Online Ranking Combination

Erzsébet Frigó

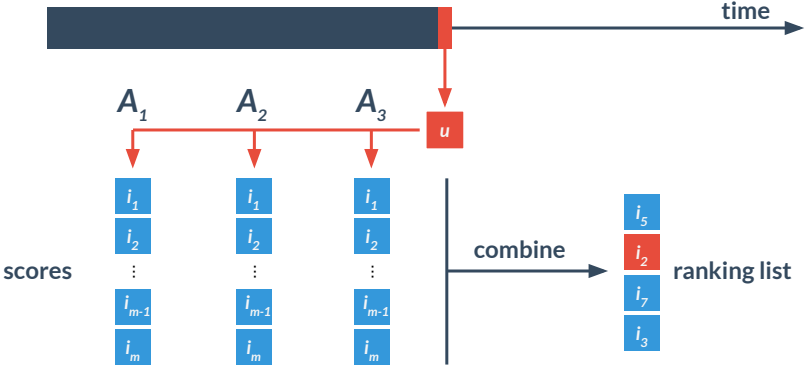
Institute for Computer Science and Control (MTA SZTAKI)

Joint work with Levente Kocsis

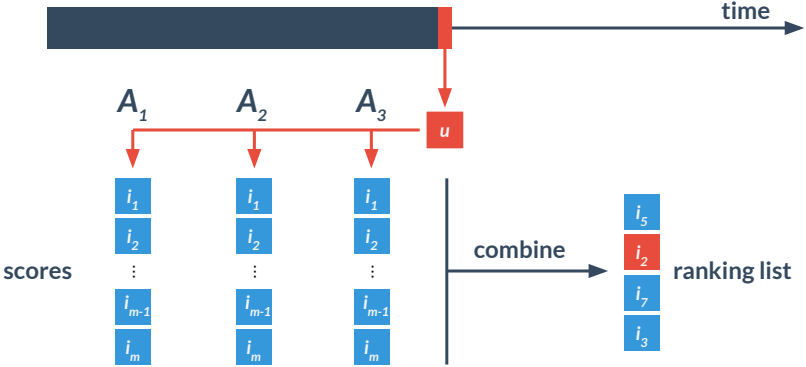
Overview

- Framework: prequential ranking evaluation
- Goal: optimize convex combination of ranking models
- Our proposal: direct optimization of the ranking function

Model combination in prequential framework with ranking evaluation



Model combination in prequential framework with ranking evaluation



Objective: choosing combination weights.

New idea: optimize ranking function directly

- Standard method: take a surrogate function and use its gradient
  - E.g. MSE
- Drawback: optimum of the surrogate $\neq$ optimum of ranking function
- Proposed solution: optimize the ranking function directly
- Two approaches:
  - Global search in the weight space
  - Gradient approximation (finite differences)

ExpW

- Choose a subset Q of the weight space Θ
  - e.g., lay a grid to the parameter space
- Apply exponentially weighted forecaster on Q

$$P(\text{select } \mathbf{q} \in Q \text{ in round } t) = \frac{e^{-\eta_t \sum_{\tau=1}^{t-1}(1-r_\tau(\mathbf{q}))}}{\sum_{\mathbf{s}\in Q} e^{-\eta_t \sum_{\tau=1}^{t-1}(1-r_\tau(\mathbf{s}))}}$$

- Theoretical guarantee:

$$E[R_T(\text{best static combination in } \Theta) - R_T(ExpW)] < O(\sqrt{T})$$

  - if the cumulative reward function (R_T) is sufficiently smooth
  - and Q is sufficiently large
- Difficulty: size of Q is exponential in number of base rankers, can't scale

Simultaneous Perturbation Stochastic Approximation (SPSA)

- Approximated gradient (for the weight of base ranker i in round t):

$$g_{ti} = \frac{r_t(\boldsymbol{\theta}_t + c_t \boldsymbol{\Delta}_t) - r_t(\boldsymbol{\theta}_t - c_t \boldsymbol{\Delta}_t)}{c_t \Delta_{ti}}$$

  - $\boldsymbol{\theta}_t$ is the current combination weight vector
  - $\boldsymbol{\Delta}_t = (\Delta_{t1}, \ldots)$ is a random vector of +/-1
  - c_t is perturbation step size
- Online update step: one gradient step using the approximated gradient

RSPSA

- RSPSA = SPSA + Resilient Backpropagation (RProp)

RSPSA

- RSPSA = SPSA + Resilient Backpropagation (RProp)
- RProp defines gradient step sizes for each weight
- Perturbation step size is tied to gradient step size
- Update step sizes using RProp

Resilient Backpropagation (RProp)

- Gradient update rule
- Predefined step size for each coordinate
  - ignores the length of the gradient vector
- Step size is updated based on the sign of the gradient
  - decrease step if gradient changed direction
  - increase otherwise

$$g_{ti} = \frac{r_t(\theta_t + c_t \Delta_t) - r_t(\theta_t - c_t \Delta_t)}{c_t \Delta_{ti}}$$

RFDSA+

- Switch to finite differences (FD)
  - allows to detect 0 gradient w.r.t. one coordinate
- If the gradient is 0 w.r.t. a coordinate, then
  - increase perturbation size (+) for that coordinate
  - escape flat section in the right direction
- RFDSA+ = RSPSA - simultaneous perturbation + finite differences + zero gradient detection
- The modifications might seem to be minor, but are essential to make the algorithm work

Experiments - Datasets, base rankers

- 5 datasets
  - Amazon
    - CDs and Vinyl
    - Movies and TV
    - Electronics
  - MovieLens 10M
  - Twitter
    - hashtag prediction
- Size
  - # of events: 2M-10M
  - # of users: 70k-4M
  - # of items: 10k-100k

- Base rankers:
  - Models updated incrementally
    - SGD Matrix Factorization
    - Asymmetric Matrix Factorization
    - Item-to-item similarity
    - Most popular
  - Traditional models updated periodically
    - SGD Matrix Factorization
    - Implicit Alternating Least Squares MF

Combination algorithms in the experiments

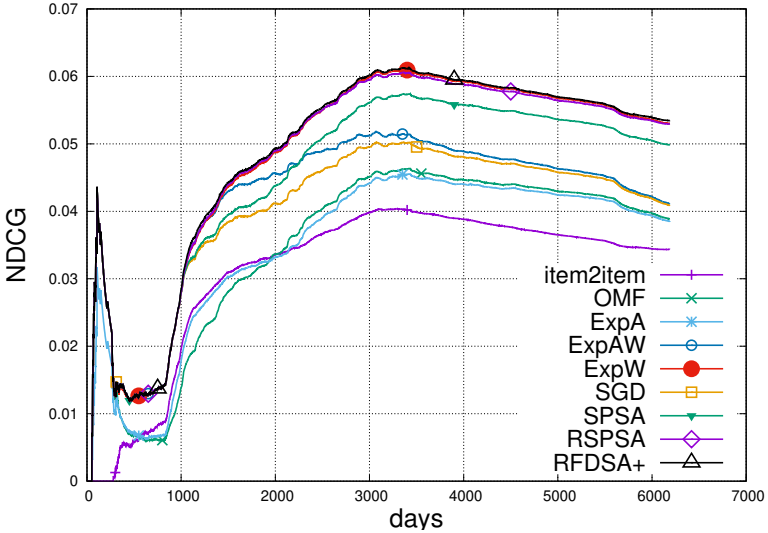
Direct optimization:

- **ExpW**
  - exponentially weighted forecaster on a grid
  - global optimization
- **SPSA**
  - gradient method with simultaneous perturbation
- **RSPSA**
  - SPSA with RProp
- **RFDSA+**
  - our new algorithm
  - finite differences, flat section detection

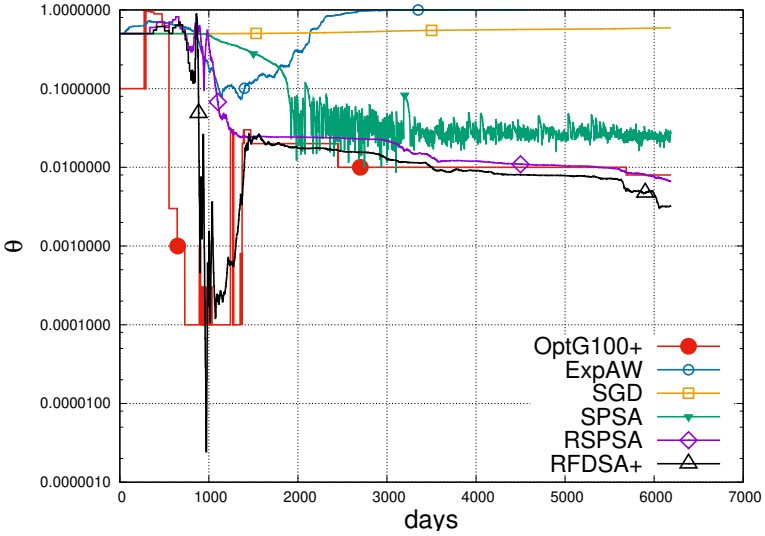
Baselines:

- **ExpA**
  - exponentially weighted forecaster on the base rankers
- **ExpAW**
  - use probabilities of ExpA as weights
- **SGD**
  - use MSE as a surrogate
  - target=1 for positive sample
  - target=0 for generated negative samples

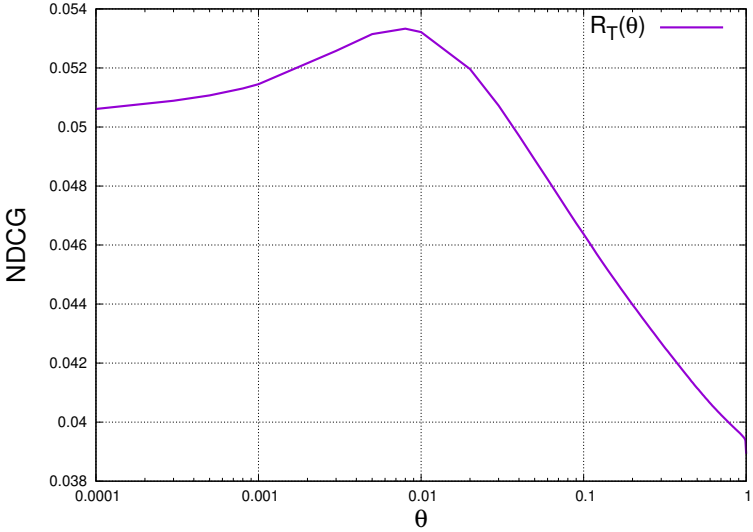
Results - 2 base rankers (i2i, OMF) - nDCG



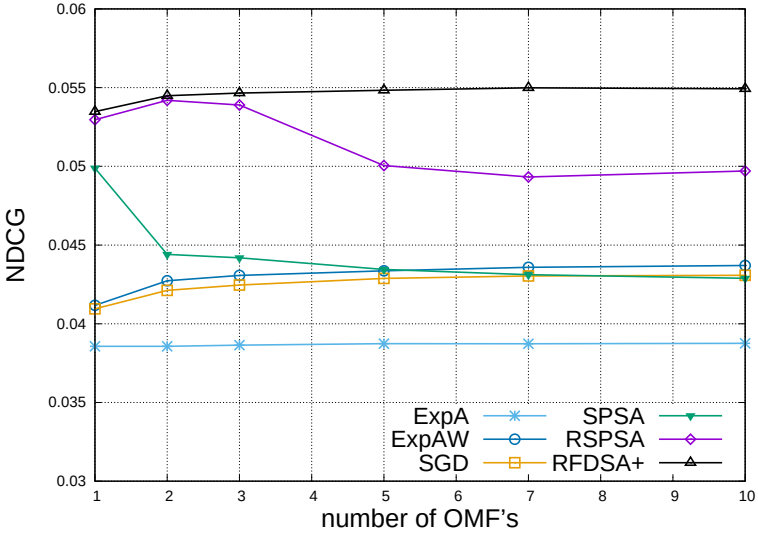
Results - 2 base rankers - Combination weights



Cumulative reward as function of combination weight



Results - Scalability



Results - 6 base rankers - DCG

		item2item	OMF	Pop	OAMF	MF	iALS	ExpA	ExpAW	SGD	SPSA	RSPSA	RFDSA+
Amazon	CD	0.03	0.04	0.06	0.03	0.01	0.00	0.06	0.06	0.06	0.07	0.06	0.09
	Movies	0.04	0.04	0.07	0.03	0.01	0.01	0.07	0.07	0.07	0.07	0.07	0.09
	Electro	0.02	0.02	0.03	0.02	0.01	0.01	0.03	0.03	0.04	0.03	0.04	0.05
MovieLens		0.14	0.14	0.09	0.17	0.01	0.01	0.17	0.17	0.16	0.17	0.14	0.19
Twitter		0.02	0.35	0.35	0.31	0.01	0.01	0.35	0.35	0.36	0.37	0.45	0.46

Conclusions

- Problem: combine ranking algorithms
- Our proposal: optimize the ranking measure directly
- Global optimization (ExpW) works well in case of two base algo
- Our new algo: RFDSA+
  - solves problems (scaling, constant sections w.r.t one coordinate)
  - strong combination

The End

Online Ranking Combination

Erzsébet Frigó

Institute for Computer Science and Control (MTA SZTAKI)

Joint work with Levente Kocsis