User-Dependent Learning to Debias for Recommendation

Fangyuan Luo, Jun Wu
Beijing Jiaotong University, Beijing 100044, China

Motivation

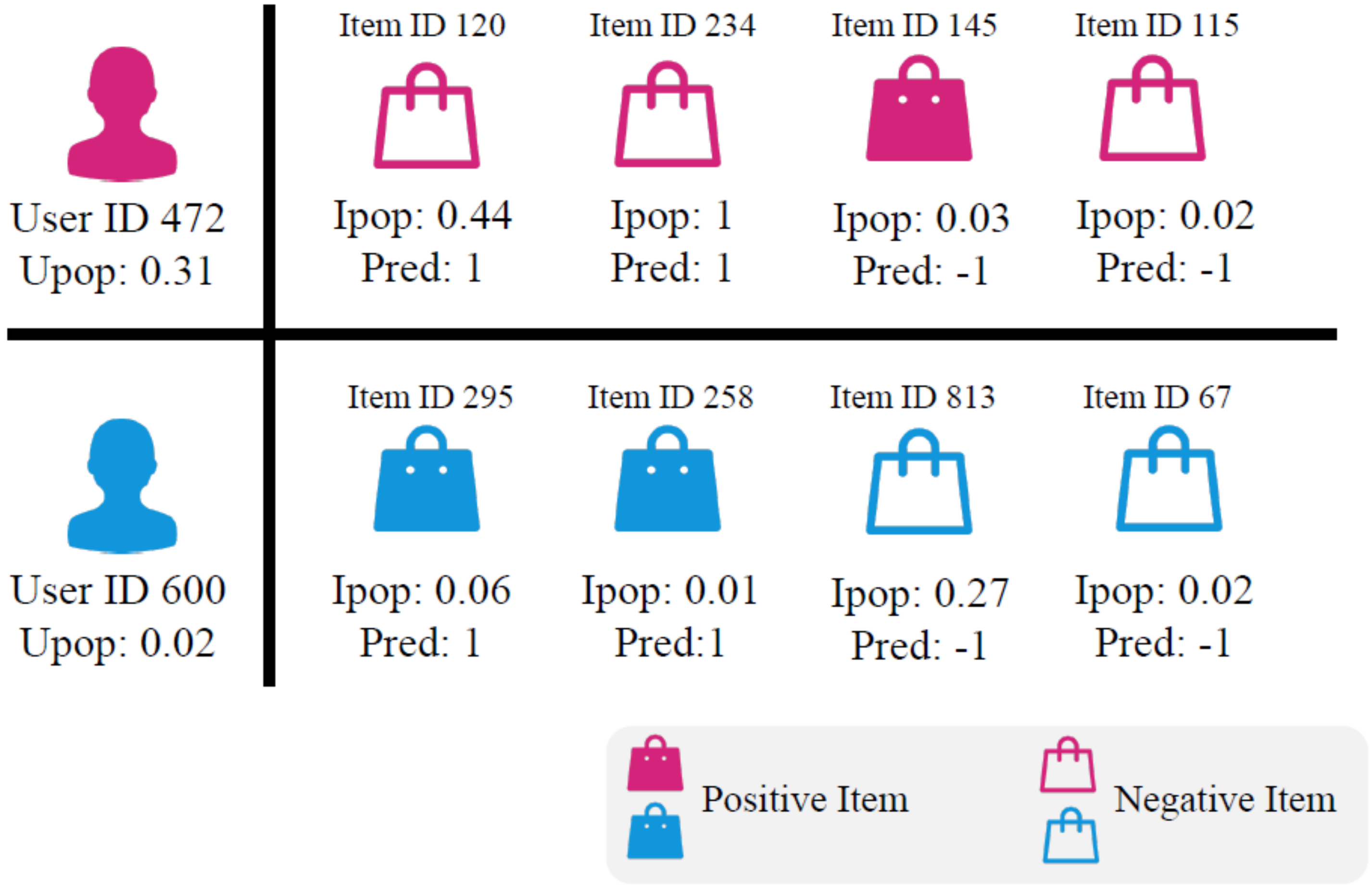

Figure 1: A case study of MF on Yahoo! R3 dataset

- Users with high Upop tend to give higher prediction scores to items with higher Ipop.
- Users with low Upop is less influenced by Ipop.

Motivation

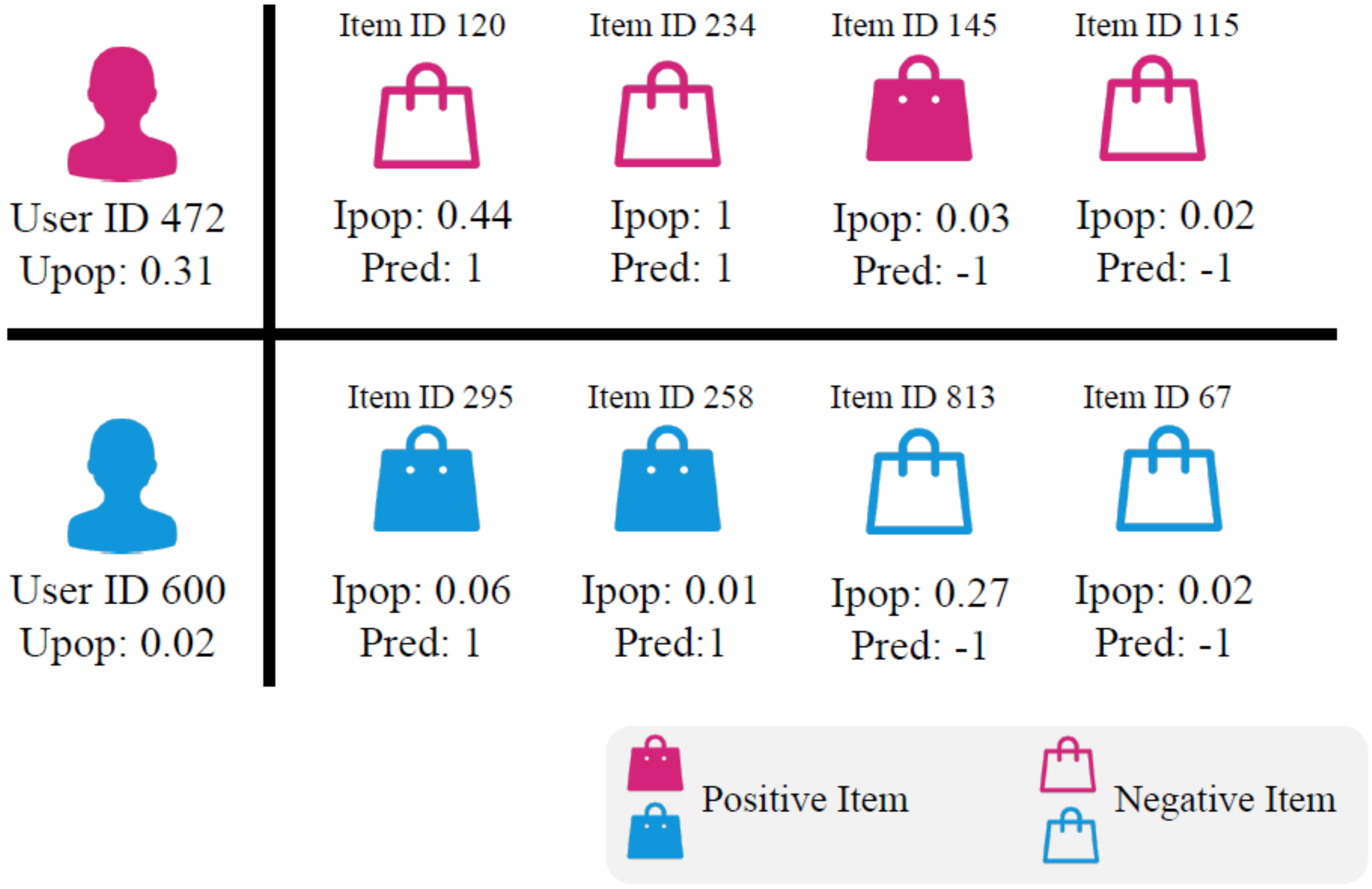

Figure 1: A case study of MF on Yahoo! R3 dataset

- Users with high Upop tend to give higher prediction scores to items with higher Ipop.
- Users with low Upop is less influenced by Ipop.

It is sub-optimal to treat all users equally and necessary to take into account users' popularity sensitivity, which has not been studied in unbiased recommender learning.

Related Work

Existing methods for addressing popularity bias mainly fall into four lines.

- Inverse Propensity Scoring (IPS) [1-5]
- Causal Intervention [6-8]
- Regularization constraints [9-10]
- Reranking [11-12]

[1] Alois Gruson, Praveen Chandar, Christophe Charbuillet, James McInerney, Samantha Hansen, Damien Tardieu, and Ben Carterette. 2019. Offline Evaluation to Make Decisions About PlaylistRecommendation Algorithms. In WSDM. 420–428.
[2] Jin Huang, Harrie Oosterhuis, and Maarten de Rijke. 2022. It Is Different When Items Are Older: Debiasing Recommendations When Selection Bias and User Preferences Are Dynamic. In WSDM. 381–389.
[3] Thorsten Joachims, Adith Swaminathan, and Tobias Schnabel. 2017. Unbiased Learning-to-Rank with Biased Feedback. In WSDM. 781–789.
[4] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. In ICML, Vol. 48. 1670–1679.
[5] Longqi Yang, Yin Cui, Yuan Xuan, Chenyang Wang, Serge J. Belongie, and Deborah Estrin. 2018. Unbiased offline recommender evaluation for missing-not-atrandom implicit feedback. In RecSys. 279–287.
[6] Wenjie Wang, Fuli Feng, Xiangnan He, Xiang Wang, and Tat-Seng Chua. 2021. Deconfounded Recommendation for Alleviating Bias Amplification. In KDD. 1717–1725.
[7] Tianxin Wei, Fuli Feng, Jiawei Chen, Ziwei Wu, Jinfeng Yi, and Xiangnan He. 2021. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System. In KDD. 1791–1800.
[8] Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling, and Yongdong Zhang. 2021. Causal Intervention for Leveraging Popularity Bias in Recommendation. In SIGIR. 11–20.
[9] Wondo Rhee, Sung Min Cho, and Bongwon Suh. 2022. Countering Popularity Bias by Regularizing Score Differences. In RecSys. 145–155.
[10] Ziwei Zhu, Yun He, Xing Zhao, Yin Zhang, Jianling Wang, and James Caverlee. 2021. Popularity-Opportunity Bias in Collaborative Filtering. In WSDM. 85–93.
[11] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. 2019. Managing Popularity Bias in Recommender Systems with Personalized Re-Ranking. In FLAIRS. 413–418.
[12] Mi Zhang and Neil Hurley. 2008. Avoiding monotony: improving the diversity of recommendation lists. In RecSys. 123–130.

Method

- Propensity Estimation.

item popularity: $Ipop_i = \frac{\sum_{u \in \mathcal{U}} \mathbb{I}(r_{ui} = 1)}{max\_Ipop}$,

users' popularity sensitivity: $\rho_{ui} = Upop_u = \frac{\sum_{i \in \Omega_u} Ipop_i}{|\Omega_u|}$,

- Unbiased Learning Objective.

$$\mathcal{L}_{UDIPS} = \frac{1}{|\mathcal{U}||\mathcal{I}|} \sum_{(u,i):O_{ui}=1} \left(\frac{\alpha_u}{\rho_{ui}} \cdot \delta(\hat{r}_{ui}, r_{ui}) + (1 - \alpha_u) \cdot \delta(\hat{r}_{ui}, r_{ui}) \right),$$

where $\alpha_u \in \{0, 1\}$ is a binary variable which is used to determine whether a user is sensitive to item popularity, and ρ_{ui} is computed by Eq.(6). Empirically, we set $\alpha_u = 1$ when users' popularity is larger than a threshold θ; otherwise, $\alpha_u = 0$.

Method

➢ Propensity Estimation.

item popularity: $Ipop_i = \frac{\sum_{u \in \mathcal{U}} \mathbb{I}(r_{ui} = 1)}{max\_Ipop},$

users' popularity sensitivity: $\rho_{ui} = Upop_u = \frac{\sum_{i \in \Omega_u} Ipop_i}{|\Omega_u|},$

➢ Unbiased Learning Objective.

$$\mathcal{L}_{UDIPS} = \frac{1}{|\mathcal{U}||\mathcal{I}|} \sum_{(u,i):O_{ui}=1} \left(\frac{\alpha_u}{\rho_{ui}} \cdot \delta(\hat{r}_{ui}, r_{ui}) + (1 - \alpha_u) \cdot \delta(\hat{r}_{ui}, r_{ui}) \right)$$

where $\alpha_u \in \{0, 1\}$ is a binary variable which is used to determine whether a user is sensitive to item popularity, and ρ_{ui} is computed by Eq.(6). Empirically, we set $\alpha_u = 1$ when users' popularity is larger than a threshold θ; otherwise, $\alpha_u = 0$.

➢ Unbiasedness Analysis.

$$\mathbb{E}\left[\mathcal{L}_{UDIPS}(\hat{\mathrm{R}}|\alpha_u = 1)\right] = \mathbb{E}\left[\frac{1}{|\mathcal{U}||\mathcal{I}|} \sum_{(u,i):O_{ui}=1} \alpha_u \cdot \frac{1}{\rho_{ui}} \cdot \delta(\hat{r}_{ui}, r_{ui})\right]$$
$$= \frac{1}{|\mathcal{U}||\mathcal{I}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \frac{\mathbb{E}[O_{ui}]}{\rho_{ui}} \cdot \delta(\hat{r}_{ui}, r_{ui})$$
$$= \frac{1}{|\mathcal{U}||\mathcal{I}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \frac{\rho_{ui}}{\rho_{ui}} \cdot \delta(\hat{r}_{ui}, r_{ui}) = \frac{1}{|\mathcal{U}||\mathcal{I}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \delta(\hat{r}_{ui}, r_{ui}).$$

$$\mathbb{E}\left[\mathcal{L}_{UDIPS}(\hat{\mathrm{R}}|\alpha_u = 0)\right] = \mathbb{E}\left[\frac{1}{|\mathcal{U}||\mathcal{I}|} \sum_{(u,i):O_{ui}=1} (1 - \alpha_u) \cdot \delta(\hat{r}_{ui}, r_{ui})\right]$$
$$= \frac{1}{|\mathcal{U}||\mathcal{I}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \mathbb{E}(O_{ui}) \cdot \delta(\hat{r}_{ui}, r_{ui}) = \frac{1}{|\mathcal{U}||\mathcal{I}|} \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}} \delta(\hat{r}_{ui}, r_{ui}).$$

Experiments

- Datasets

Table 1: Statistics of the datasets, NB-Tr and UB-Tr are short for normal biased training data and unbiased training data, respectively. NB-Te and UB-Te are short for normal biased testing data and unbiased testing data, respectively.

Dataset	#Users	#Items	#NB-Tr	#UB-Tr	#Val	#NB-Te	#UB-Te
Yahoo! R3	15.4k	1.0k	249k	5.4k	33.8k	31.2k	48.6k
Coat	290	300	5.6k	464	928	696	4.1k

- Baselines

MF, MF-IPS [ICML'16], InterD [SIGIR'22], KD_Label [SIGIR'20], DR [ICML'19]

- Metrics

NDCG, Precision, Recall, AUC

Experiments

Model		Yahoo! R3				Coat			
		NDCG	Presicion	Recall	AUC	NDCG	Presicion	Recall	AUC
Unbiased Test (UB-Te)	MF	0.547	0.256	0.730	0.649	0.492	0.327	0.537	0.667
	MF-IPS	0.548	0.257	0.730	0.649	0.494	0.329	0.539	0.665
	MF-UDIPS	**0.579**	**0.263**	**0.753**	**0.677**	**0.499**	**0.332**	**0.547**	**0.673**
	InterD	0.669	0.288	0.822	0.753	0.519	0.337	0.558	0.682
	InterD-UDIPS	**0.676**	**0.291**	**0.833**	**0.762**	**0.526**	**0.342**	**0.570**	**0.692**
	KD_Label	0.575	0.259	0.751	0.674	0.502	0.325	0.540	0.679
	KDLabel-UDIPS	**0.585**	**0.263**	**0.759**	**0.681**	**0.506**	**0.328**	**0.555**	**0.686**
	DR	0.548	0.256	0.731	0.650	0.493	0.328	0.540	0.667
	DR-UDIPS	**0.552**	**0.261**	**0.749**	**0.660**	**0.504**	**0.334**	**0.563**	**0.670**
Normal Biased Test (NB-Te)	MF	0.825	0.313	0.970	0.652	0.810	0.267	0.995	0.667
	MF-IPS	0.815	0.311	0.966	0.626	0.809	0.265	0.986	0.636
	MF-UDIPS	**0.830**	**0.314**	**0.971**	**0.661**	**0.820**	**0.270**	**0.995**	**0.660**
	InterD	0.837	0.316	0.973	0.673	0.830	0.271	0.994	0.669
	InterD-UDIPS	**0.841**	**0.319**	**0.976**	**0.683**	**0.832**	**0.274**	**0.995**	**0.671**
	KD_Label	0.814	0.313	0.968	0.628	0.814	0.263	0.991	0.620
	KDLabel-UDIPS	**0.827**	**0.316**	**0.969**	**0.655**	**0.820**	**0.268**	**0.995**	**0.640**
	DR	0.791	0.308	0.957	0.571	0.812	0.259	0.986	0.638
	DR-UDIPS	**0.825**	**0.317**	**0.970**	**0.651**	**0.823**	**0.274**	**0.991**	**0.656**

- UDIPS-based methods consistently outperforms existing models on UB-Te and NB-Te across two datasets.
- Compared with MF, all debiased model perform better on UB-Te but show inferior performance on the NB-Te except UDIPS-based methods and InterD, which illustrates the most debiased methods improve the debiased performance with the sacrifice of biased performance.

Experiments

➢ Comparison in terms of popular sensitive/insensitive users.

		NDCG	Recall	Precision	AUC
MF-IPS	PI users	0.53435	0.69128	0.25156	0.62877
	PS users	0.56170	0.72476	0.24195	0.65501
MF-UDIPS	PI users	0.57058	0.73551	0.26484	0.66592
	Gain(%)	**6.78%**	**6.40%**	**5.28%**	**5.91%**
	PS users	0.60515	0.78583	0.25957	0.70340
	Gain(%)	**7.74%**	**8.43%**	**7.28%**	**7.39%**

- The recommendation performance of PI users and PS users are both boosted.
- The performance gain from PS users is larger than that from PI users. It indicates that our proposed method is more effective in handling under-debiasing of PS users compared with the over-debiasing of PI users.

➢ Impact of Hyper-Parameter.

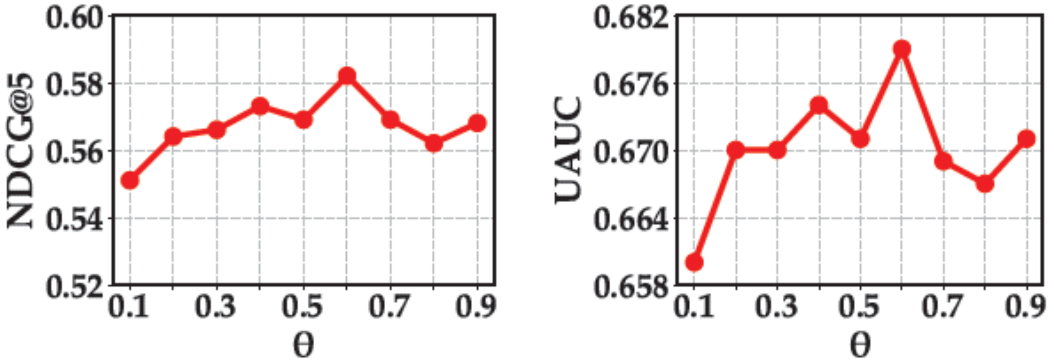

Figure 2: Impact of threshold θ with MF-UDIPS on Yahoo! R3 dataset with the evaluation metric NDCG@5 and UAUC.

Thank you!