On the Effectiveness of Linear Models for One-Class Collaborative Filtering

Suvash Sedhain[1,2], Aditya Menon[2,1], **Scott Sanner[3,1]**,
Darius Braziunas[4]

Australian National University[1]
NICTA[2]
Oregon State University[3]
Rakuten Kobo Inc [4]

Recommender Systems

- Recommender Systems
  - Objective: Present personalized items to users

- Collaborative filtering
  - De-facto method for multiuser recommender systems
  - Find people **like you** and **leverage their preferences**
  - One-class: only observe positive feedback

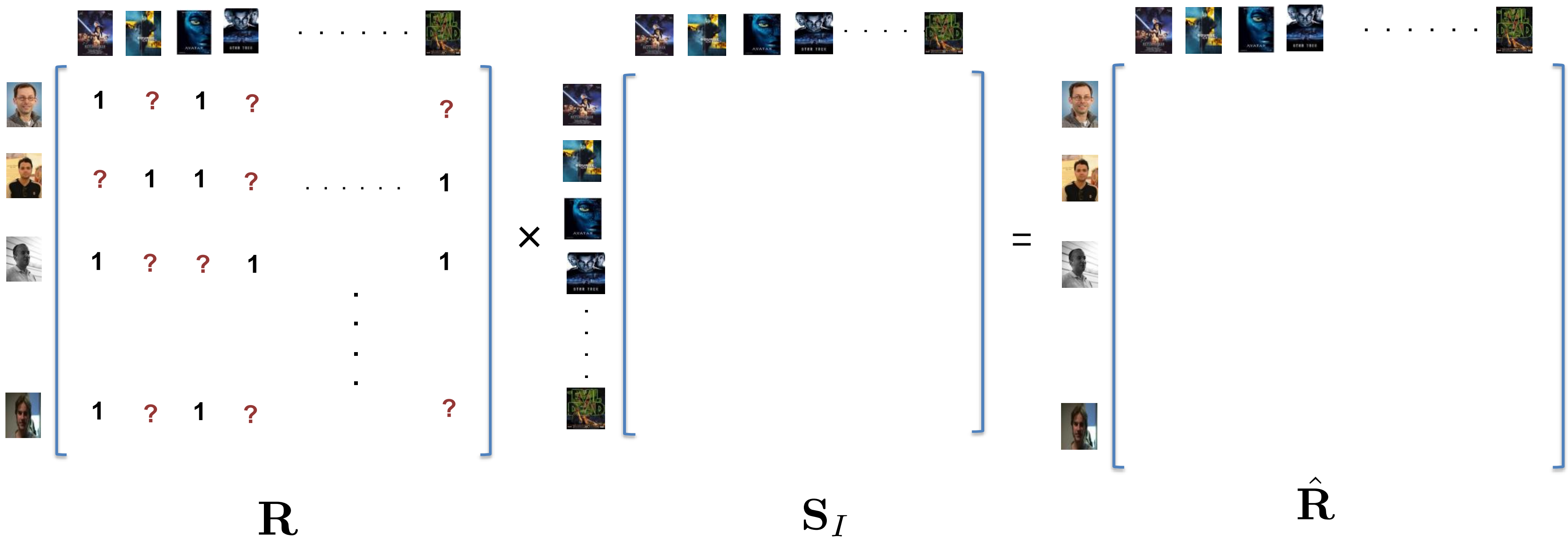
Sneak Peak: Model Proposal

- Personalized user focused linear model
- Convex
- Embarrassingly parallel
  - Each user trained individually

State-of-the-art Collaborative Filtering

- Neighborhood methods
- Matrix Factorization
- SLIM (Sparse Linear Method)

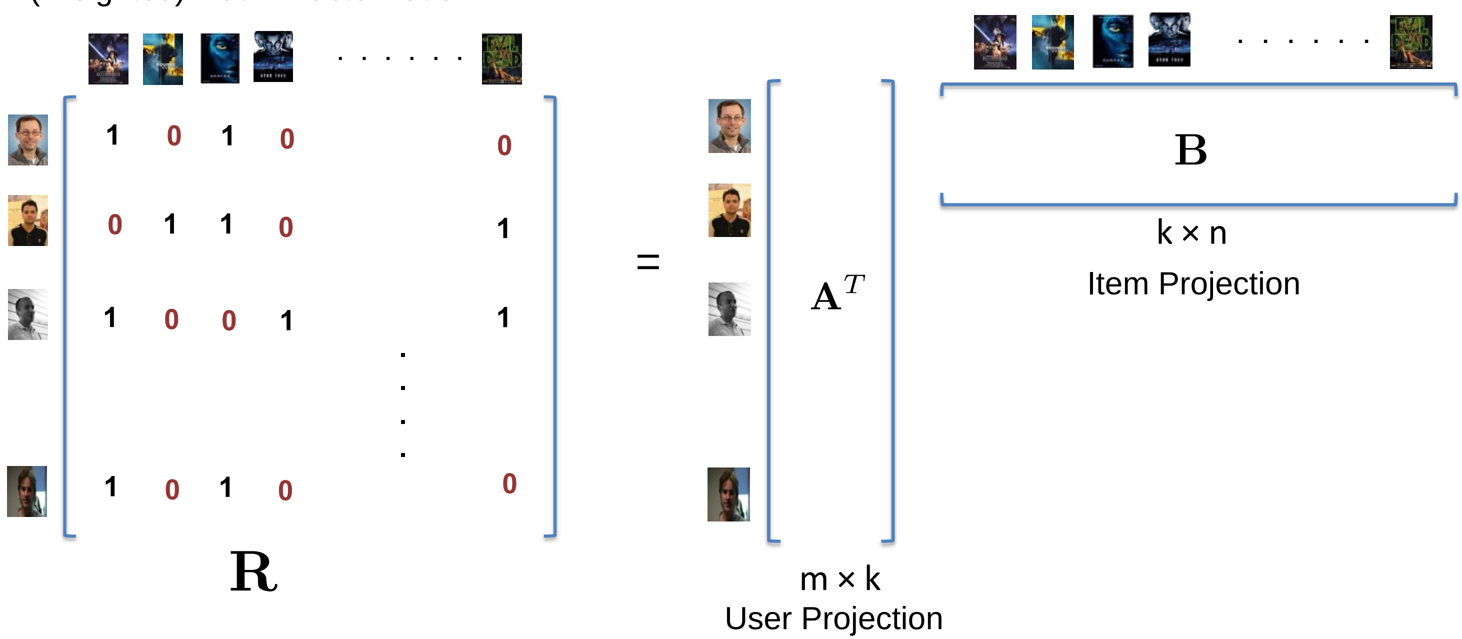
Nearest Neighbors: A Matrix View



- { Jaccard, Cosine} similarity S_I used in practice
- Keep only top k similarities
- Simple, but learning is limited

Factorization Model

(Weighted) Matrix Factorization

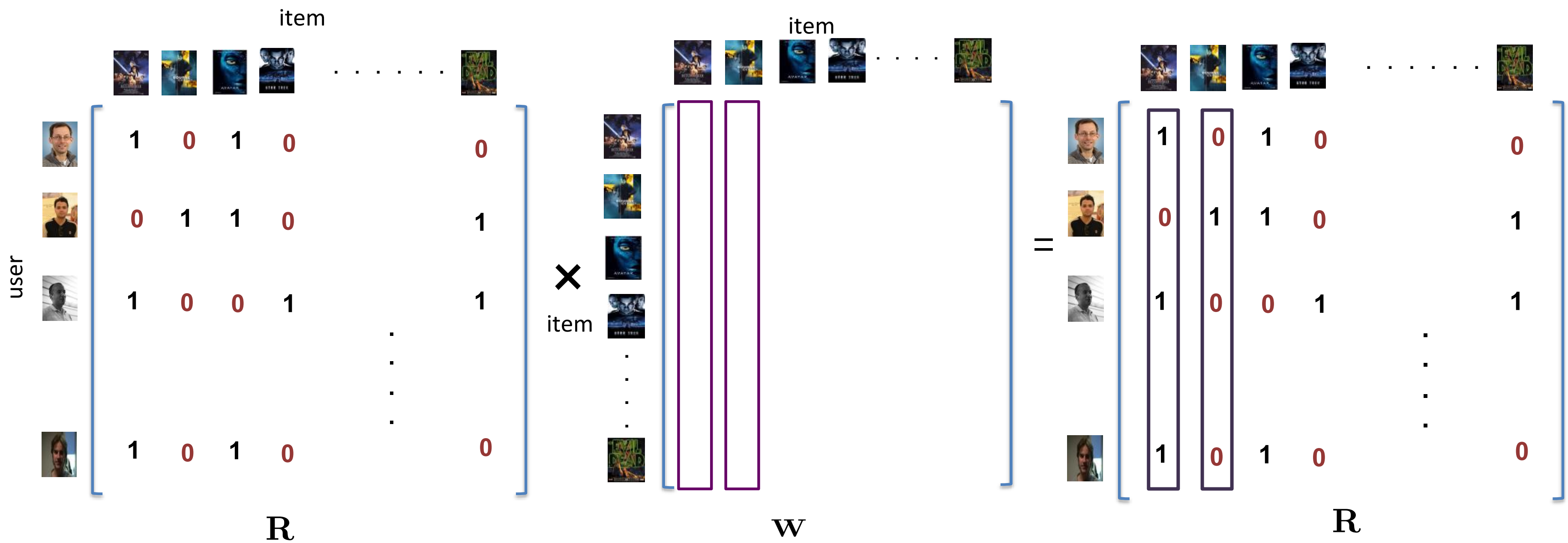


$$\min_{\mathbf{A},\mathbf{B}} \sum_{u \in \mathcal{U}, i \in \mathcal{I}} \mathbf{J}_{ui}(\mathbf{R}_{ui} - \mathbf{A}_u^T \mathbf{B}_i)^2 + \frac{\lambda}{2}(||\mathbf{A}||_F^2 + ||\mathbf{B}||_F^2)$$

$$\mathbf{J}_{ui} = [\![\mathbf{R}_{ui} = 0]\!] + \alpha \cdot [\![\mathbf{R}_{ui} > 0]\!]$$

- Works well in general, but non-convex!

SLIM



$$\min_{\mathbf{W}\in\mathcal{C}} ||\mathbf{R}-\mathbf{R}\mathbf{W}||_F^2 + \frac{\lambda}{2}||\mathbf{W}||_F^2 + \mu||\mathbf{W}||_1$$

$$\mathcal{C} = \{\mathbf{W}\in\mathbb{R}^{n\times n} \colon \mathrm{diag}(\mathbf{W}) = 0, \mathbf{W}\geq 0\}$$

- Effectively trying to learn item-to-item similarities
- Not user-focused, complicated optimization

Recommender Systems Desiderata

- Learning based
- Convex objective
- User focused
- Parallelizable

Comparison of recommendation methods for OC-CF

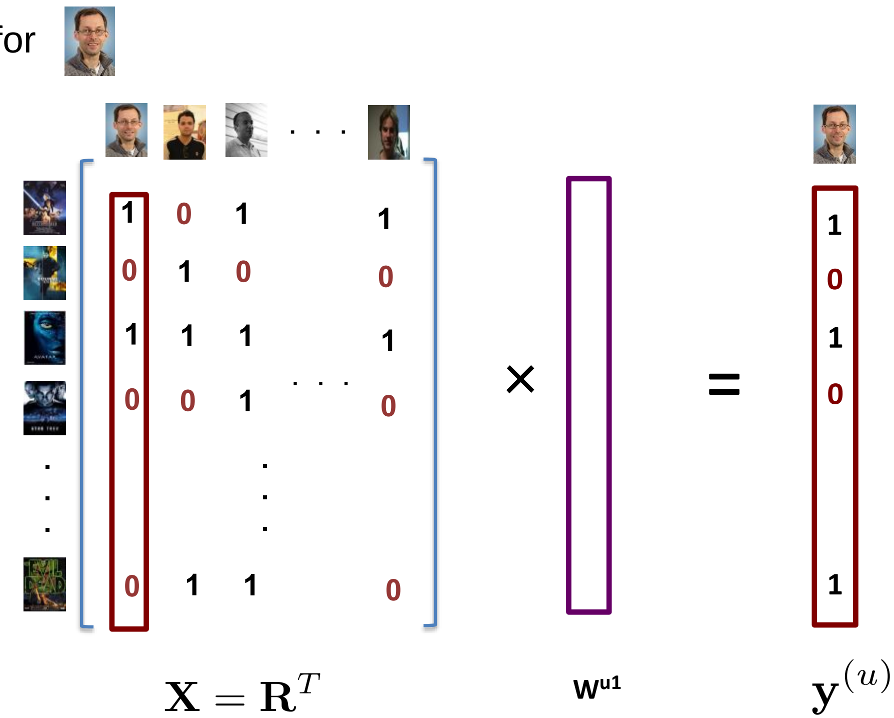
Method	Reference	Learning?	Convex?	User-focussed?	Embarrasingly Parallelisable
U-KNN	(Herlocker et al., 1999)	×	NA	✓	✓
I-KNN	(Sarwar et al., 2001)	×	NA	×	✓
PureSVD	(Cremonesi, Koren, and Turrin, 2010)	✓	✓*	✓	×
WRMF	(Pan et al., 2008)	✓	×	✓	×
LogisticMF	(Johnson, 2014)	✓	×	✓	×
BPR	(Rendle et al., 2009)	✓	×	✓*	×
SLIM	(Ning and Karypis, 2011)	✓	✓	×	✓
LRec	This paper	✓	✓	✓	✓

Outline

- Problem statement
- Background
- LRec Model
- Experiments
- Results
- Summary

LRec

Recommendation for



- Each item is a training instance
- Can be interpreted as learning user-user affinities
- Regularizer prevents from the trivial solution

$$\min_{\mathbf{W}} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \ell(\mathbf{y}_i^{(u)}, \mathbf{X}_{i:}\mathbf{w}^{(u)}) + \Omega(\mathbf{W}),$$

$$\Omega(\mathbf{W}) = \frac{\lambda}{2}||\mathbf{W}||_F^2$$

Any loss function
- Squared
- Logistic

Recommendation

Learning a model per user

Properties of LRec

- User focused
  - Recommendation as learning a model per user
- Convex objective
  - Guarantees optimal solution for the formulation
- Embarrassingly parallel
  - Each model is completely independent of other

Relationship with Existing Models

LRec

$$\min_{\mathbf{W}} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \ell(\mathbf{y}_i^{(u)}, \mathbf{X}_{i:} \mathbf{w}^{(u)}) + \Omega(\mathbf{W}),$$

- **User focused**
- L2 penalty
- Optimization
  - L2 loss
  - Logistic Loss : Liblinear (dual iff #users >> #items)

SLIM

$$\min_{\mathbf{W} \in \mathcal{C}} \sum_{i \in \mathcal{I}} \sum_{u \in \mathcal{U}} \ell(\mathbf{y}_u^{(i)}, \mathbf{X}_{u:}^{(i)} \mathbf{w}^{(i)}) + \Omega(\mathbf{W}),$$

$$\min_{\mathbf{W} \in \mathcal{C}} ||\mathbf{R} - \mathbf{R}\mathbf{W}||_F^2 + \frac{\lambda}{2} ||\mathbf{W}||_F^2 + \mu ||\mathbf{W}||_1$$

$$\mathcal{C} = \{\mathbf{W} \in \mathbb{R}^{n \times n} : \mathrm{diag}(\mathbf{W}) = 0, \mathbf{W} \geq 0\}$$

- Item focused
- Elastic-net penalty + non-negativity constraints
- Optimization:
  - Coordinate descent
  - Levy et.al. relaxed the non-negativity constraints; optimization via SGD Truncated Gradient

Relationship with Existing Models

LRec

- Learns weight matrix via classification/regression problem
  - can be interpreted as learning user-user similarities

Neighborhood models

- Computes similarities using predefined similarity metrics(eg: Cosine, Jaccard)

Relationship with Existing Models

LRec

- Learns weight matrix via classification/regression problem
  - can be interpreted as learning user-user similarities

- Convex objective
- Full rank
- Embarrassingly parallel

Matrix Factorization

$$\min_{\theta} \sum_{u \in \mathcal{U}, i \in \mathcal{I}} \mathbf{J}_{ui} \cdot (\mathbf{R}_{ui} - A_u^T B_i)^2 + \frac{\lambda}{2} \cdot (||\mathbf{A}||_F^2 + ||\mathbf{B}||_F^2)$$

If $\mathbf{J}_{ui} = 1$

$$\mathbf{B} = (\mathbf{A}\mathbf{A}^T + \lambda \mathbf{I})^{-1} \mathbf{A}\mathbf{R}$$

Recommendation

$$\hat{\mathbf{R}} = \mathbf{S}\mathbf{R}$$

Where, $\mathbf{S} = \mathbf{A}^T (\mathbf{A}\mathbf{A}^T + \lambda \mathbf{I})^{-1} \mathbf{A}$

- Non Convex objective
- Low rank
- Parallelism via distributed communication

Other Advantages of LRec

- Efficient hyper-parameter tuning for ranking
  - Validate on small subset of users
- Model can be fine-tuned per user

Other Advantages of LRec: Incorporating Side Information

- Can easily incorporate abundant item-side information

Outline

- Problem statement
- Background
- LRec Model
- Experiment & Results
- Summary

Dataset Description and Evaluation

- Movielens 1M (ML1M)
- Kobo
- Last FM (LASTFM)
- Million Song Dataset (MSD)

Dataset	m	n	$\lvert \mathbf{R}_{ui} > 0 \rvert$
ML1M	6,038	3,533	575,281
KOBO	38,868	170,394	89,815
LASTFM	992	107,398	821,011
MSD	1,019,318	384,546	48,373,586

- 10 random train-test split
  - 80%-20% split
  - For MSD, we evaluate on random 500 users
- Error bars => 95% confidence interval

Evaluation Metrics

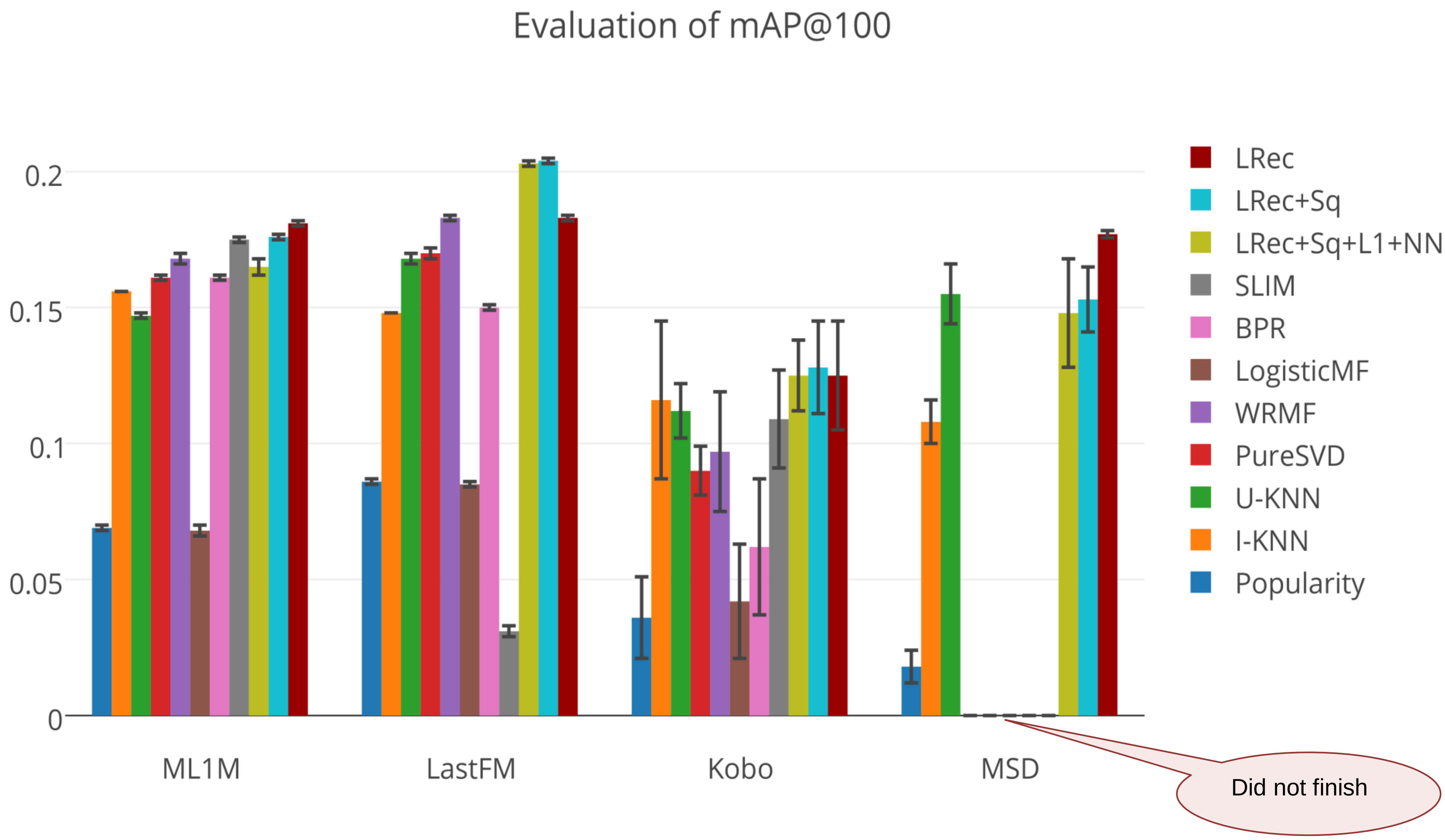
- precision@k
- mean Average Precision@100

Experiment Setup

- Baselines
  - Most Popular
  - Neighborhood
    - User KNN (U-KNN)
    - Item KNN (I-KNN)
  - Matrix Factorization
    - PureSVD
    - WRMF
    - LogisticMF
    - Bayesian Personalized Ranking (BPR)

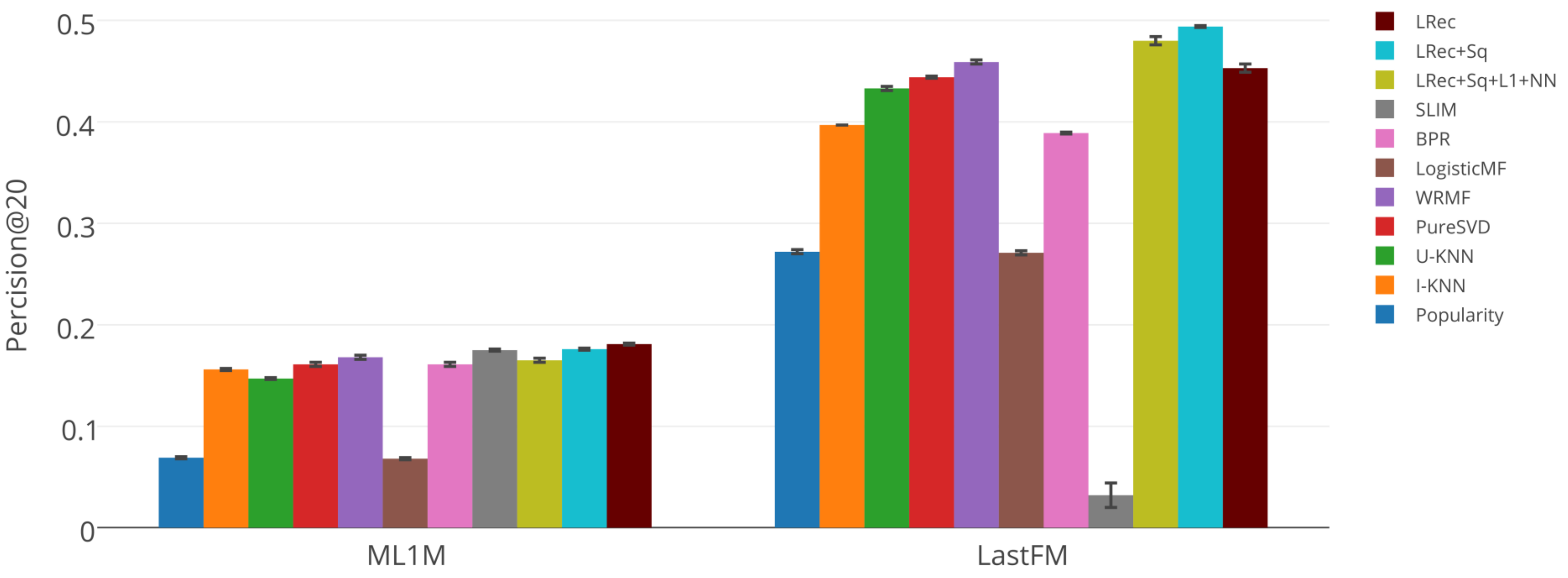
- SLIM
- LREC
  - Elastic Net Lrec + Non-Negativity (Lrec + Sq + L_1+ NN)
  - Squared Loss LRec (Lrec + Sq)
  - Logistic Loss LRec (LRec)

Results



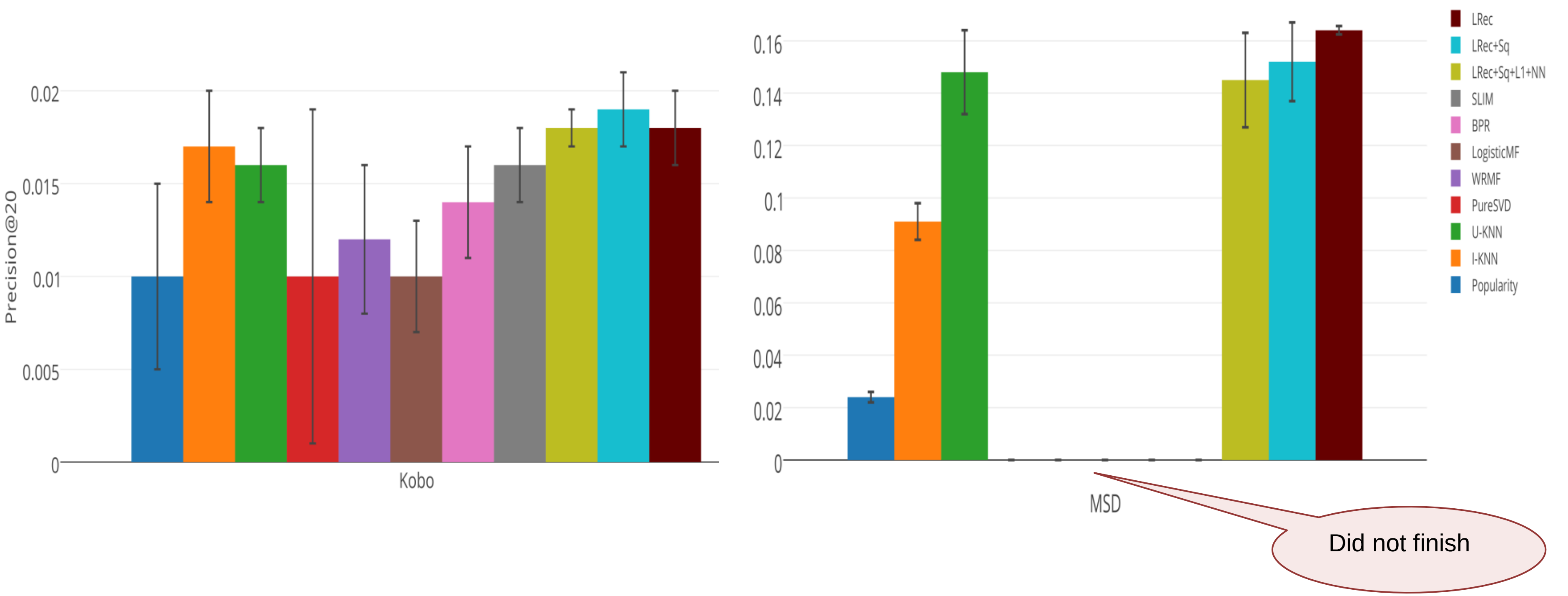
Evaluation of mAP@100
0.2
0.15
0.1
0.05
0
ML1M
LastFM
Kobo
MSD
LRec
LRec+Sq
LRec+Sq+L1+NN
SLIM
BPR
LogisticMF
WRMF
PureSVD
U-KNN
I-KNN
Popularity
Did not finish

Results



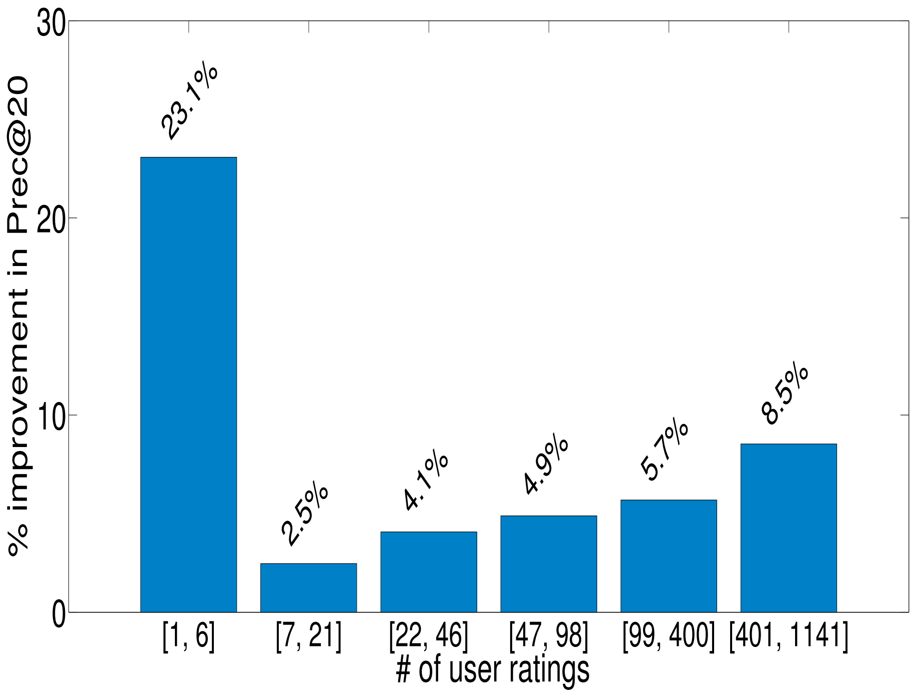
Precision@20 on ML1M and LastFM dataset

Results



Precision@20 on Kobo and LastFM dataset

Performance Evaluation



Users segmented by the number of observation

% improvement over WRMF on ML1M dataset

Case Study

Recommendation from WRMF vs LRec

Preferred training movies	WRMF recommendations	LRec recommendations	Preferred test movies
• Day the Earth Stood Still, The	• Planet of the Apes	• **Them!**	• Blob, The
• Forbidden Planet	• Thing, The	• Godzilla (Gojira)	• Them!
• Kronos	• Night of the Living Dead	• **Blob, The**	• It Came from Outer Space
• Tarantula	• Star Trek: The Wrath of Khan	• 20,000 Leagues Under the Sea	
• Thing From Another World, The	• Fly, The	• Soylent Green	
• War of the Worlds, The	• Alien	• Village of the Damned	
• It Came from Beneath the Sea	• Dark City	• Metropolis	
• Invasion of the Body Snatchers	• Star Trek IV: The Voyage Home	• Quatermass and the Pit	
• Earth Vs. the Flying Saucers	• 2001: A Space Odyssey	• **It Came from Outer Space**	
• It Conquered the World	• Gattaca	• Plan 9 from Outer Space	

LRec is more personalized

Summary

- LRec
  - Personalized user focused linear recommender
  - Convex objective
  - Embarrassingly parallel
- Future work
  - Further scale LRec
    - Computational
    - Memory footprint

Thanks