Team OpenWebSearch at QuantumCLEF: Feature Selection with Bootstrapping

CLEF 2024, 9–12 September 2024, Grenoble, France






Maik Fröbe[1] Daria Alexander[2] Gijs Hendriksen[2] Ferdinand Schlatt[1] Martin Potthast[3] Matthias Hagen[1]

[1] FRIEDRICH-SCHILLER-UNIVERSITÄT JENA

[2] Radboud Universiteit

[3] UNIVERSITÄT LEIPZIG

webis.de

OWS at QuantumCLEF: Feature Selection with Bootstrapping

Motivation: Effective Learning-to-Rank Approaches are not Deterministic

Compare k-nearest neighbors with LambdaMART for shuffled training data

OWS at QuantumCLEF: Feature Selection with Bootstrapping

Motivation: Effective Learning-to-Rank Approaches are not Deterministic

Compare k-nearest neighbors with LambdaMART for shuffled training data

K-nearest neighbors:

- Con: Most likely not effective
- Pro: Two models trained on shuffled data are identical

OWS at QuantumCLEF: Feature Selection with Bootstrapping

Motivation: Effective Learning-to-Rank Approaches are not Deterministic

Compare k-nearest neighbors with LambdaMART for shuffled training data

K-nearest neighbors:

- Con: Most likely not effective
- Pro: Two models trained on shuffled data are identical

LambdaMART:

- Pro: Most likely effective
- Con: Two models trained on shuffled data are different

OWS at QuantumCLEF: Feature Selection with Bootstrapping

Motivation: Effective Learning-to-Rank Approaches are not Deterministic

Compare k-nearest neighbors with LambdaMART for shuffled training data

K-nearest neighbors:

- Con: Most likely not effective
- Pro: Two models trained on shuffled data are identical

LambdaMART:

- Pro: Most likely effective
- Con: Two models trained on shuffled data are different

Different LambdaMART models could contradict themself

- Potentially different predictions
- Potentially different feature importance

Goal: Select features that are robust accross diverse training procedures

OWS at QuantumCLEF: Feature Selection with Bootstrapping

Bootstrapping

[Efron'94]

Bootstrapping is a statistical technique in which repeated samples are drawn from data to obtain a distribution for subsequent statistical analyses

OWS at QuantumCLEF: Feature Selection with Bootstrapping

Bootstrapping

[Efron'94]

Bootstrapping is a statistical technique in which repeated samples are drawn from data to obtain a distribution for subsequent statistical analyses

We use bootstrapping to sample LambdaMART models

- Feature importance of the models contradicts each other
- Use annealing to find a subset of features important accross many samples

OWS at QuantumCLEF: Feature Selection with Bootstrapping

Bootstrapping

[Efron'94]

Bootstrapping is a statistical technique in which repeated samples are drawn from data to obtain a distribution for subsequent statistical analyses

We use bootstrapping to sample LambdaMART models

- Feature importance of the models contradicts each other
- Use annealing to find a subset of features important accross many samples

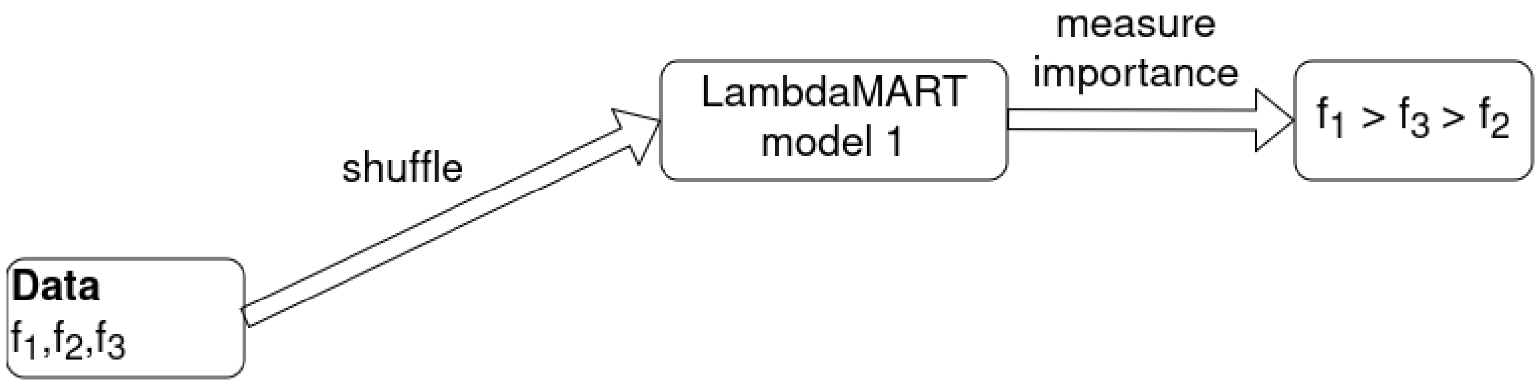
Related Work: Bootstrapping used in IR to

- analyse evaluation measures
  [Sakai'06, Sakai'07, Zobel'20]
- conduct significance tests
  [Savoy'97, Smucker'07, Ferro'22]

  estimate if system rankings transfer to different corpora
  [Cormack'06]
- handle unjudged documents
  [Froebe'23]

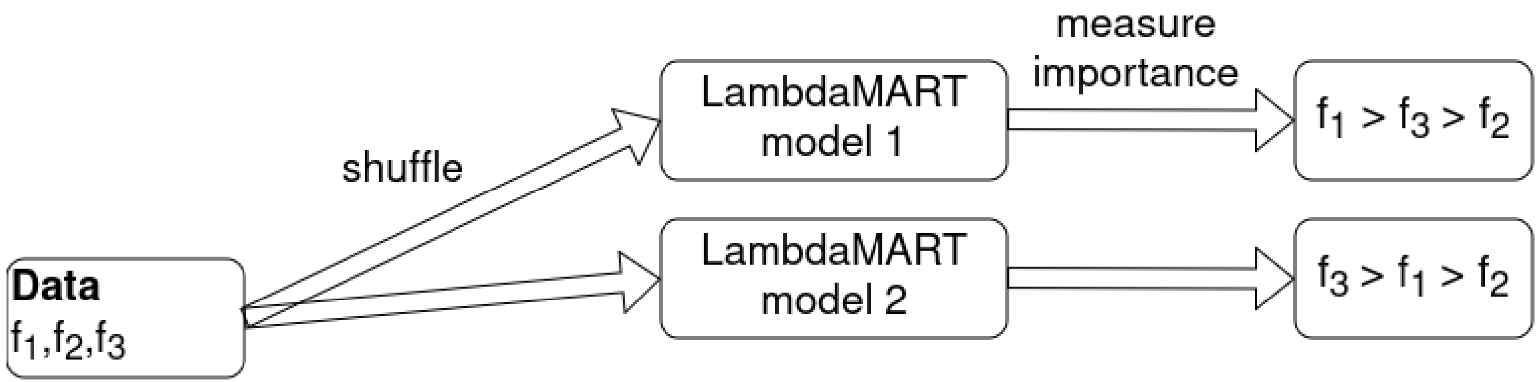
OWS at QuantumCLEF: Feature Selection with Bootstrapping

Bootstrapping LambdaMART Models: Example



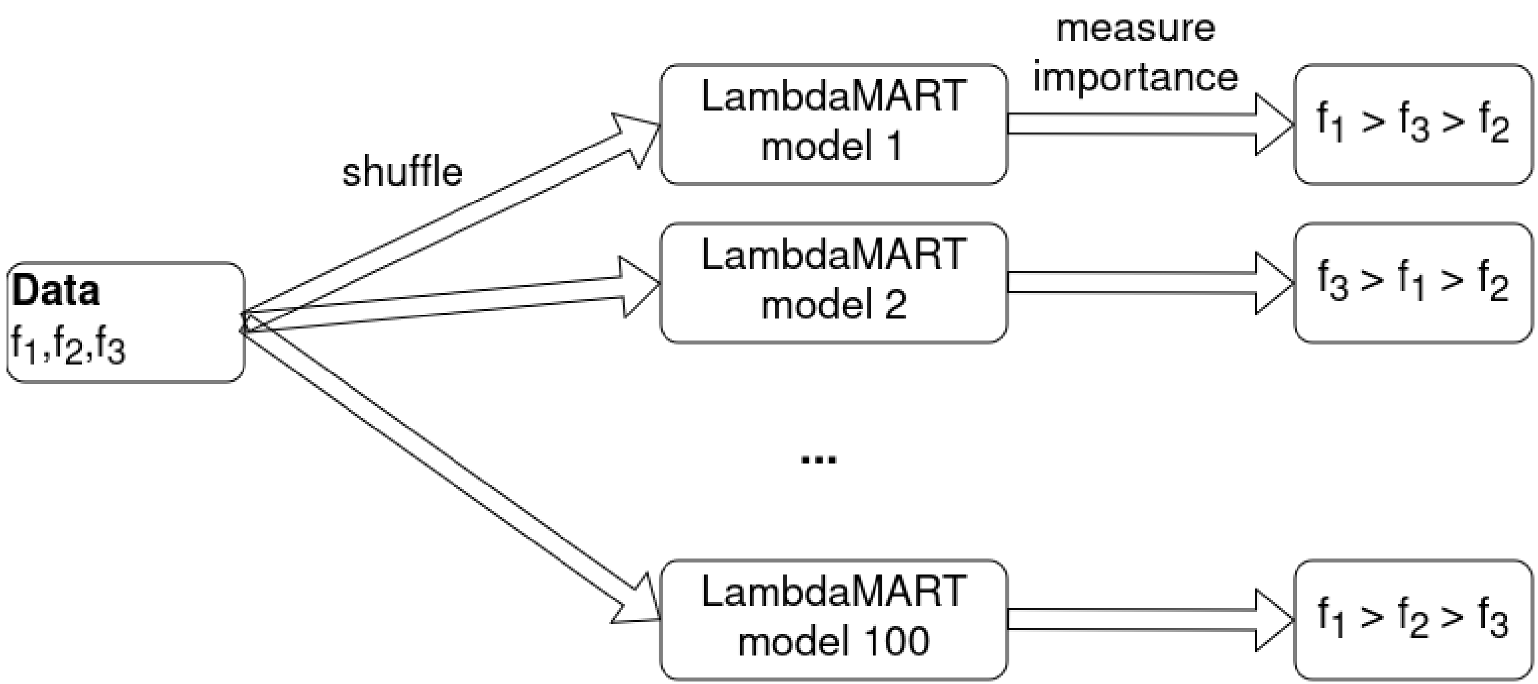
OWS at QuantumCLEF: Feature Selection with Bootstrapping

Bootstrapping LambdaMART Models: Example



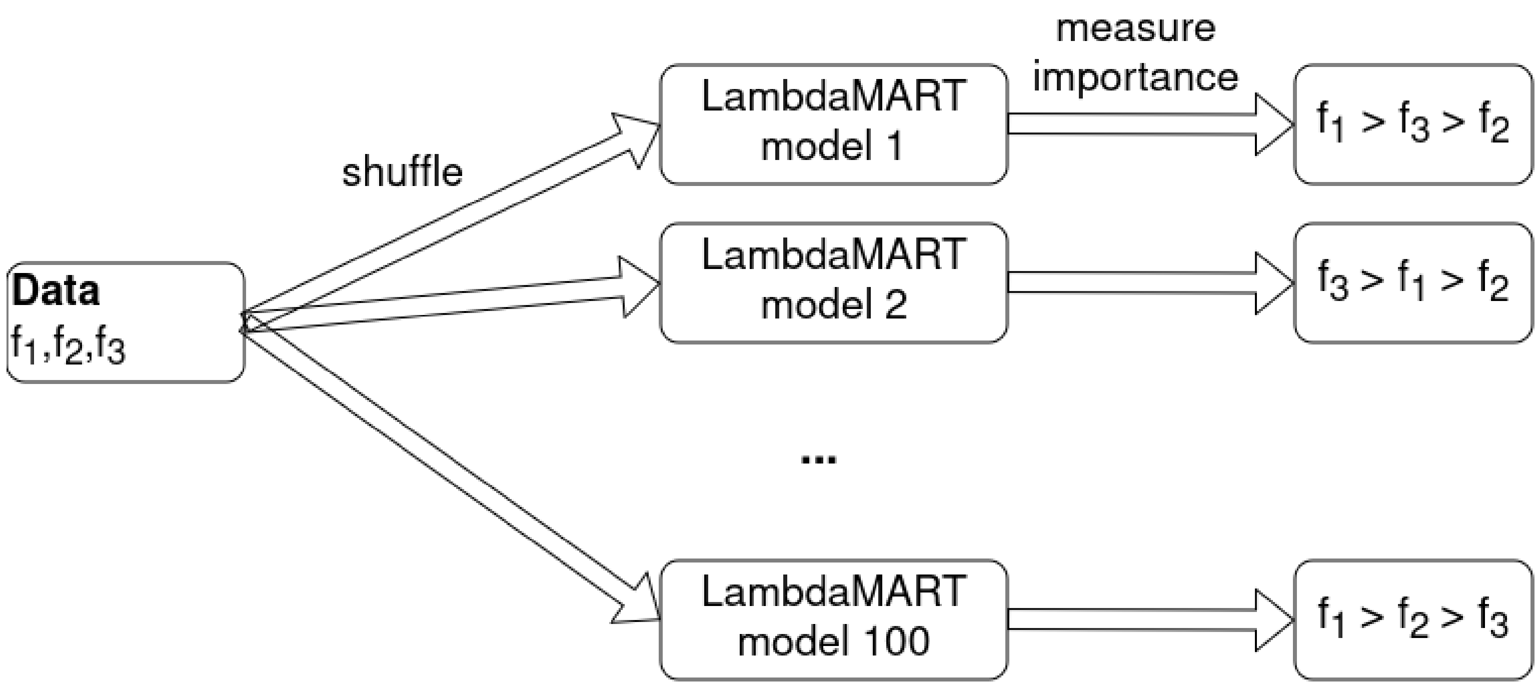
OWS at QuantumCLEF: Feature Selection with Bootstrapping

Bootstrapping LambdaMART Models: Example



OWS at QuantumCLEF: Feature Selection with Bootstrapping

Bootstrapping LambdaMART Models: Example



Result:

- We get 100 LambdaMart models
- For each model, we calculate the feature importance
- Integration into QUBO via normalized importance of a feature respectively feature pair accross all bootstrapps

OWS at QuantumCLEF: Feature Selection with Bootstrapping

QUBO Formulation: The Linear Part

Normalized importance of a feature accross all bootstrapps

OWS at QuantumCLEF: Feature Selection with Bootstrapping

QUBO Formulation: The Linear Part

Normalized importance of a feature accross all bootstrapps

Formally:

$$\sum_{k=1}^{b} \frac{X_i^k}{|X|}$$

- b: number of bootstrapps
- X_i^k: importance of feature i in the k-th bootstrapped model
- $|X|$: Overall importance

OWS at QuantumCLEF: Feature Selection with Bootstrapping

QUBO Formulation: The Linear Part

Normalized importance of a feature accross all bootstrapps

Formally:

$$\sum_{k=1}^{b} \frac{X_i^k}{|X|}$$

- b: number of bootstrapps
- X_i^k: importance of feature i in the k-th bootstrapped model
- $|X|$: Overall importance

Results (nDCG@10 on MQ2007):

- Linear part from bootstrapping
- Quadratic part from official baseline (i.e., conditional mutual information)
- We select 25 features (not optimized)

	Linear
Simulated Annealing	0.451
Quantum Annealing	0.448

OWS at QuantumCLEF: Feature Selection with Bootstrapping

QUBO Formulation: The Quadratic Part

Normalized importance of feature pairs accross all bootstrapps

OWS at QuantumCLEF: Feature Selection with Bootstrapping

QUBO Formulation: The Quadratic Part

Normalized importance of feature pairs accross all bootstrapps

Formally:

$$\sum_{k=1}^{b} \frac{X_i^k + X_j^k}{|X|}$$

- b: number of bootstrapps
- X_i^k, X_j^k: importance of feature i respectively j in the k-th bootstrapped model
- $|X|$: Overall importance

OWS at QuantumCLEF: Feature Selection with Bootstrapping

QUBO Formulation: The Quadratic Part

Normalized importance of feature pairs accross all bootstrapps

Formally:

$$\sum_{k=1}^{b} \frac{X_i^k + X_j^k}{|X|}$$

- b: number of bootstrapps
- X_i^k, X_j^k: importance of feature i respectively j in the k-th bootstrapped model
- $|X|$: Overall importance

Results (nDCG@10 on MQ2007):

- Linear part from bootstrapping
- Quadratic part from bootstrapping
- We select 25 features (not optimized)

	Linear	Quadratic
Simulated Annealing	0.451	0.452
Quantum Annealing	0.448	0.451

Conclusion

We implemented feature selection via bootstrapping

- Repeatedly train LambdaMART models on randomized feature sets
- Measure importance of features in trained model

Implementation available online: bitbucket.org/eval-labs/qc24-ows

Conclusion

We implemented feature selection via bootstrapping

- Repeatedly train LambdaMART models on randomized feature sets
- Measure importance of features in trained model

Implementation available online: bitbucket.org/eval-labs/qc24-ows

Future work

- Determine the number of to-be-selected features
- Vary hyperparameters during bootstrapps
- Why was quantum annealing less effective than simulated annealing for us?

Conclusion

We implemented feature selection via bootstrapping

- Repeatedly train LambdaMART models on randomized feature sets
- Measure importance of features in trained model

Implementation available online: bitbucket.org/eval-labs/qc24-ows

Future work

- Determine the number of to-be-selected features
- Vary hyperparameters during bootstrapps
- Why was quantum annealing less effective than simulated annealing for us?

Thank You!