Collaborative Ranking for Local Preferences

Berk Kapicioglu — *YP Mobile Labs*

David S. Rosenberg — *YP Mobile Labs*

Robert E. Schapire — *Princeton University*

Tony Jebara — *Columbia University*

General Problem

- Goal: Design a ranking algorithm for recommender systems with local feedback.
- A recommender system with local feedback works as follows:
  1. User queries a database.
  2. Database returns a subset of items.
  3. An algorithm ranks these items for the user.
  4. User chooses an item from the ranked list.
  5. The ranking algorithm is retrained based on the user's local feedback.

Movie Recommendation

User queries recent action movies.

Algorithm returns a user-specific ranking.

User chooses a movie to watch and the local feedback is used to retrain the learning algorithm.

Previous Work

- Collaborative Ranking (CofiRank, Weimer et al., NIPS, 2007).
- Collaborative Ranking with Implicit Feedback
- (BPR, Rendle et al., UAI, 2009).

items

	...	...	...	...	...
	...	4	x	4	...
	...	3	x	5	...
users	...	4	3	x	...
	...	5	2	x	...
	...	...	...	...	...

Current Work

- Collaborative Local Ranking (CLR).
- Local and relative feedback.

items

	...	...	...	...	...
	...	x	x	x	...
	...	x	x	x	...
users	...	-1	+1	-1	...
	...	x	x	x	...
	...	...	...	...	...

Formal Setup

- Local ranking loss:

$$L_M(u,C,i) = \underset{c \overset{U}{\sim} C^i}{P}\left[M_{u,i} - M_{u,c} \le 0\right]$$

- Example:
  - Correct movie is ranked highest. Loss is 0.
  - Correct movie is ranked lowest. Loss is 1.

Generalization Bound

- Theorem: With probability at least $1-\delta$, for any matrix $M \in \mathbb{R}^{m \times n}$ with rank at most r,

$$\underset{(u,C,i)\sim\mathcal{D}}{\mathbb{E}} L_M(u,C,i) \le \underset{(u,C,i)\overset{U}{\sim}S}{\mathbb{E}} L_M(u,C,i) + 2\sqrt{\frac{2r(m+n)\ln\left(\frac{16emn}{r}\right)}{d}} + \sqrt{\frac{2\ln\left(\frac{2}{\delta}\right)}{d}}.$$

- Why is this bound significant?
  - Only bound for collaborative ranking that we know of.
  - Just slightly looser than the bound for collaborative binary classification (Srebro and Jaakkola, NIPS, 2005).

Objective Function

- Goal: Minimize empirical local ranking loss.
- First, make it tractable.

$$\underset{(u,C,i)\overset{U}{\sim}S}{\mathbb{E}} L_M(u,C,i) \le \frac{1}{|S|}\sum_{(u,C,i)\in S}\frac{1}{|C^i|}\sum_{c\in C^i} h\left(\left(UV^T\right)_{u,i} - \left(UV^T\right)_{u,c}\right)$$

- Second, bound the trace norm.

$$f^{\mathrm{CLR}}(S;U,V) = \frac{\lambda}{2}\left(\|U\|_F^2 + \|V\|_F^2\right) + \frac{1}{|S|}\sum_{(u,C,i)\in S}\frac{1}{|C^i|}\sum_{c\in C^i} h\left(\left(UV^T\right)_{u,i} - \left(UV^T\right)_{u,c}\right)$$

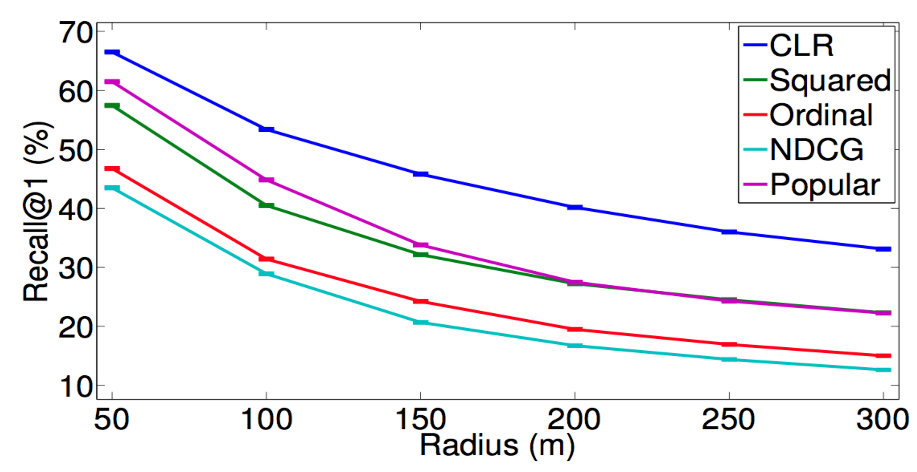
Algorithm

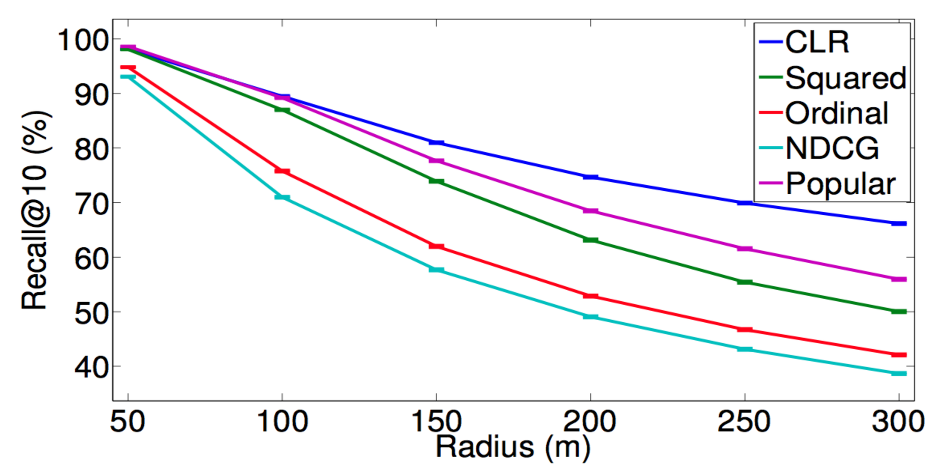
- Objective is bi-convex (alternating minimization).
- Minimizing U
  - Equivalent to training m ranking SVMs (Joachims, KDD, 2003).
- Minimizing V
  - Non-standard objective function.
  - SVM-like.
- $\sum_{u=1}^{m} f^{\mathrm{rSVM}}(S_u; U_u, V)$
- Projected stochastic subgradient descent.
  - Shalev-Shwartz et al., ICML, 2007.
  - Supervised learning vs. Collaborative filtering.
  - $\tilde{O}\left(\frac{br}{\lambda^2\epsilon}\right)$
  - Running time is independent of training data!

Experiments

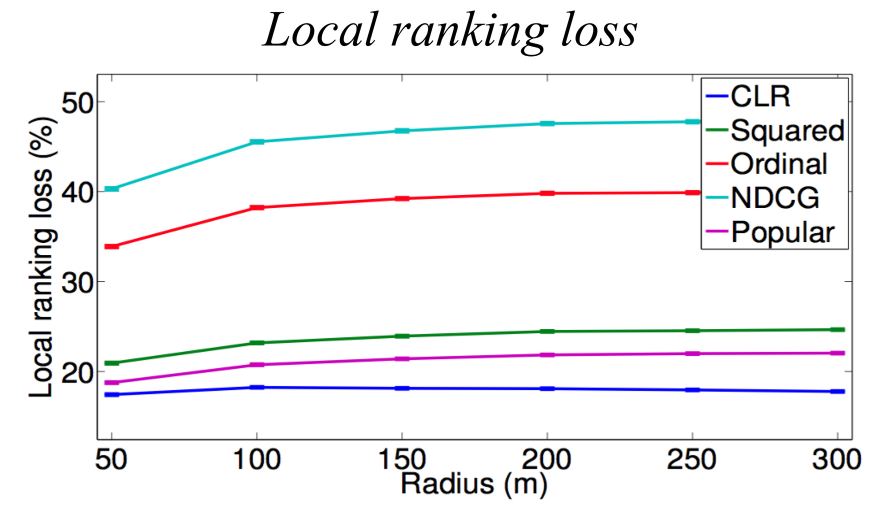
- 269,597 check-ins spanning 9 months.
  - 13,750 users, 11,750 venues.
- Baseline algorithms:
  - Popular.
  - CofiRank+NDCG.
  - CofiRank+Ordinal.
  - CofiRank+Squared.
- Evaluation metrics:
  - Local ranking loss.
  - Recall@k for k={1, 5, 10}.

Recall@k for k={1, 10}





Local ranking loss



Running time