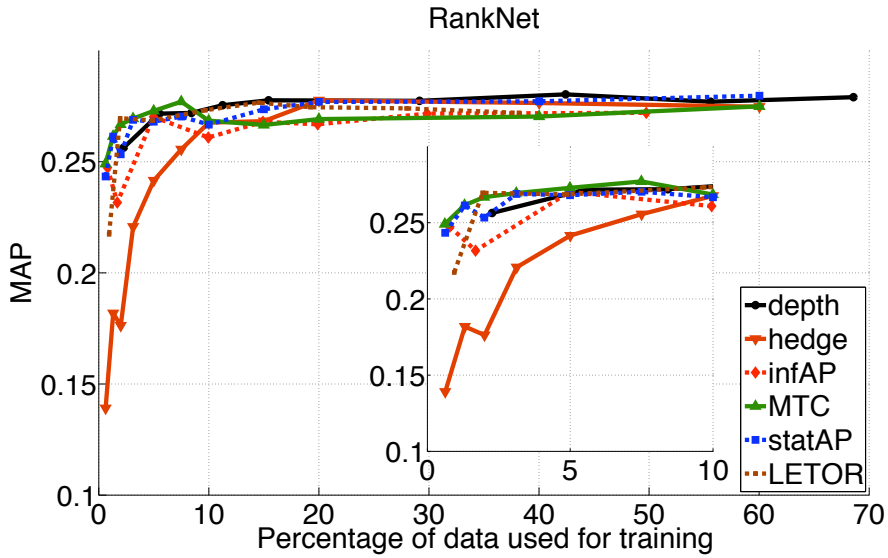

Ranking SVM
MAP
Percentage of data used for training
depth
hedge
infAP
MTC
statAP
LETOR

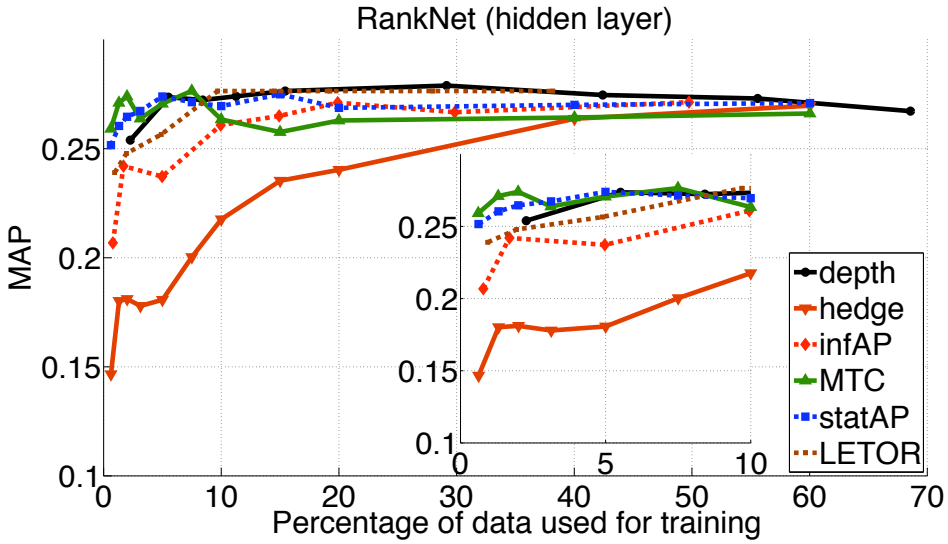
Results (2)


RankBoost
MAP
Percentage of data used for training
depth
hedge
infAP
MTC
statAP
LETOR


LambdaRank
MAP
Percentage of data used for training
depth
hedge
infAP
MTC
statAP
LETOR

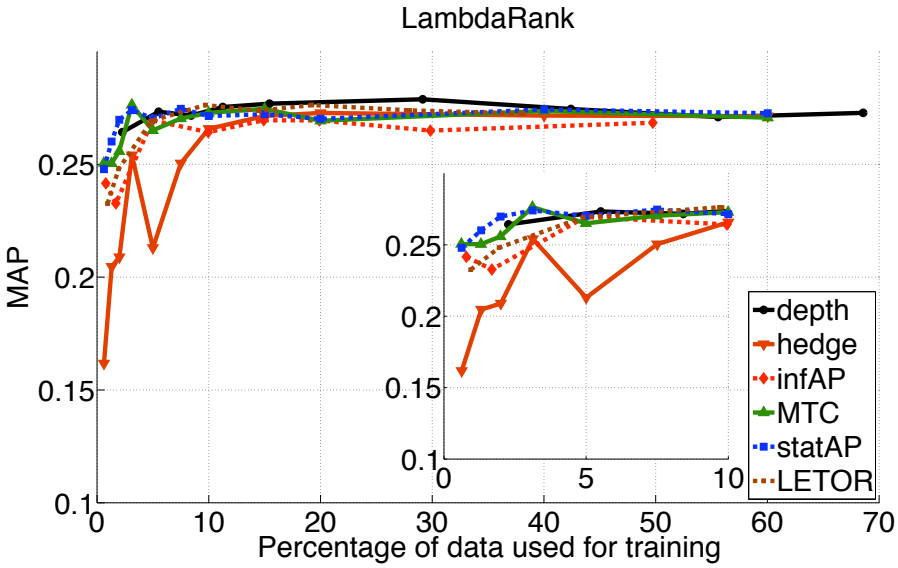
Results (3)

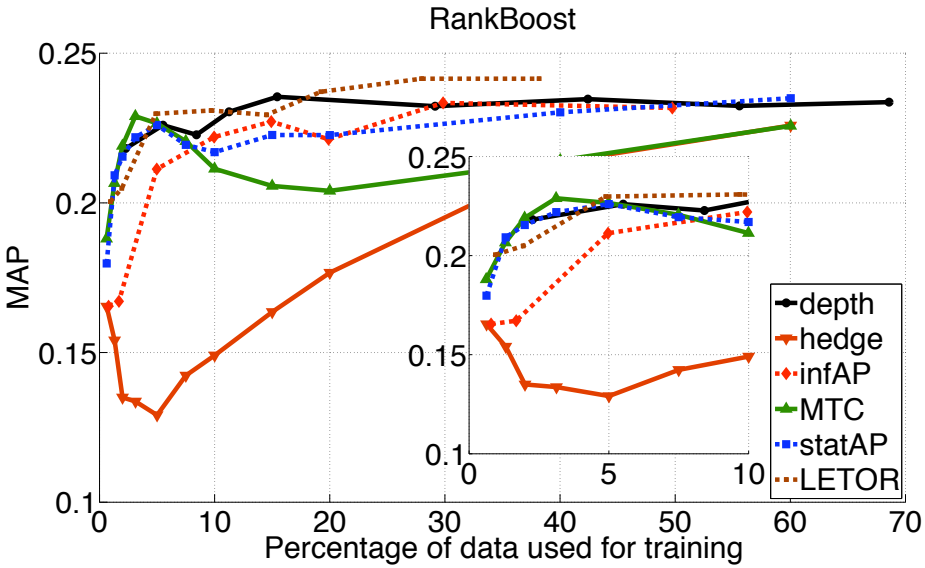

RankNet
MAP
Percentage of data used for training
depth
hedge
infAP
MTC
statAP
LETOR


RankNet (hidden layer)
MAP
Percentage of data used for training
depth
hedge
infAP
MTC
statAP
LETOR

Observations (1)

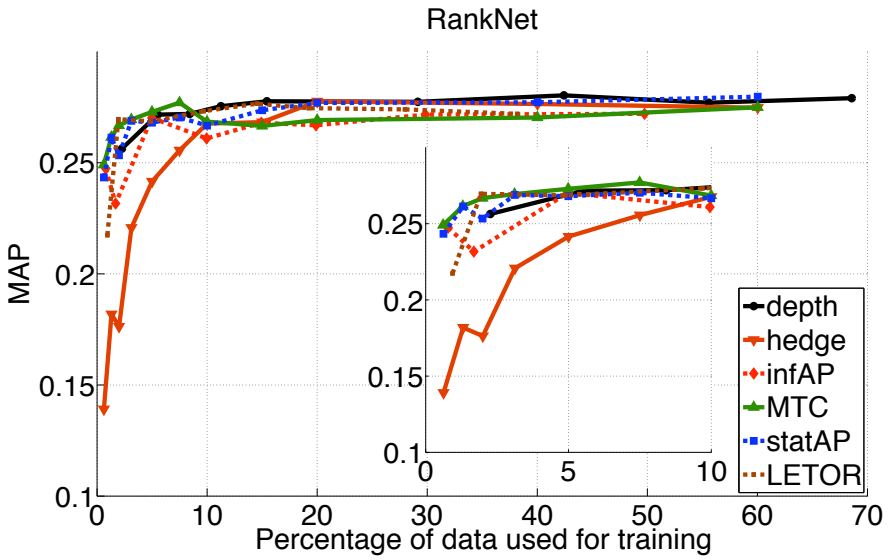
- Some Learning-to-Rank algorithms are robust to document selection methodologies
  - LambdaRank vs. RankBoost





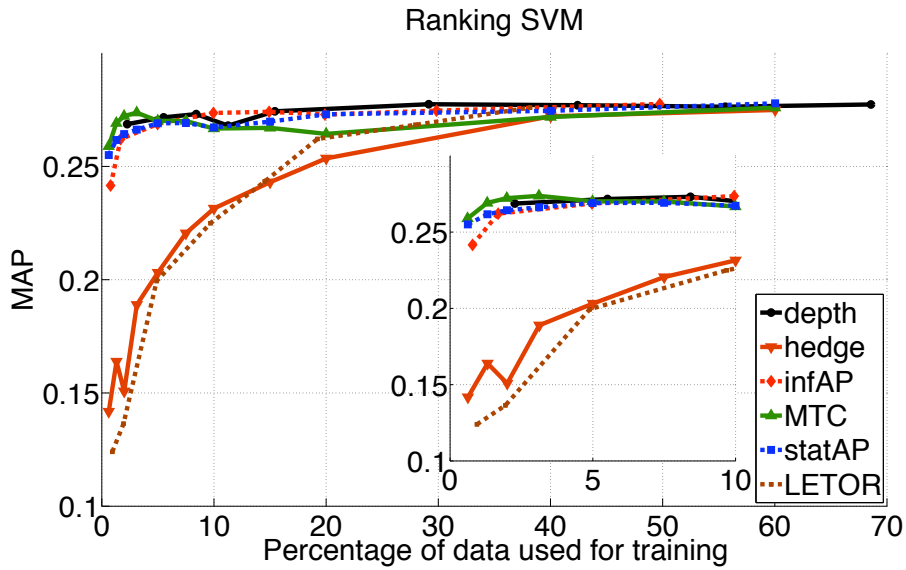
Observations (2)

- Near-optimal performance with 1%-2% of complete collection (depth-100 pool)
  - No significant differences at greater % (t-test)
  - Number of features matter [Taylor et.al '06]



Observations (3)

- Selection methodology matters
  - Hedge (worst performance)
  - Depth-k pooling and statAP (best performance)
  - LETOR-like (neither most efficient nor most effective)



Relative Importance on Effectiveness

- Learning-to-Rank algorithm vs. document selection methodology
  - 2-way ANOVA model
- Variance decomposition over all data sets
  - 26% due to document selection
  - 31% due to LTR algorithm
- Variance decomposition (small data sets, <10%)
  - 44% due to document selection
  - 31% due to LTR algorithm

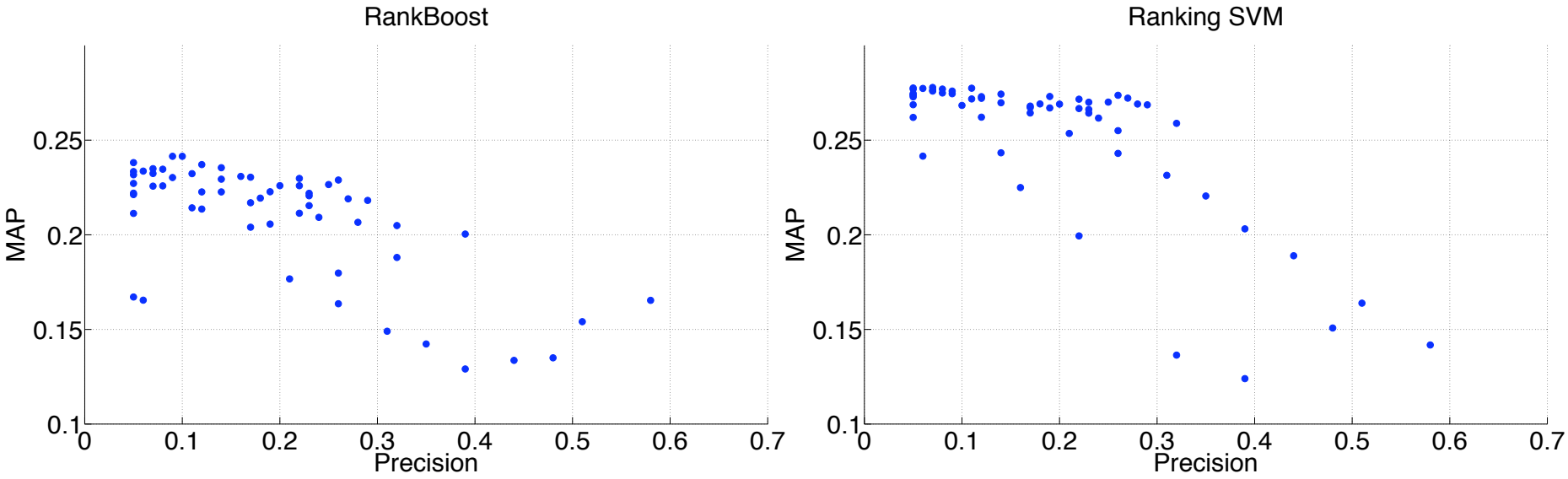
What makes one training set better than another?

- Different methods have different properties
  - Precision
  - Recall
  - Similarities between relevant documents
  - Similarities between relevant and non-relevant documents
  - …
- Model selection

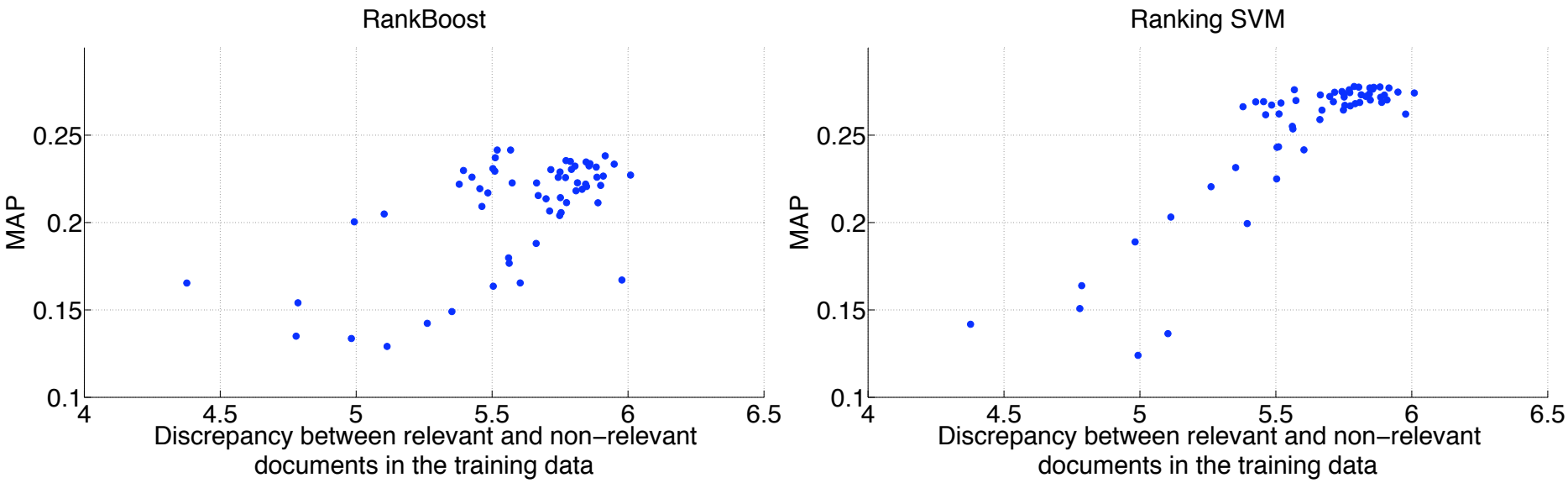
What makes one training set better than another?

- Different methods have different properties
  - Precision
  - Recall
  - Similarities between relevant documents
  - Similarities between relevant and non-relevant documents
  - …
- Model selection
  - Linear model (adjusted $R^2 = 0.99$)

What makes one training set better than another?



What makes one training set better than another?



Conclusions

- Some LTR algorithms are robust to document selection methodologies
- For those not, selection methodology matters
  - Depth-k pooling, stratified sampling
- Harmful to select too many relevant docs
- Harmful to select relevant and non-relevant docs that are too similar