Surrogate Functions for Maximizing Precision at the Top

Purushottam Kar[#], Harikrishna Narasimhan[*], and Prateek Jain[#]

[#]Microsoft Research, Bengaluru, INDIA
[*]Indian Institute of Science, Bengaluru, INDIA

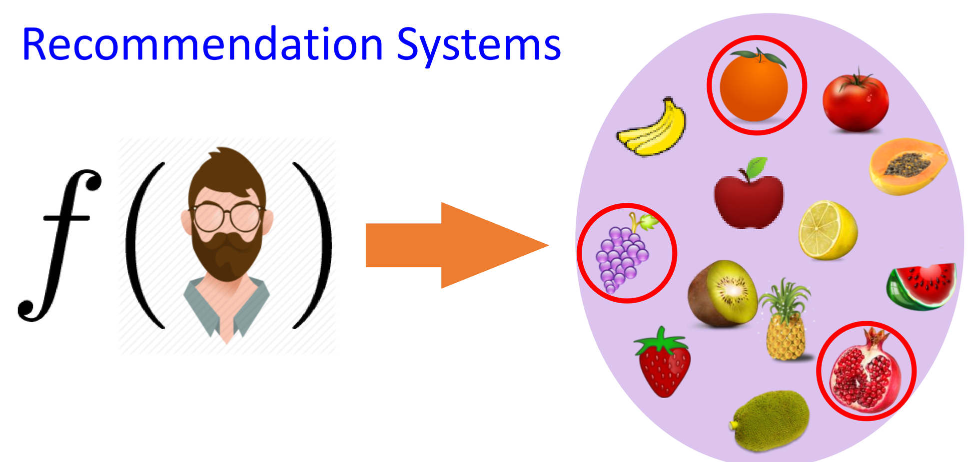
The Goal

Scalable routines for provable maximization of precision at the top of ranked lists

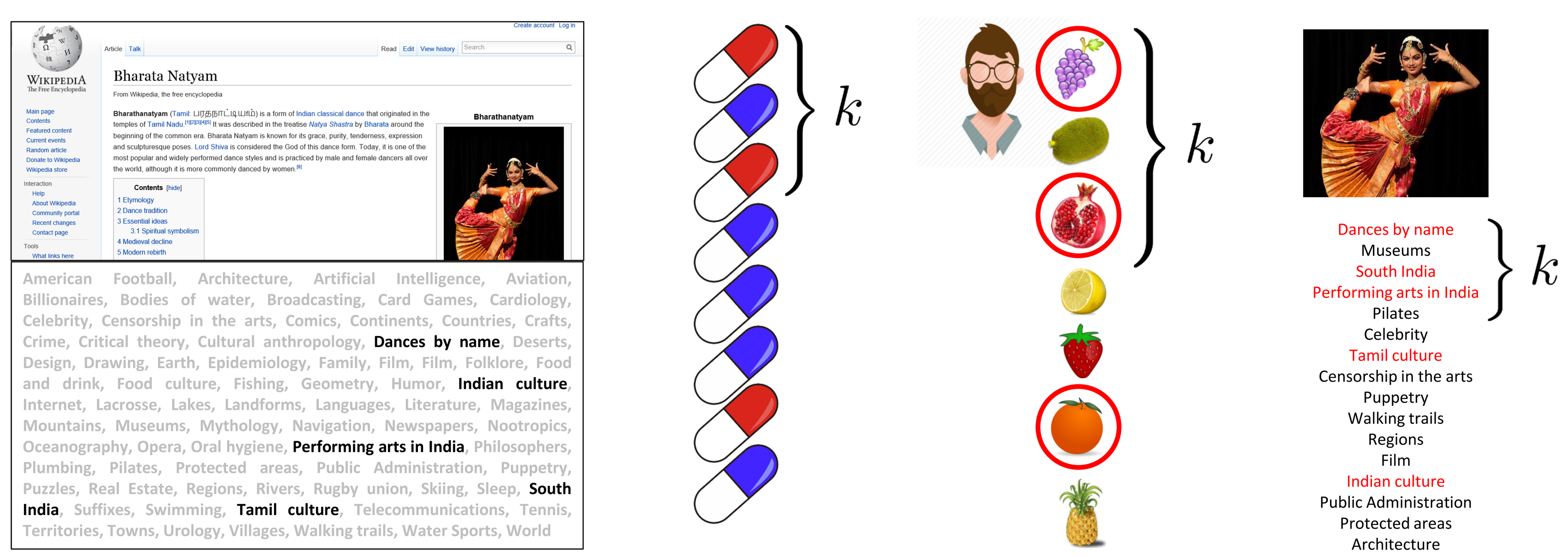
Applications

Drug Discovery

Recommendation Systems



Document Tagging



Rank objects in order of (perceived) relevance scores and retrieve top ranked objects

prec@$k(\cdot)$ Error Function

Prec@$k(s)$: # irrelevant objects in top-k positions if sorted by scores s

- Non-convex, non-smooth performance measure
- Non-additive: direct application of SGD/Perceptron methods not possible
- Existing Work [Joachims 05]
  Struct-SVM surrogates: not satisfactory
  Cutting-plane solvers: unsuitable for large, streaming data

Methodology

- Given n objects $(\mathbf{x}_i, y_i)$, $y_i \in \{0,1\}$
- Assign scores $s = (s_1, s_2, \ldots, s_n)$
- Predict top-k scoring objects as *relevant*
- Learn models that predict *good* score vectors
- Learning on streaming data?

$$s(\;) = \langle \mathbf{w}, \phi(\;) \rangle$$

Question I

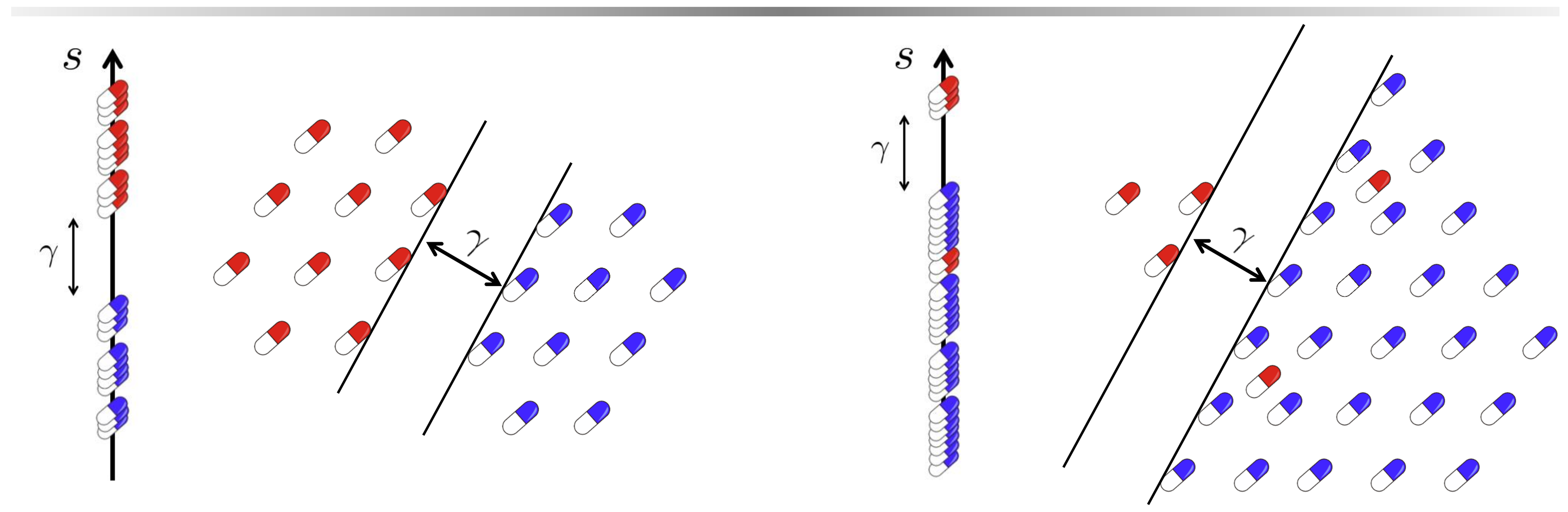
Convex, Upper-bounding and Conditionally Consistent surrogates for prec@k

What is a *good* Surrogate for prec@k?

- Convexity (CV)
  $\ell_{\text{prec}@k}(s)$ is convex over $\mathbb{R}^n$
- Upper-bounding Property (UB)
  $\ell_{\text{prec}@k}(s) \geq \text{prec}@k(s) \quad \forall s$
- Tight under a Margin (TuM)
  For classes of score vectors $\mathcal{S}$ satisfying an appropriate margin condition
  $\min_{s \in \mathcal{S}} \ \ell_{\text{prec}@k}(s) = \min_{s \in \mathcal{S}} \ \text{prec}@k(s)$

	Struct-SVM	Ramp (Our)	Avg (Our)
CV	✓	✗	✓
UB	✗	✓	✓
TuM	✗	✓	✓

A Notion of *Margin* for prec@k



Classification Margin

(Weak) prec@k Margin

Surrogates for prec@k

Ramp Surrogate $\ell^{\text{ramp}}_{\text{prec}@k}(s)$

$$\max_{\|\hat{\mathbf{y}}\|_0 = k} \left\{ \text{prec}@k(\hat{\mathbf{y}}) + \sum \hat{\mathbf{y}}_i s_i \right\} - \max_{\substack{\|\tilde{\mathbf{y}}\|_0 = k \\ \text{prec}@k(\tilde{\mathbf{y}})=0}} \sum \tilde{\mathbf{y}}_i s_i$$

Penalize score vectors that don't give k relevant objects the highest scores

Avg Surrogate $\ell^{\text{avg}}_{\text{prec}@k}(s)$

$$C(\hat{\mathbf{y}}) = \frac{n_+ - k + \text{prec}@k(\hat{\mathbf{y}})}{n_+ - k}$$

$$\max_{\|\hat{\mathbf{y}}\|_0 = k} \left\{ \text{prec}@k(\hat{\mathbf{y}}) + \sum (\hat{\mathbf{y}}_i - \mathbf{y}_i) s_i + \frac{1}{C(\hat{\mathbf{y}})} \sum (1 - \hat{\mathbf{y}}_i) \mathbf{y}_i s_i \right\}$$

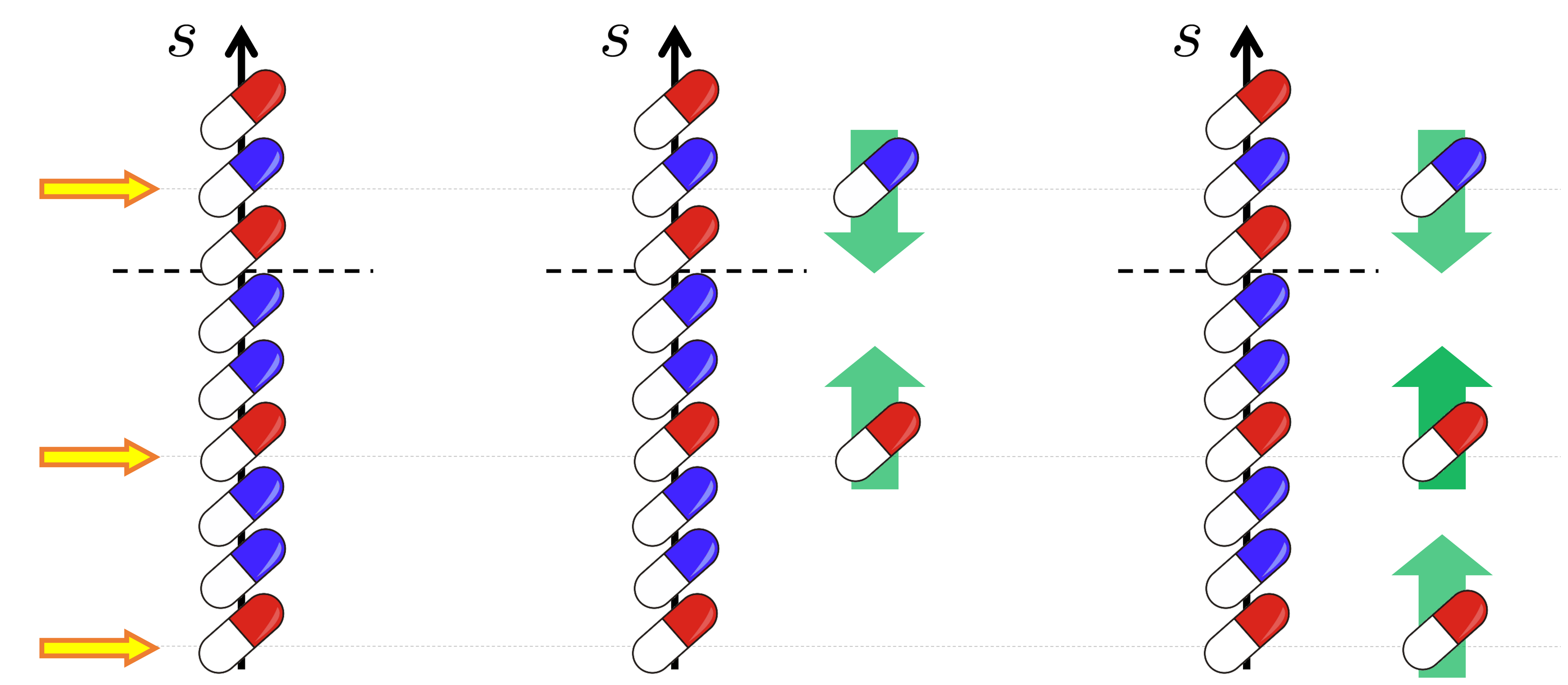
Relax the Ramp surrogate or else add corrections to the struct-SVM surrogate

Lemma: The avg (ramp) surrogate is tight for any class of score vectors $\mathcal{S}$ that contains a score vector realizing a unit (weak) prec@k margin.

Question II

Scalable optimization of prec@k in large-scale and streaming data settings

PERCEPTRON@k Algorithm for Optimizing prec@k



PERCEPTRON@k-MAX $\quad \mathbf{w}_{\text{new}} = \mathbf{w}_{\text{old}} + \phi(\;) - \phi(\;)$

PERCEPTRON@k-AVG $\quad \mathbf{w}_{\text{new}} = \mathbf{w}_{\text{old}} + \frac{\phi(\;) + \phi(\;)}{2} - \phi(\;)$

Lemma: If there exists a linear scoring function that realizes a prec@k margin of γ, then PERCEPTRON@k-AVG terminates in $4k/\gamma^2$ steps

Lemma: If PERCEPTRON@k-AVG is executed for T steps and $\Delta_t = \text{prec}@k(\mathbf{w}_t)$,

$$\sum_{t=1}^{T} \Delta_t \leq \min_{\mathbf{w} \in \mathcal{W}} \left(\|\mathbf{w}\|_2 \sqrt{4k} + \sqrt{T \cdot \ell^{\text{avg}}_{\text{prec}@k}(\mathbf{w})} \right)^2$$

Other Results: OTB and UC bounds for preck@k and its surrogates

Experiments

- Gradient descent-based approach GD@k based on surrogates
- Mini-batch versions of PERCEPTRON@k and GD@k
- Mistake/generalization bounds via OTB/UC