



Fast Adaptively Weighted Matrix Factorization for Recommendation with Implicit Feedback


Speaker: Liqi Yang

Date: 2020/04/21


1. Introduction & background

Implicit feedback recommendation problem

- User set U , item set I , User-item interaction matrix X
- recommend items that users may consume or like

Four blue circular icons representing users are positioned to the left of the matrix.

1	?	0	?
?	?	1	?
?	0	?	1
0	1	0	?

Four document icons representing items are positioned below the matrix.

- **Implicit feedback**

- Derived from **monitoring and analyzing user's activities**
- E.g. click, view, buy, add to cart, delete, etc
- Does not require any active user involvement

1. Introduction & background

Implicit feedback scenarios

- One class problem
 - Only positive feedback are observed
 - the unobserved user item feedback data (e.g. a user has not clicked an item yet)
a mixture :
 - real negative feedback (i.e. a user does not like it)
 - missing values (i.e. a user just does not know it).
- ✓ un-observed data as negative (dislike) but downweighting their confidence

1. Introduction & background

Implicit feedback scenarios

- One class problem
 - Only positive feedback are observed
 - the unobserved user item feedback data (e.g. a user has not clicked an item yet)
a mixture :
 - real negative feedback (i.e. a user does not like it)
 - missing values (i.e. a user just does not know it).
 - ✓ un-observed data as negative (dislike) but downweighting their confidence
 - User-item matrix is sparse
- Motivation:
- a) assign adaptively confidence weights
 - b) handle the massive volume of the unobserved data efficiently



1. Introduction & background

How to assign adaptively confidence weight ?

- The data with larger exposure are more reliable



- Exposure-based method

1. Introduction & background

How to assign adaptively confidence weight ?

- The data with larger exposure are more reliable



- Exposure-based Method
- recent literatures in social science claim a phenomenon:
 - each of us belongs to some information-sharing communities
 - users exposure exhibit correlations when they belong to common communities



capturing latent correlations between users' exposure

1. Introduction & background

How to assign adaptively confidence weight ?

- The data with larger exposure are more reliable



- Exposure-based Method
- recent literatures in social science claim a phenomenon:
 - each of us belongs to some information-sharing communities
 - users exposure exhibit correlations when they belong to common communities



capturing latent correlations between users' exposure

Motivation

2. Preliminaries

MF

$$n * n \rightarrow n * k \quad k * n$$

		Item											
		W	X	Y	Z			W	X	Y	Z		
User	A		4.5	2.0		A	=	X	1.5	1.2	1.0	0.8	
	B	4.0		3.5					X	1.7	0.6	1.1	0.4
	C		5.0		2.0								
	D		3.5	4.0	1.0								
Rating Matrix						User Matrix		Item Matrix					

$$\min_{p^*, q^*} \sum_{(u,i) \in k} (r_{ui} - q_i^T p_u)^2 + \lambda(\|q_i\|^2) + \|p_u\|^2)$$



2. Preliminaries

Exposure-based matrix factorization (EXMF)

- Problem definition
 - user set U (n users) ,
 - item set I (m items),
 - the implicit feedback data $n * m$ matrix $X, x_{ij} \in \{0,1\}$
 - recommending items for each user that are most likely to be consumed by him

2. Preliminaries

Exposure-based matrix factorization (EXMF)

- Problem definition
 - user set U (n users) ,
 - item set I (m items),
 - the implicit feedback data $n * m$ matrix $X, x_{ij} \in \{0,1\}$
 - recommending items for each user that are most likely to be consumed by him

Note that unobserved feedback data $\{x_{ij} \in \mathbf{X} : x_{ij} = 0\}$ contain the real negative data (dislike) and the missing values (unknown). EXMF introduces a Bernoulli variable a_{ij} to model users' exposure: $a_{ij} = 1$ denotes that the user i knows the item j and $a_{ij} = 0$ denotes not. Then, EXMF models user's consumption x_{ij} based on a_{ij} as follow:

$$a_{ij} \sim \text{Bernoulli}(\eta_{ij}) \quad (1)$$

$$(x_{ij}|a_{ij} = 1) \sim N(\mathbf{u}_i^\top \mathbf{v}_j, \lambda_x) \quad (2)$$

$$(x_{ij}|a_{ij} = 0) \sim \delta_0 \approx N(\varepsilon, \lambda_x) \quad (3)$$

where δ_0 denotes $p(x_{ij} = 0|a_{ij} = 0) = 1$; η_{ij} is the prior probability of exposure. Here we relax function δ_0 as $N(\varepsilon, \lambda_x)$ to make model more robust, where ε is a small constant (e.g. $\varepsilon=1e-5$). When $a_{ij} = 0$, we have $x_{ij} \approx 0$,

2. Preliminaries

Analyses of EXMF from variational perspective

- The marginal likelihood of EXMF
- Evidence lower bound (EBLO)

$$\log p(\mathbf{X}) = \sum_{i \in U, j \in I} \log p(x_{ij}).$$

$$\begin{aligned} \log p(x_{ij}) &= E_q[\log p(x_{ij}, a_{ij}) - \log q(a_{ij}|x_{ij})] \\ &\quad + E_q[\log p(a_{ij}|x_{ij}) - \log q(a_{ij}|x_{ij})] \\ &= L(u, v, q; x_{ij}) + D_{KL}(q(a_{ij}|x_{ij}) || p(a_{ij}|x_{ij})) \end{aligned}$$

$q(a_{ij}|x_{ij})$: an approximated variational posterior of a_{ij}

$$q(a_{ij}|x_{ij}) = \text{Bernoulli}(\gamma_{ij}).$$

$$p(x_{ij}, a_{ij}) = p(x_{ij}|a_{ij})p(a_{ij})$$

$$(x_{ij}|a_{ij} = 1) \sim N(\mathbf{u}_i^T \mathbf{v}_j, \lambda_x)$$

$$(x_{ij}|a_{ij} = 0) \sim \delta_0 \approx N(\varepsilon, \lambda_x)$$

$$\begin{aligned} \min J(u, v, \gamma; \mathbf{X}) &= -\frac{2}{\lambda_x} L(u, v, q; x_{ij}) \\ &= \sum_{i \in U, j \in I} \gamma_{ij} (\mathbf{u}_i^T \mathbf{v}_j - x_{ij})^2 + \sum_{i \in U, j \in I} (1 - \gamma_{ij}) (\varepsilon - x_{ij})^2 \\ &\quad - \frac{2}{\lambda_x} \sum_{i \in U, j \in I} D_{KL}(q(a_{ij}|x_{ij}) || p(a_{ij})) \end{aligned}$$

2. Preliminaries

Analyses of EXMF from variational perspective

- The marginal likelihood of EXMF
- Evidence lower bound (EBLO)

$$\log p(\mathbf{X}) = \sum_{i \in U, j \in I} \log p(x_{ij}).$$

$$\begin{aligned} \log p(x_{ij}) &= E_q[\log p(x_{ij}, a_{ij}) - \log q(a_{ij}|x_{ij})] \\ &\quad + E_q[\log p(a_{ij}|x_{ij}) - \log q(a_{ij}|x_{ij})] \\ &= L(u, v, q; x_{ij}) + D_{KL}(q(a_{ij}|x_{ij}) || p(a_{ij}|x_{ij})) \end{aligned}$$

$q(a_{ij}|x_{ij})$: an approximated variational posterior of a_{ij}

γ_{ij} : inferred confidence factors
independent, $n * m$

Inefficiency problem: overfitting,
time-consuming



$$q(a_{ij}|x_{ij}) = \text{Bernoulli}(\gamma_{ij}).$$

$$p(x_{ij}, a_{ij}) = p(x_{ij}|a_{ij})p(a_{ij})$$

$$(x_{ij}|a_{ij} = 1) \sim N(\mathbf{u}_i^T \mathbf{v}_j, \lambda_x)$$

$$(x_{ij}|a_{ij} = 0) \sim \delta_0 \approx N(\varepsilon, \lambda_x)$$

$$\min J(u, v, \gamma; \mathbf{X}) = -\frac{2}{\lambda_x} L(u, v, q; x_{ij})$$

$$= \sum_{i \in U, j \in I} \gamma_{ij} (\mathbf{u}_i^T \mathbf{v}_j - x_{ij})^2 + \sum_{i \in U, j \in I} (1 - \gamma_{ij}) (\varepsilon - x_{ij})^2$$

$$- \frac{2}{\lambda_x} \sum_{i \in U, j \in I} D_{KL}(q(a_{ij}|x_{ij}) || p(a_{ij}))$$

3. Fast adaptively weighted matrix factorization

- Inference function

$$\gamma_{ij} = g_{\Phi}(i, j, \mathbf{X})$$

$$g_{\Phi}(i, j, \mathbf{X}) = \theta_i^T \sigma \left(w_j \sum_{k \in U} \theta_k \alpha_k x_{kj} + b_j \right)$$



map cumulated consumptions
into the exposure.

θ_i : the membership that allocates each user i to a fixed D
number of communities $\theta_i \succeq 0, |\theta_i|_1 = 1$

α_k : the heterogenous roles of users.

- The inferred parameters

$$\Phi \equiv \{\theta, \alpha, w, b\}$$

- $O(nm) \rightarrow O(D(n + m))$

3. Fast adaptively weighted matrix factorization

- neural network perspective
 - tensor Z :
 - the influence of user's consumptions along different dimensions (communities)
 - a specific striped CNN, a linear layer

$$J(u, v, \gamma; \mathbf{X}) = -\frac{2}{\lambda_x} L(u, v, q; x_{ij})$$

$$= \sum_{i \in U, j \in I} \gamma_{ij} (u_i^T v_j - x_{ij})^2 + \sum_{i \in U, j \in I} (1 - \gamma_{ij}) (\varepsilon - x_{ij})^2$$

$$g_{\Phi}(i, j, \mathbf{X}) = \theta_i^T \sigma \left(w_j \sum_{k \in U} \theta_k \alpha_k x_{kj} + b_j \right)$$

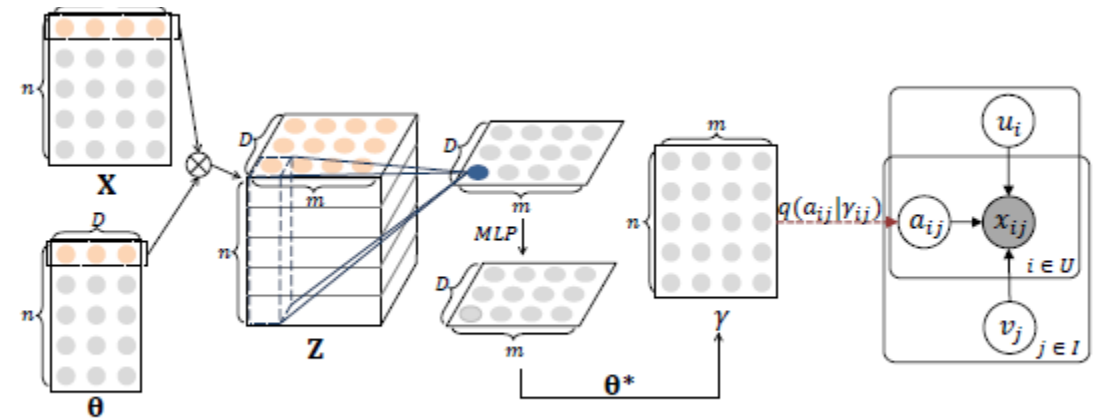


Figure 1: Illustration of the proposed FAWMF

- the exposure based MF : predict users' consumption

3. Fast adaptively weighted matrix factorization

- How does FAWNMF mitigate overfitting?

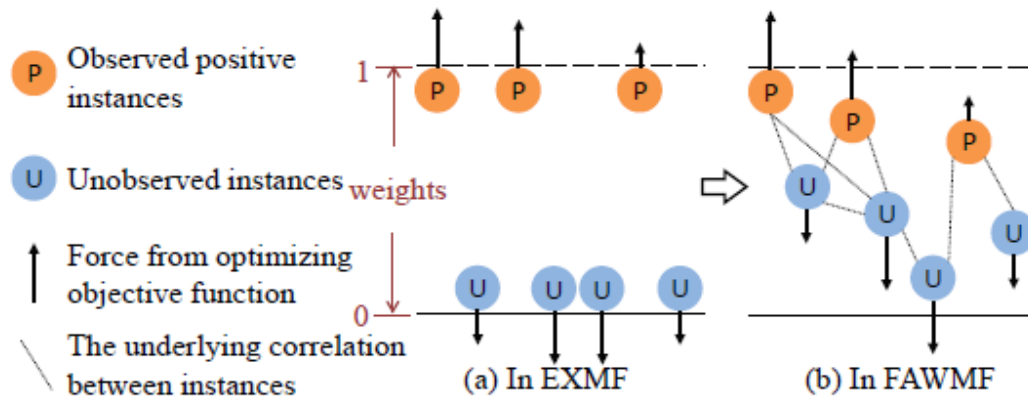


Figure 2: Illustration of how FAWMF mitigates over-fitting

4. Fast learning algorithm fBGD

- SGD VS BGD

$$J(u, v, \gamma; \mathbf{X}) = -\frac{2}{\lambda_x} L(u, v, q; x_{ij})$$

$$= \sum_{i \in U, j \in I} \gamma_{ij} (u_i^T v_j - x_{ij})^2 + \sum_{i \in U, j \in I} (1 - \gamma_{ij}) (\varepsilon - x_{ij})^2$$

$$g_{\Phi}(i, j, \mathbf{X}) = \theta_i^T \sigma(w_j \sum_{k \in U} \theta_k \alpha_k x_{kj} + b_j)$$

$$\frac{\partial J}{\partial \alpha_k} = \sum_{j \in I} \frac{\partial J}{\partial q_j} \cdot \frac{\partial q_j}{\partial \alpha_k} \quad (7)$$

$$\frac{\partial J}{\partial q_j} = \sum_{i \in U} (u_i^T v_j u_i^T v_j - 2x_{ij}(u_i^T v_j - \varepsilon) - \varepsilon^2) \theta_i \quad (8)$$

$$\frac{\partial q_j}{\partial \alpha_k} = q_j \cdot (1 - q_j) \cdot \theta_k w_j x_{kj} \quad (9)$$

- 采样
- 负例采样策略在优化的时候会用到SGD，收敛速度慢，不robust
- samples the uninformative data, low confidence and make limited contributions on gradient update

Update α : $O(nmKD)$



4. Fast learning algorithm fBGD

- SGD VS BGD

$$J(u, v, \gamma; \mathbf{X}) = -\frac{2}{\lambda_x} L(u, v, q; x_{ij})$$

$$= \sum_{i \in U, j \in I} \gamma_{ij} (\mathbf{u}_i^T \mathbf{v}_j - x_{ij})^2 + \sum_{i \in U, j \in I} (1 - \gamma_{ij}) (\varepsilon - x_{ij})^2$$

$$g_{\Phi}(i, j, \mathbf{X}) = \theta_i^T \sigma(w_j \sum_{k \in U} \theta_k \alpha_k x_{kj} + b_j)$$

$$\frac{\partial J}{\partial \alpha_k} = \sum_{j \in I} \frac{\partial J}{\partial q_j} \cdot \frac{\partial q_j}{\partial \alpha_k} \quad (7)$$

$$\frac{\partial J}{\partial q_j} = \sum_{i \in U} (\mathbf{u}_i^T \mathbf{v}_j \mathbf{u}_i^T \mathbf{v}_j - 2x_{ij}(\mathbf{u}_i^T \mathbf{v}_j - \varepsilon) - \varepsilon^2) \theta_i \quad (8)$$

$$\frac{\partial q_j}{\partial \alpha_k} = q_j \cdot (1 - q_j) \cdot \theta_k w_j x_{kj} \quad (9)$$

Update α : $O(nmKD)$

By isolating item-independent terms

$$\frac{\partial J}{\partial q_j} = \sum_{k=1}^K \sum_{l=1}^K v_{jk} v_{jl} \sum_{i \in U} u_{ik} u_{il} \theta_i - \varepsilon^2 \sum_{i \in U} \theta_i$$

$$- 2 \sum_{i \in U} x_{ij} (\mathbf{u}_i^T \mathbf{v}_j - \varepsilon) \theta_i \quad O(K^2 Dm) \quad (10)$$

$$\mathbf{M}_{kl*}^{(q)} = \sum_{i \in U} u_{ik} u_{il} \theta_i \text{ for each } 1 \leq k \leq K, 1 \leq l \leq K$$

$\mathbf{M}^{(q)}$ is a $K \times K \times D$.

$$\mathbf{S}^{(q)} = \sum_{i \in U} \theta_i.$$

$\mathbf{S}^{(q)}$ is a D-dimensional vector.

$O(K^2 Dn)$

$$O(nmKD) \rightarrow O(K^2 D(n + m) + |X^+|(K + D))$$

5. Experiments and analyses

Dataset

Table 1: Statistics of three datasets.

Datasets	#Users	#Items	#Oberseved positive feedback
Movielens	6,040	3,952	1,000,209
Amazon	10,619	37,762	256,287
Douban	123,480	20,029	16,624,937

Compared methods

Table 2: characteristics of the compared methods.

Methods	Adaptive weights?	Without sampling?	Complexity
WMF(ALS)	\	✓	$O((n+m)K^3 + \mathbf{X}^+ K^2)$
eALS	\	✓	$O((n+m)K^2 + \mathbf{X}^+ K)$
BPR	\	\	$O((n+m+ \mathbf{X}^+)K)$
CDAE	\	\	$O((n+m+ \mathbf{X}^+)K)$
EXMF	✓	✓	$O(nmK^3)$
FAWMF	✓	✓	$O((n+m)K^2D + \mathbf{X}^+ (K+D))$



5. Experiments and analyses

- 与传统的baseline比较性能：预测的 accuracy, running time
- 与基于sample的方法比较：accuracy, running time
- 超参数：accuracy受到的影响
- 对community的可视化：是否符合 motivation里面提到的现象

Dataset

Table 1: Statistics of three datasets.

Datasets	#Users	#Items	#Oberseved positive feedback
Movielens	6,040	3,952	1,000,209
Amazon	10,619	37,762	256,287
Douban	123,480	20,029	16,624,937

Compared methods

Table 2: characteristics of the compared methods.

Methods	Adaptive weights?	Without sampling?	Complexity
WMF(ALS)	\	✓	$O((n+m)K^3 + \mathbf{X}^+ K^2)$
eALS	\	✓	$O((n+m)K^2 + \mathbf{X}^+ K)$
BPR	\	\	$O((n+m+ \mathbf{X}^+)K)$
CDAE	\	\	$O((n+m+ \mathbf{X}^+)K)$
EXMF	✓	✓	$O(nmK^3)$
FAWMF	✓	✓	$O((n+m)K^2D + \mathbf{X}^+ (K+D))$

5. Experiments and analyses

- Evaluation Metrics
- Performance comparison
- Running time comparisons

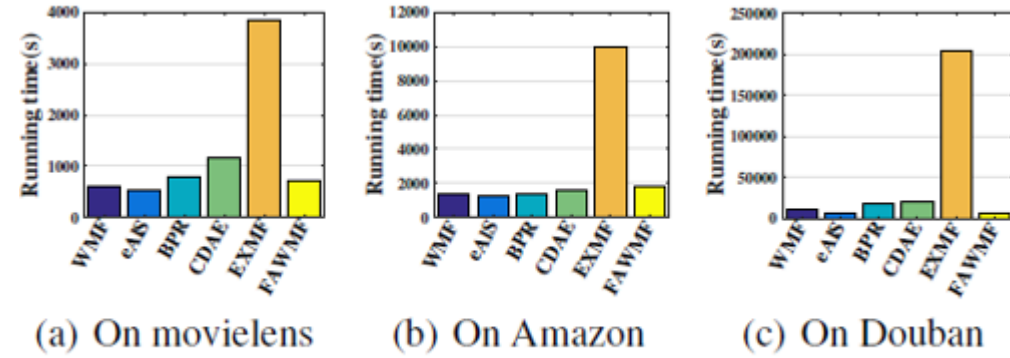


Figure 3: Running time comparisons.

Table 3: The performance metrics of the compared methods. The boldface font denotes the winner in that column. The row ‘Impv’ indicates the relative performance gain of our FAWMF compared to the best results among baselines. ‘†’ indicates that the improvement is significant with t-test at $p < 0.05$.

Methods	Movielens				Amazon				Douban			
	Pre@5	Rec@5	NDCG@5	MRR	Pre@5	Rec@5	NDCG@5	MRR	Pre@5	Rec@5	NDCG@5	MRR
Item-pop	0.2092	0.0400	0.2201	0.8958	0.0027	0.0048	0.0055	0.0191	0.1409	0.0332	0.1582	0.6308
WMF(ALS)	0.3841	0.0924	0.4059	1.5751	0.0789	0.0406	0.0858	0.3269	0.2400	0.0656	0.2598	1.0113
eALS	0.3955	0.0917	0.4175	1.5998	0.0984	0.0348	0.1051	0.3951	0.2329	0.0646	0.2520	0.9880
BPR	0.3613	0.0798	0.3794	1.5023	0.0988	0.0469	0.1060	0.3969	0.2371	0.0582	0.2570	1.0093
CDAE	0.3786	0.0860	0.3950	1.5454	0.0948	0.0472	0.0947	0.3994	0.2377	0.0589	0.2573	1.0162
EXMF	0.3871	0.0936	0.4071	1.5720	0.0847	0.0418	0.0928	0.3683	0.2353	0.0666	0.2588	1.0016
FAWMF	0.4054	0.0949	0.4275	1.6279	0.1129	0.0441	0.1285	0.4470	0.2661	0.0680	0.2915	1.0984
Impv%	2.49%†	1.41%†	2.40%†	1.76%†	14.34%†	-6.42%	21.18%†	11.93%†	10.88%†	2.11%†	12.19%†	8.09%†

5. Experiments and analyses

- Effect of batch-based learning algorithm
- Effect of the parameter K
- Case study

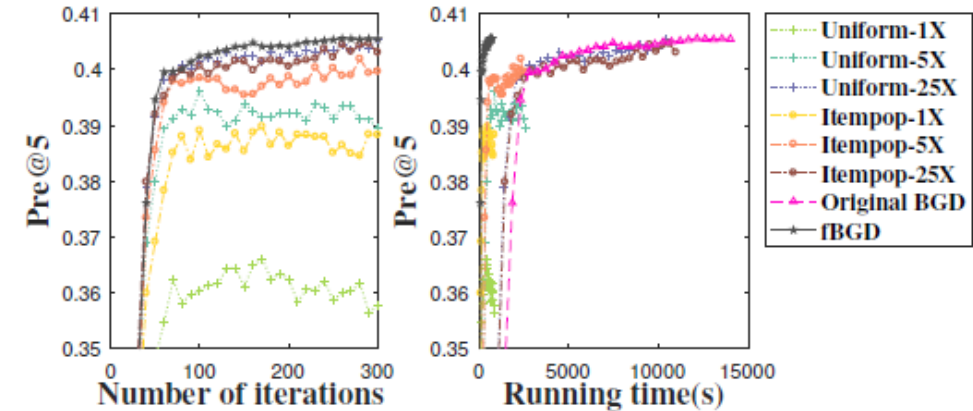
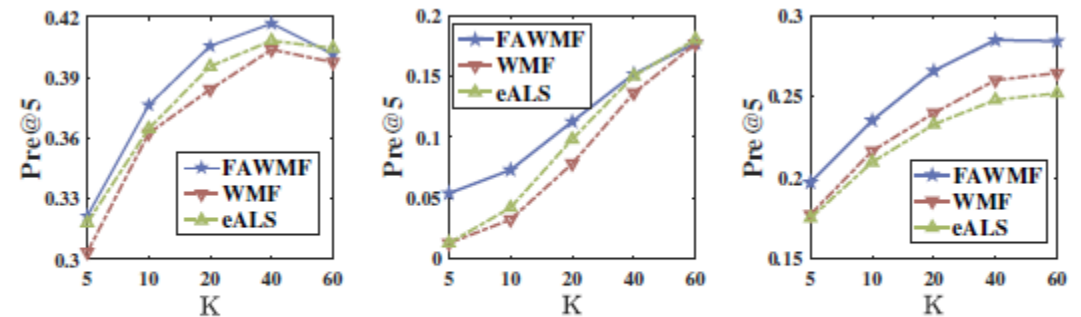


Figure 4: Recommendation accuracy for different learning strategy versus the number of iterations and running time.

Community 1		Community 2	
Movie Name	Genres	Movie Name	Genres
1. Gattaca	Drama, Sci-Fi, Thriller	1. My Favorite Year	Comedy
2. Thinner	Horror, Thriller	2. Annie Hall	Comedy, Romance
3. Bride of Frankenstein	Horror	3. Citizen Kane	Drama
4. The New Age	Drama	4. Unforgiven	Western
5. Halloween 4	Horror	5. Father of the Bride	Comedy
Ratio of male/female	Average age	Ratio of male/female	Average age
82.3%/17.7%	33.8	66.70%/33.3%	40.5

Figure 6: Case study of two communities: the top rows show top-5 items that have highest exposure to the users in the two communities; the bottom rows show the ratio of male/female and the average age of the users in each communities. Note that these side information is not used in model training.



(a) On movielens (b) On Amazon (c) On Douban

Figure 5: Impact of the parameter K .



Thanks for your listening!

Q & A