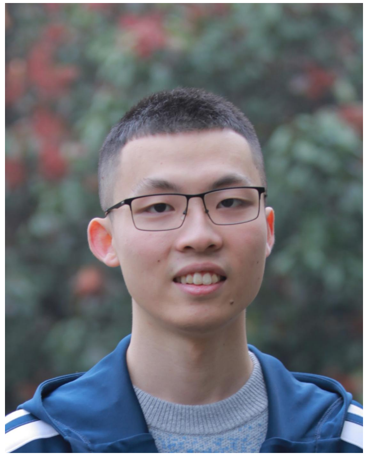
Bidder Selection Problem in Position Auctions:

A Fast and Simple Algorithm via Poisson Approximation



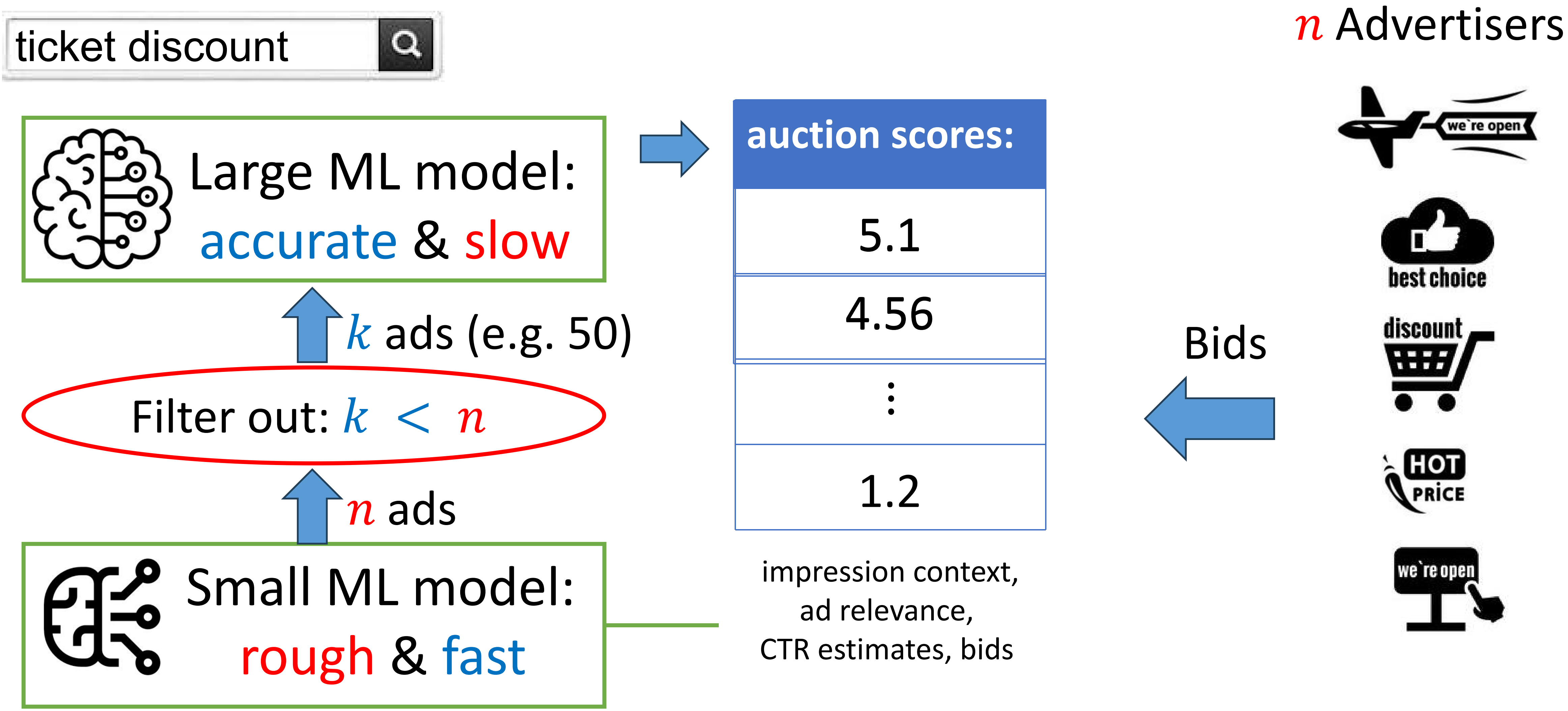


Nick Gravin
ITCS, Shanghai University of Finance and Economics

Yixuan Even Xu
IIIS, Tsinghua University

Renfei Zhou
IIIS, Tsinghua University

Bidder Selection Problem



Theoretical Model

k-max problem (<u>single-item auction</u>):

- INPUT: n independent random variables $X_1, \ldots, X_n$ and $k < n$
- OUTPUT: find k r.v. $S \subset [n]$ to $\max_{|S|=k} \mathbf{E}\left[\max_{i \in S} X_i\right]$

k-max for <u>position auctions</u>:

- generalizes single-item auction
- INPUT: r.v. $X_1, \ldots, X_n$ and $k < n$
- OUTPUT: k r.v. $\max_{|S|=k} \mathbf{E}\left[Welfare(S)\right]$

Our Results: Theory

New relaxation of k-max: $\max_{\|\boldsymbol{x}\|_1 \le k} f(\boldsymbol{x})$

1. $f(\boldsymbol{x})$ – nicely structured concave function $\Rightarrow$

   Easy to find $\boldsymbol{x}^* = \operatorname{argmax}_{\|\boldsymbol{x}\|_1 \le k} f(\boldsymbol{x})$

2. $f(\boldsymbol{x}^*) \to \max_{|S|=k} \mathbf{E}\left[\max_{i \in S} X_i\right]$ as k grows

3. Works for general position auctions.

$\Rightarrow$ Position auctions: Implementable PTAS

Our Results: Practice

- Homebrew implementation (python + standard convex libs):
  - Benchmarks: Greedy (Submodular Opt.) + Local Search
  - Instances as large as n = 1000, k = 200.
  - Good approximation on all instances.

Runtime: 1 day

Runtime: 45 sec.

Approx. to benchmarks: >99%

- Previous 2 EPTAS algorithms on k-max:
  - Unimplementable

Runtime: years $\left(2^{O(1/\varepsilon)^{O(1/\varepsilon)}}\right)$

n=3, k=2, ε=0.2