KDD 2017

ACM SIGKDD Conference on Knowledge Discovery and Data Mining

Discrete Content-aware Matrix Factorization

Rui Liu, University of Electronic Science and Technology of China

Joint work with
Defu Lian (UESTC), Yong Ge (UA), Kai Zheng (UESTC),
Xing Xie (Microsoft Research) and Longbing Cao (UTS)

Outline

- Motivation
- Proposed framework
- Experiment
- Conclusion

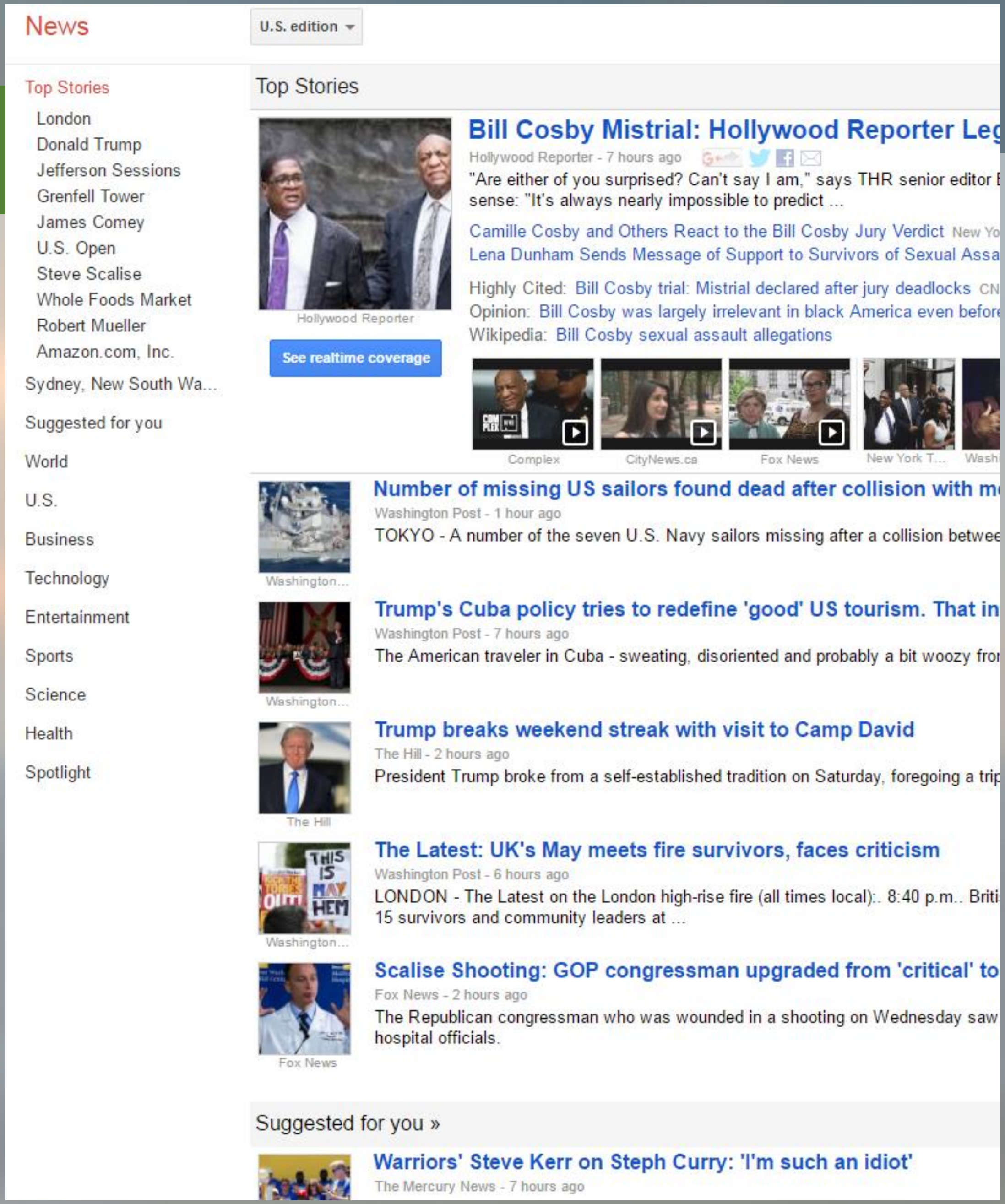
Motivation

Google News Recommendation

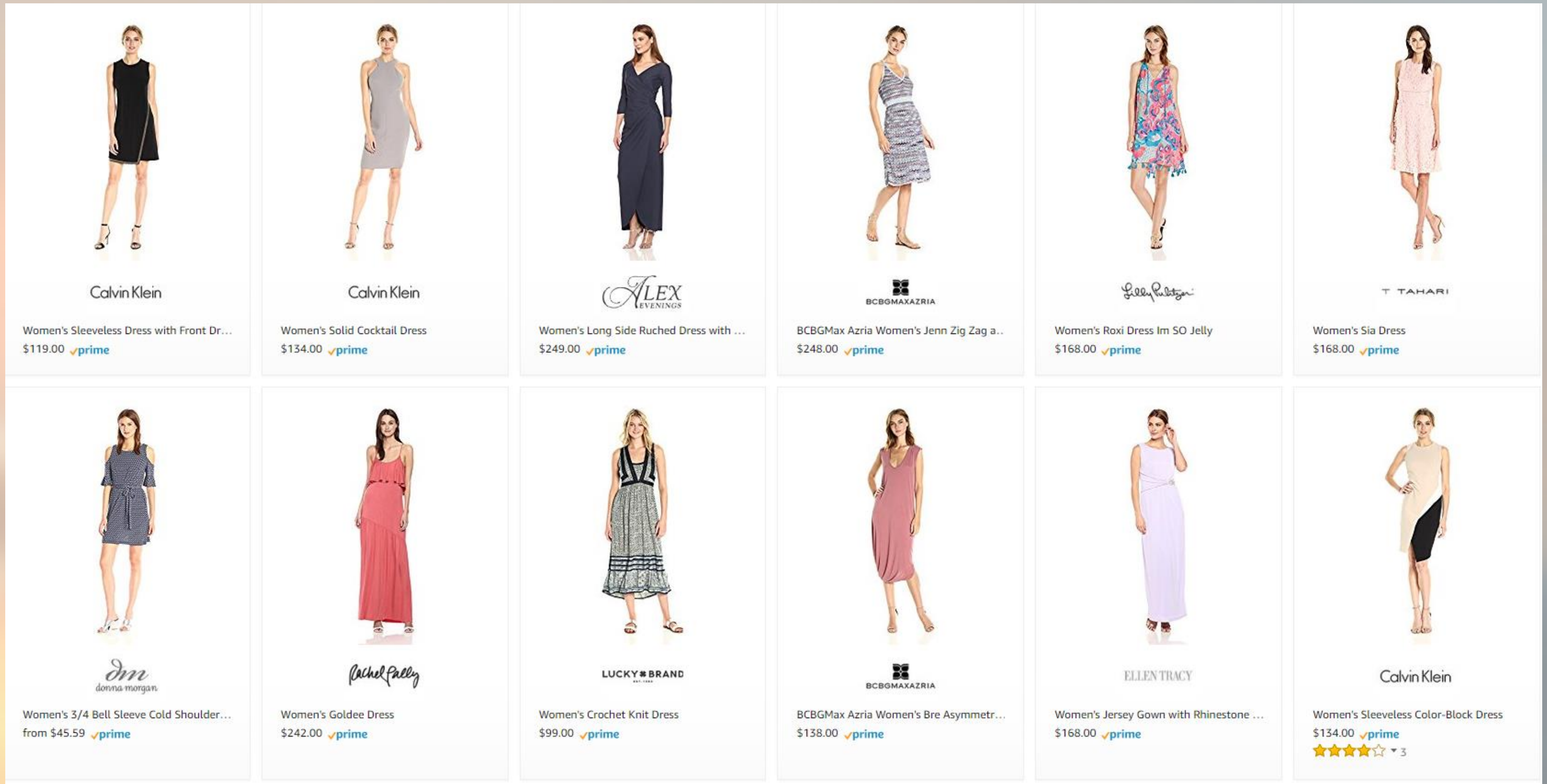
Tasks: recommends new articles based on click and search History

Scale:

- millions of users
- millions of articles



Motivation



Amazon Product Recommendation
Tasks: recommends new articles based on click and search History

Scale:
- 300 million users
- 480 million products

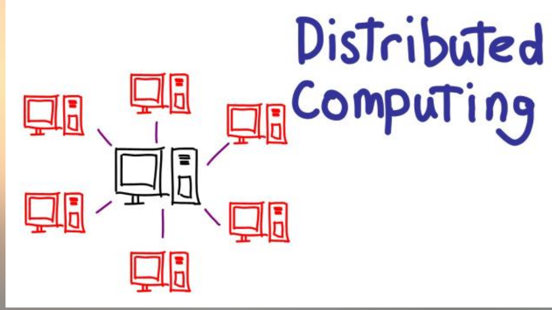
Motivation

- How to generate immediate recommendation?
  - Pre-computing top-k preferred items for users?

User interests are evolving

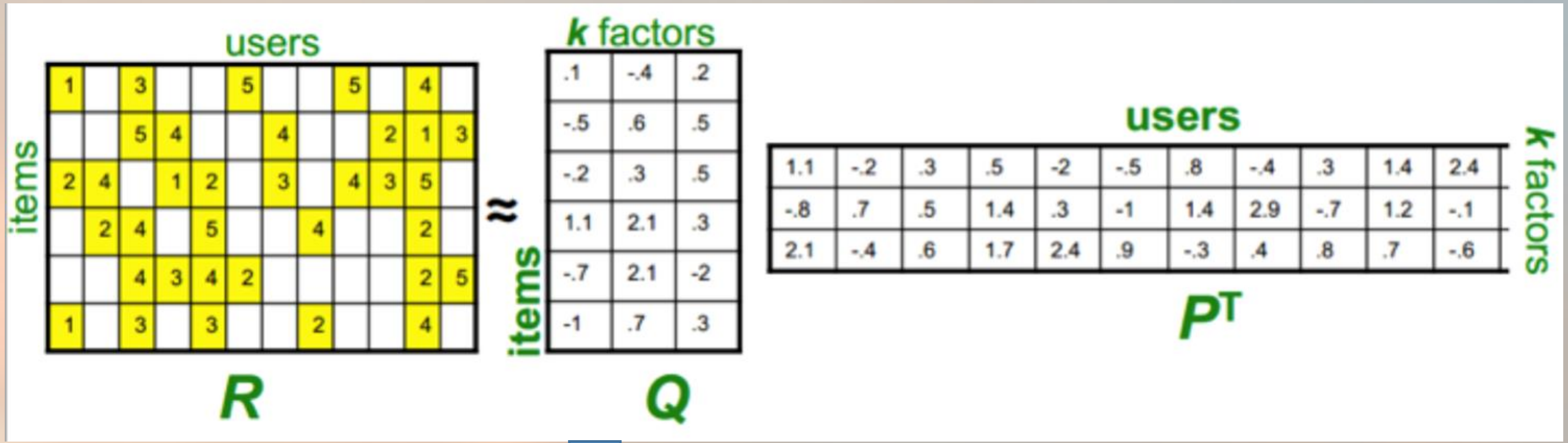
New items are emerging

  - Distributed/parallel computing

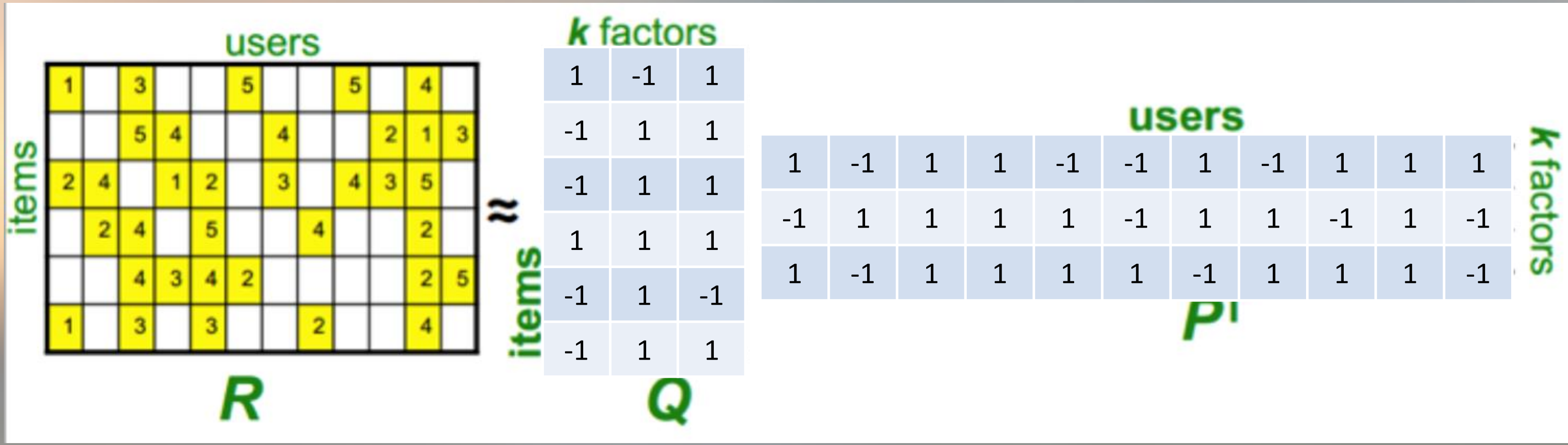


A lot of machines

Economic and Effective Way



$$\mathrm{sign}(x) = \begin{cases} 1, x > 0 \\ -1, x < 0 \end{cases}$$



1	-1	1
-1	1	1
-1	1	1
1	1	1
-1	1	-1
-1	1	1

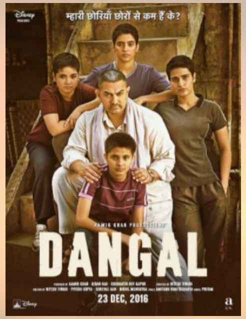
1	-1	1	1	-1	-1	1	-1	1	1	1
-1	1	1	1	1	-1	1	1	-1	1	-1
1	-1	1	1	1	1	-1	1	1	1	-1

Economic and Effective Way

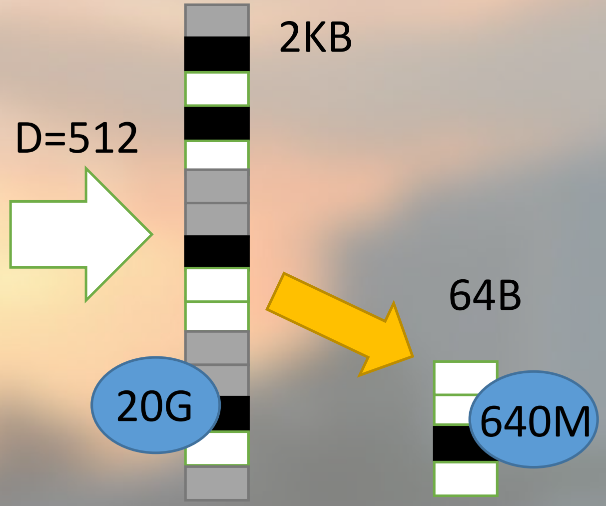
$$< p_u, q_i > = \sum_{d=1}^{D} p_{ud} q_{id} \qquad p_u, q_i \in R^D$$

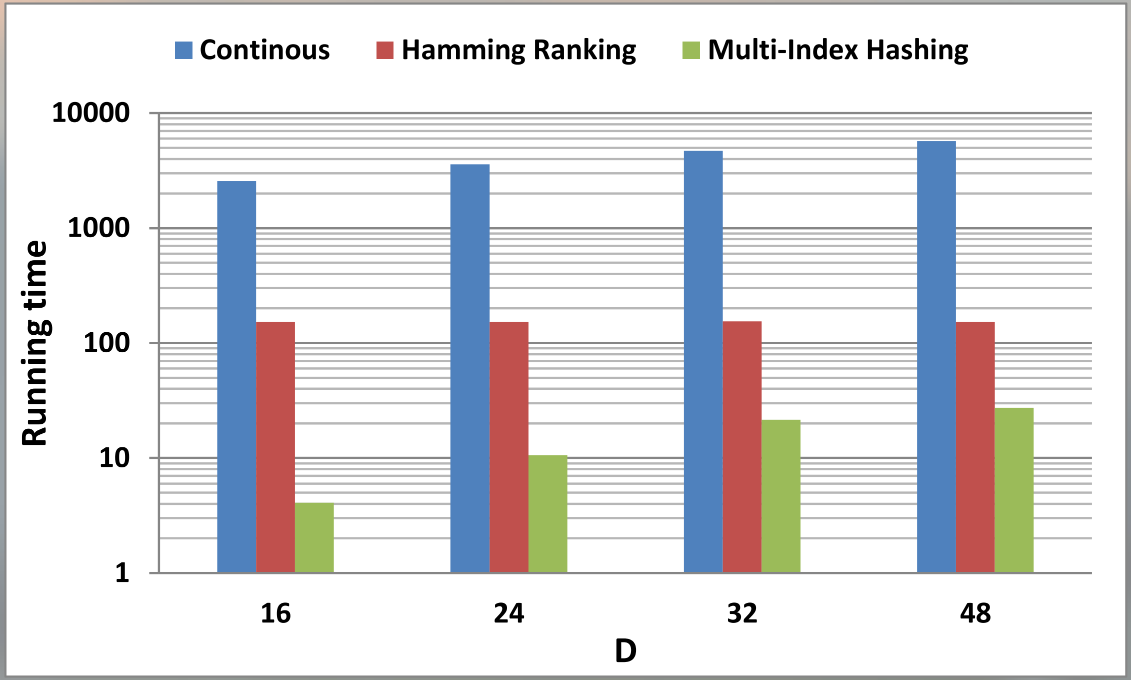
$$< \phi_u, \psi_i > = 2H(\phi_u, \psi_i) - D \quad \phi_u, \psi_i \in \{\pm 1\}^D$$

ϕ_u	1	1	0	1	1	1	0	0
ψ_i	1	1	1	1	0	1	1	0
				XOR				
$H(\phi_u, \psi_i)$	0	0	1	0	1	0	1	0



10 Million products





Related Work

- Dimension Reduction Techniques for Recommendation Systems
  - Matrix Factorization (KDD'11)
    - Map users and items into low-dimension latent space
  - Content-aware Matrix Factorization (RecSys'13, ICDM'15)
    - Take content information into account, help solve cold-start problem
  - Weighted Regularized Matrix Factorization
    - Work well in collaborative filtering for implicit feedback
- Distributed Computing Techniques for RS
  - Large-scale Parallel Collaborative Filtering
    - Parallel update to improve efficiency

Related Work

- Learning Hashing Codes for Recommendation Systems
  - Binary Code learning method for Collaborative Filtering (KDD’12)
    - Obtained binary codes preserve preference of users
  - Preference Preserving Hashing (SIGIR’14)
    - Come up with a novel quantization algorithm
  - Discrete Collaborative Filtering (SIGIR’16)
    - Tackle discrete optimization directly and efficiently

1. Cold-start problem
2. Implicit feedback
3. Classification

Task

- DCMF: a joint framework
  - Incorporate content information into the total framework
    - Solve cold-start problem
  - Combine the Weighted Regularized Matrix Factorization
    - Solve for implicit feedback
  - Find a solution for logit loss
    - Solve for classification
  - A direct discrete optimization model
    - Update efficiently

Outline

- Motivation
- Proposed framework
- Experiment
- Conclusion

Total Framework

$$\min_{\substack{\mathbf{\Phi},\mathbf{\Psi},\mathbf{P},\mathbf{Q}\\ \mathbf{U},\mathbf{V}}} \sum_{i,j\in\Omega} \ell(r_{ij},\phi_i'\psi_j) + \alpha_1\|\mathbf{\Phi}-\mathbf{P}\|_F^2 + \alpha_2\|\mathbf{\Psi}-\mathbf{Q}\|_F^2$$

balance and decorrelation constraints

$$+\lambda_1\|\mathbf{\Phi}-\mathbf{XU}\|_F^2 + \lambda_2\|\mathbf{\Psi}-\mathbf{YV}\|_F^2 + \gamma_1\|\mathbf{U}\|_F^2 + \gamma_2\|\mathbf{V}\|_F^2 + \beta\sum_{i,j}(\phi_i'\psi_j)^2$$

- *Regression task:*

  $\ell(r_{ij},\phi_i'\psi_j) = (r_{ij}-\phi_i'\psi_j)^2$

- *Classification task:*

  $\ell(r_{ij},\phi_i'\psi_j) = \log\left(1+e^{-r_{ij}\phi_i'\psi_j}\right)$

(challenge 3)

Content-aware term

Take content information of users and items into account (challenge 1)

Regularization

Interaction regularization

Make use of implicit feedback (challenge 2)

$$s.t.\ \mathbf{\Phi}\in\{\pm1\}^{M\times D}, \mathbf{\Psi}\in\{\pm1\}^{N\times D}$$

$$\mathbf{1}'\mathbf{P}=0, \mathbf{1}'\mathbf{Q}=0, \mathbf{P}'\mathbf{P}=M\mathbf{I}, \mathbf{Q}'\mathbf{Q}=N\mathbf{I}$$

Alternative Update Rule

$$\phi_{id}^* = sgn\left(K\left(\hat{\phi}_{id}, \phi_{id}\right)\right)$$

$$K(x,y) = \begin{cases} x, x \neq 0 \\ y, x = 0 \end{cases}$$

$$\psi_{jd}^* = sgn\left(K\left(\hat{\psi}_{jd}, \psi_{jd}\right)\right)$$

$$\hat{\phi}_{id} = \sum_{j \in \mathbb{I}_i} \underline{(r_{ij} - \hat{r}_{ij} + \phi_{id}\psi_{jd})\psi_{jd}} + \alpha_1 p_{id}$$

$$(r_{ij}/4 - \lambda(\hat{r}_{ij})\hat{r}_{ij} + \lambda(\hat{r}_{ij})\phi_{id}\psi_{jd})\psi_{jd}$$

$$+\lambda_1 \boldsymbol{u}'_d \boldsymbol{x}_i - \beta \phi'_i \boldsymbol{\Psi}' \psi_d + \beta N \phi_{id}$$

$$\hat{\psi}_{jd} = \sum_{i \in \mathbb{I}_j} \underline{(r_{ij} - \hat{r}_{ij} + \phi_{id}\psi_{jd})\phi_{id}} + \alpha_2 q_{jd}$$

$$(r_{ij}/4 - \lambda(\hat{r}_{ij})\hat{r}_{ij} + \lambda(\hat{r}_{ij})\phi_{id}\psi_{jd})\phi_{id}$$

$$+\lambda_2 \boldsymbol{v}'_d \boldsymbol{y}_j - \beta \psi'_j \boldsymbol{\Phi}' \phi_d + \beta M \psi_{jd}$$

$$\mathbf{P} = \sqrt{M}\left[\boldsymbol{S_P}, \widehat{\boldsymbol{S}}_{\boldsymbol{P}}\right]\left[\boldsymbol{T_P}, \widehat{\boldsymbol{T}}_{\boldsymbol{P}}\right]'$$

$$\boldsymbol{S_P} \in \mathbb{R}^{M \times \tilde{D}} \quad \widehat{\boldsymbol{S}}_{\boldsymbol{P}} \in \mathbb{R}^{M \times (D - \tilde{D})}$$

$$\boldsymbol{T_P} \in \mathbb{R}^{D \times \tilde{D}} \quad \widehat{\boldsymbol{T}}_{\boldsymbol{P}} \in \mathbb{R}^{D \times (D - \tilde{D})}$$

$$\mathbf{Q} = \sqrt{N}\left[\boldsymbol{S_Q}, \widehat{\boldsymbol{S}}_{\boldsymbol{Q}}\right]\left[\boldsymbol{T_Q}, \widehat{\boldsymbol{T}}_{\boldsymbol{Q}}\right]'$$

$$\boldsymbol{S_Q} \in \mathbb{R}^{N \times \tilde{D}} \quad \widehat{\boldsymbol{S}}_{\boldsymbol{Q}} \in \mathbb{R}^{N \times (D - \tilde{D})}$$

$$\boldsymbol{T_Q} \in \mathbb{R}^{D \times \tilde{D}} \quad \widehat{\boldsymbol{T}}_{\boldsymbol{Q}} \in \mathbb{R}^{D \times (D - \tilde{D})}$$

$$\mathbf{U} = \left(\mathbf{X}'\mathbf{X} + \frac{\gamma_1}{\lambda_1}\mathbf{I}_F\right)^{-1} \mathbf{X}'\mathbf{\Phi}$$

$$\mathbf{V} = \left(\mathbf{Y}'\mathbf{Y} + \frac{\gamma_2}{\lambda_2}\mathbf{I}_L\right)^{-1} \mathbf{Y}'\mathbf{\Psi}$$

Outline

- Motivation
- Proposed framework
- Experiment
- Conclusion

Dataset and Metric

- ✓ MovieLens, classic MovieLens 10M dataset
- ✓ Yelp, the latest Yelp Challenge dataset
- ✓ Amazon, a subset of product reviews and metadata for Amazon books

Data statistics

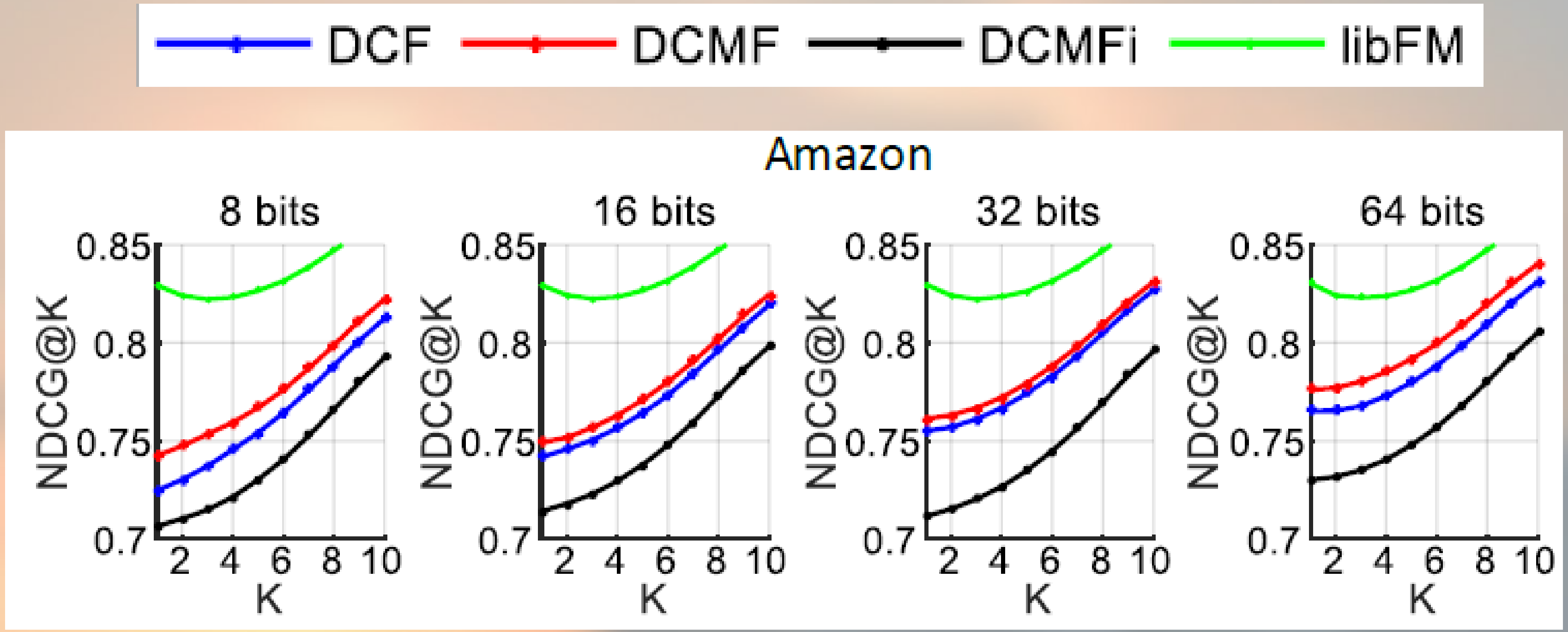
Dataset	#users	#items	#ratings	Density
MovieLens	69,838	8,940	9,983,758	1.60%
Yelp	13,679	12,922	640,143	0.36%
Amazon	35,151	33,195	1,732,060	0.15%

- ✓ Metric: NDCG for regression
  MPR for classification

Baselines

- DCF
  - Hashing-based collaborative filtering.
  - Outperforms almost all two-stage binary code learning methods for collaborative filtering including BCCF, PPH, CH.
- libFM:
  - Feature-based recommendation system.
  - Has achieved the best sole-model for the track I challenge-link prediction-of KDD-Cup 2012.
  - Supports both regression and classification tasks of recommendation.

Comparison with baselines - Regression



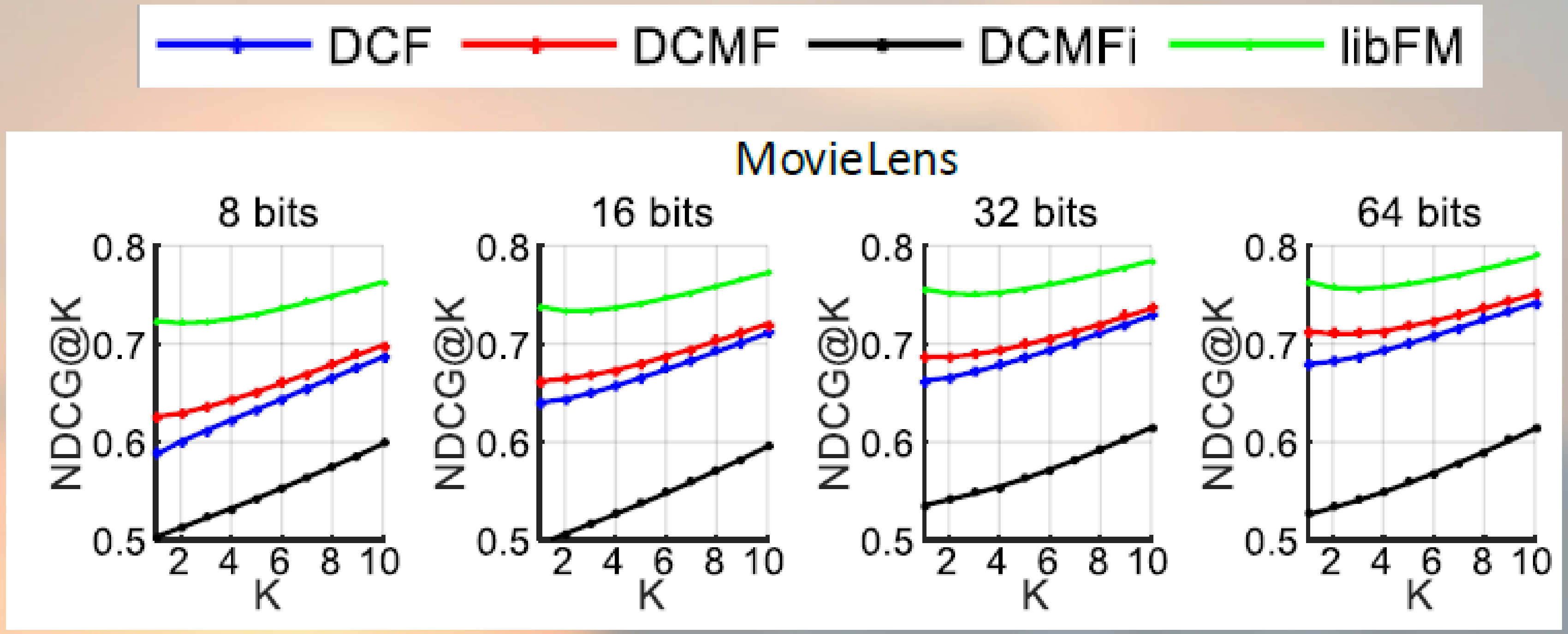
In-matrix regression

- Add content information

  → performance ↑

- Effective discrete optimization algorithm

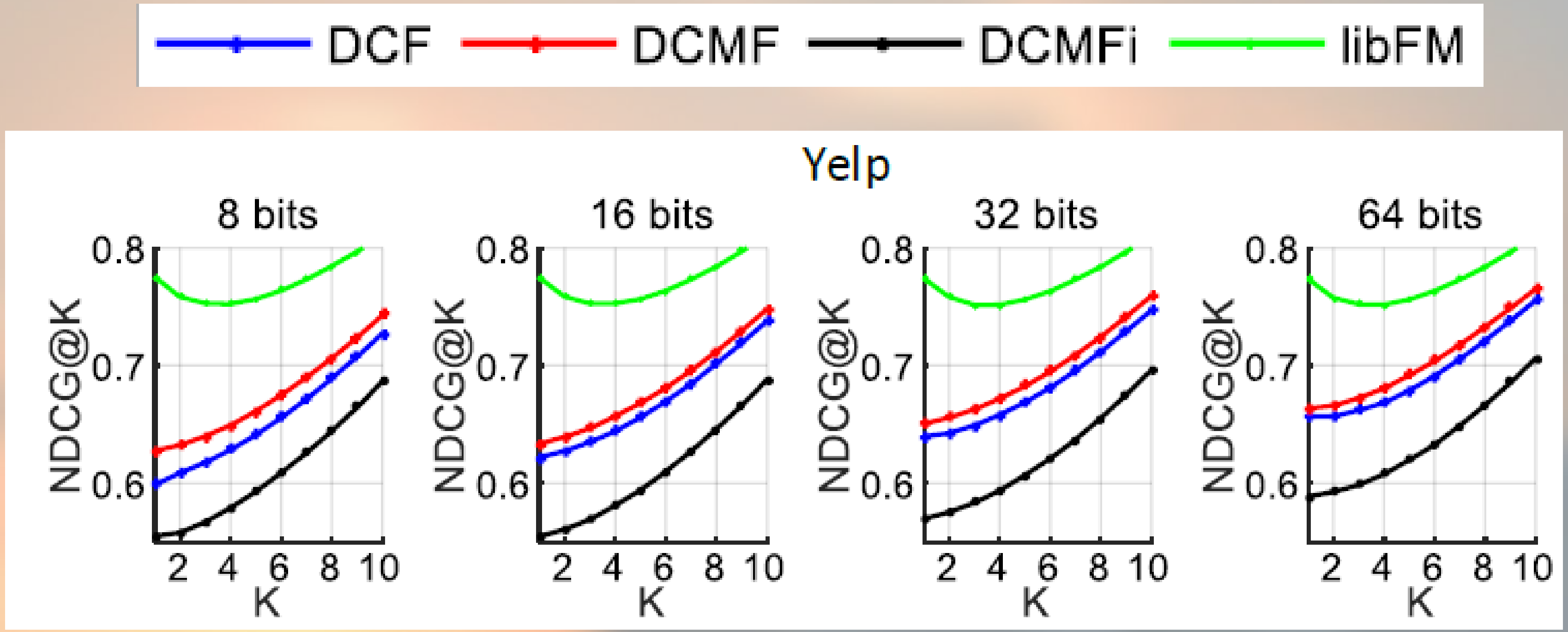
Comparison with baselines - Regression



In-matrix regression

- Add content information → performance ↑
- Effective discrete optimization algorithm

Comparison with baselines - Regression

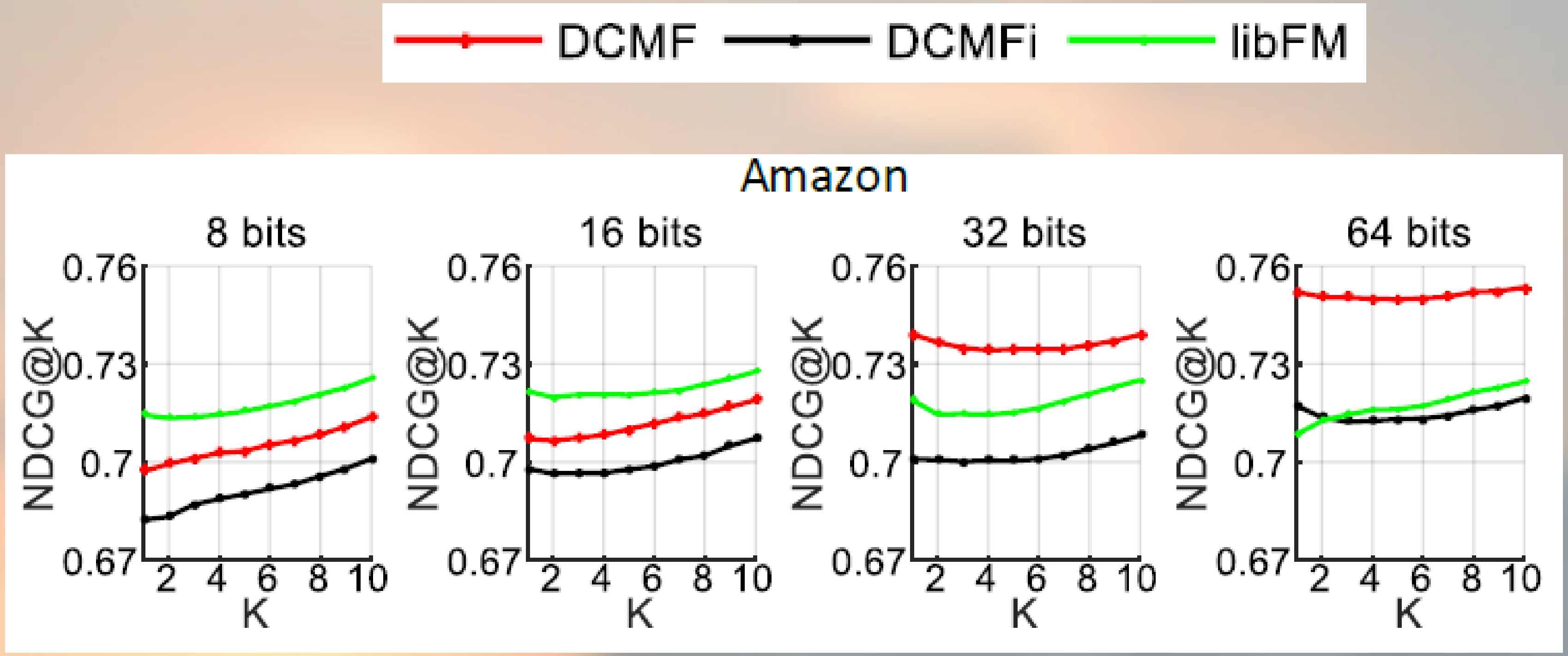


In-matrix regression

- Add content information

  → performance ↑
- Effective discrete optimization algorithm

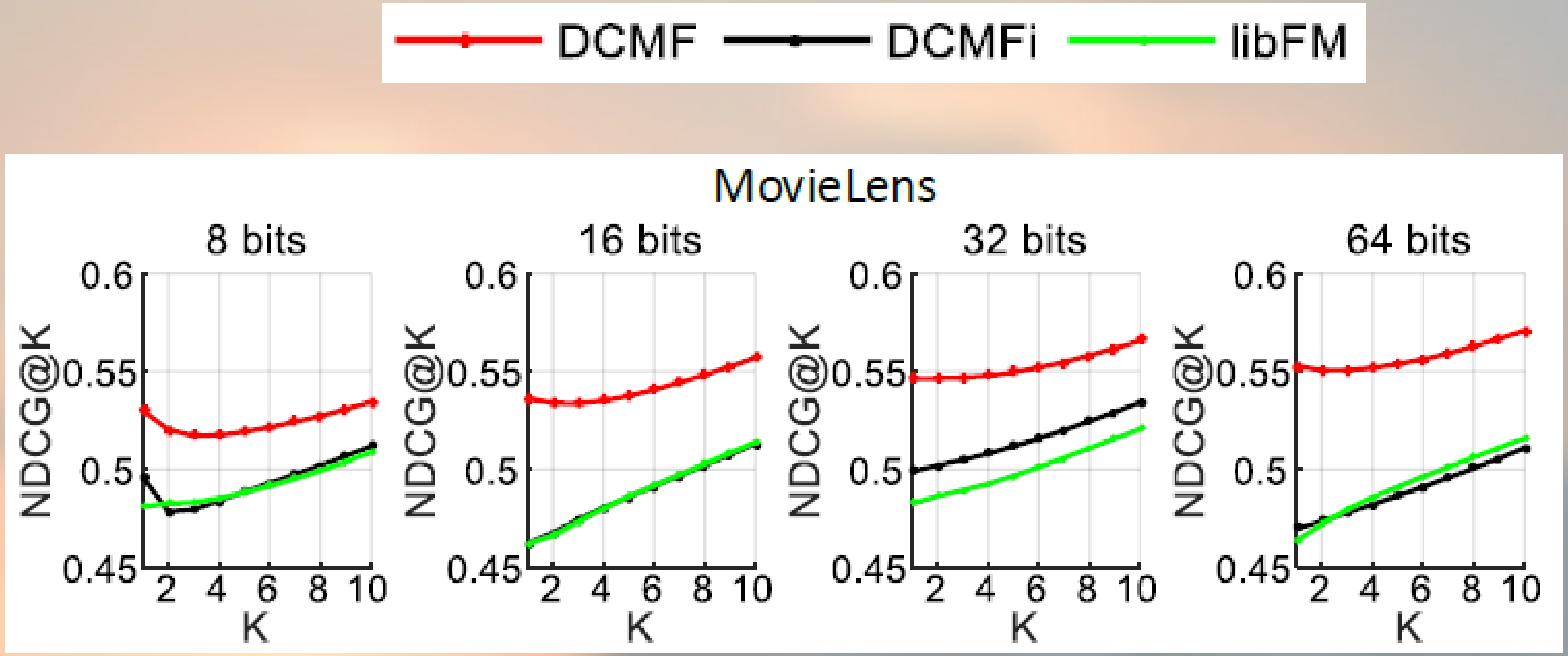
Comparison with baselines - Regression



Out-matrix regression

- Address cold-start problem well
- Effective discrete optimization algorithm

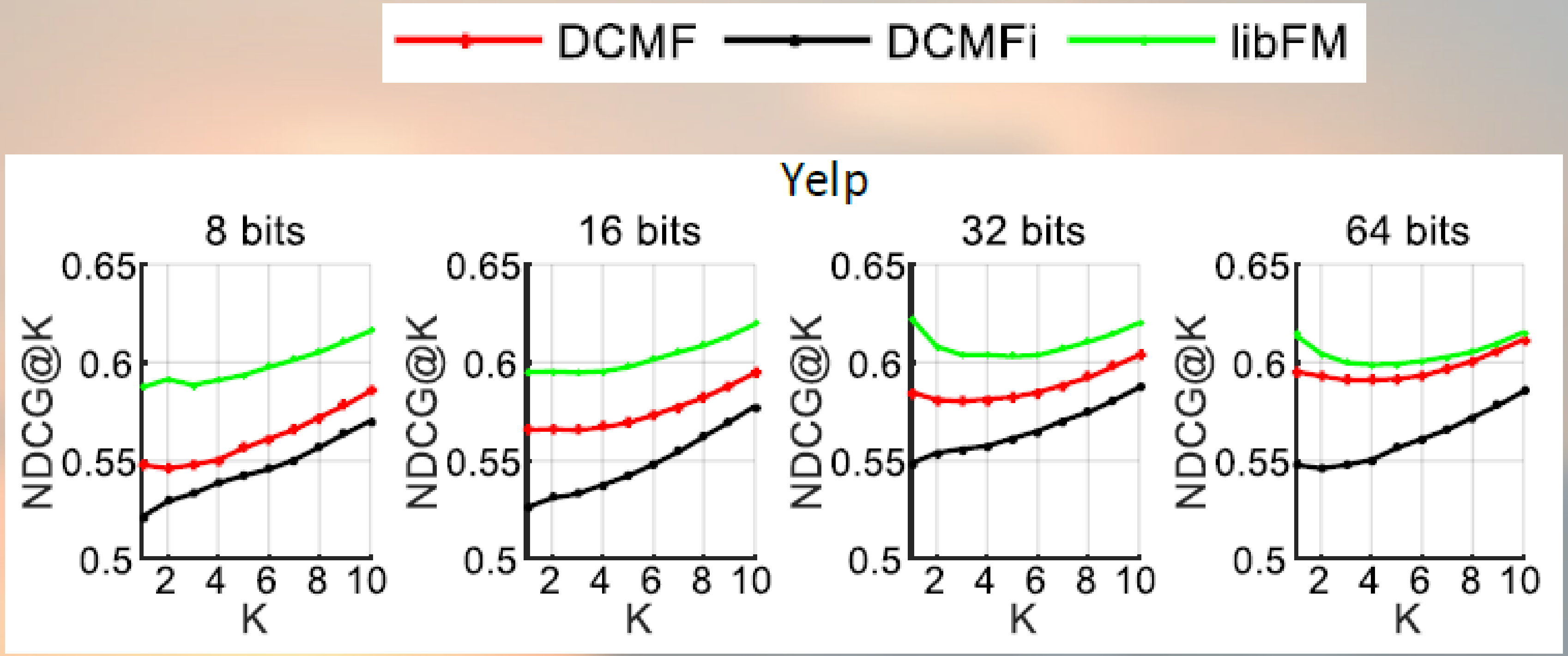
Comparison with baselines - Regression



Out-matrix regression

- Address cold-start problem well
- Effective discrete optimization algorithm

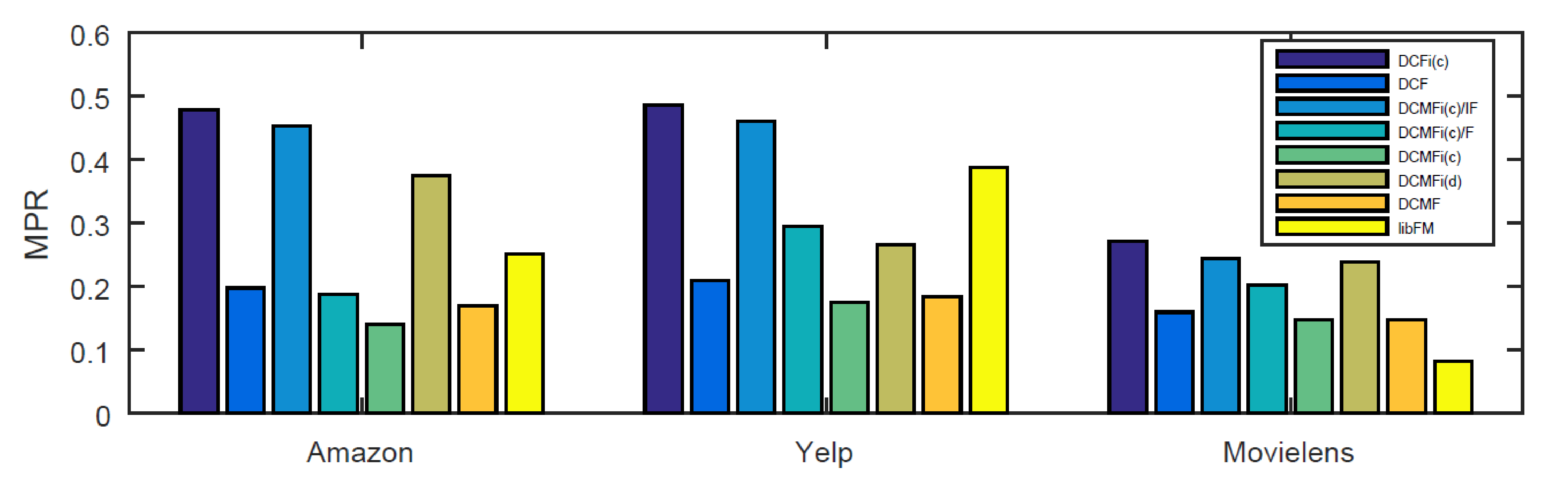
Comparison with baselines - Regression



Out-matrix regression

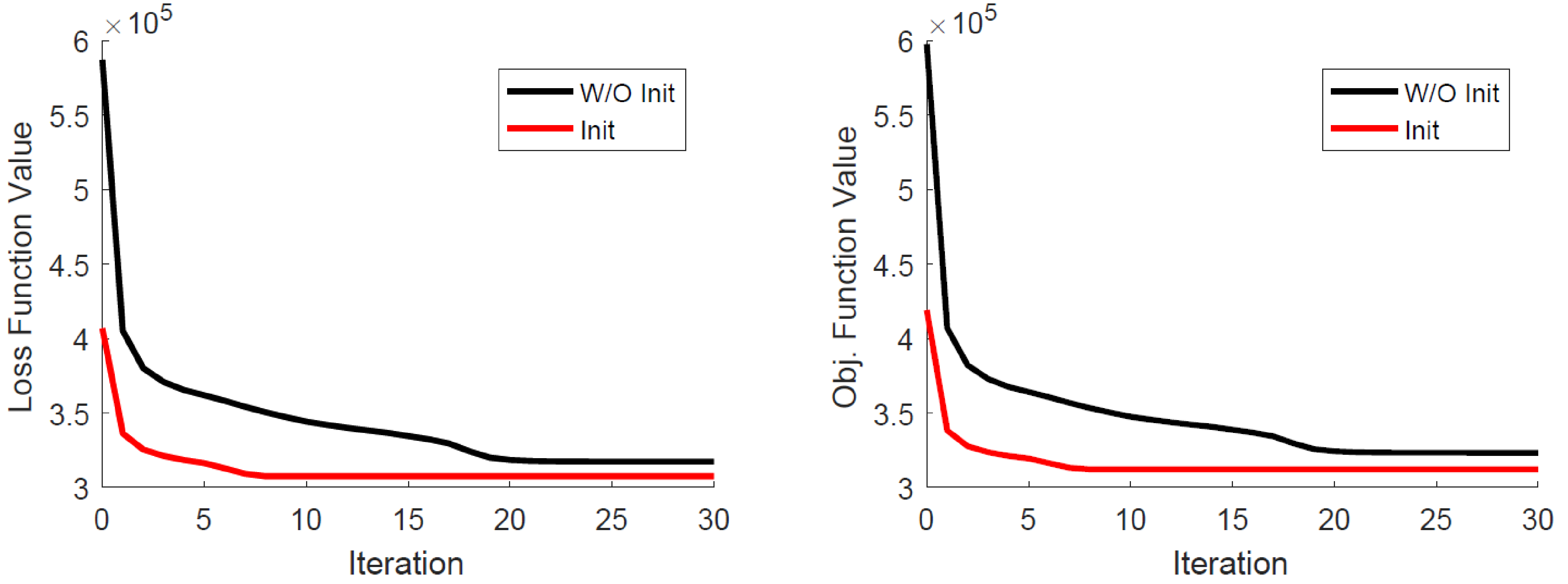
- Address cold-start problem well
- Effective discrete optimization algorithm

Comparison with baselines - Classification



- ✓ Effectiveness of content information and iteration regularization
- ✓ Benefit of the use of logit loss
- ✓ Superiority of DCMF to DCF
- ✓ Validity of the proposed discrete optimization algorithm
- ✓ Superiority of DCMF to libFM on relatively sparse datasets

Convergence Study



(c) Classification-Loss Function Value (d) Classification-Objective Function Value

Outline

- Motivation
- Proposed framework
- Experiment
- Conclusion

Conclusion

- Proposed a new framework called DCMF to hash users and items with content information in both regression and classification tasks.
- Developed an efficient discrete optimization algorithm for tackling discretized constraints as well as interaction regularization.
- Proved superiority in 3 public datasets

Thank you!

Rui Liu, UESTC
ruiliu011@gmail.com