中国科学技术大学
University of Science and Technology of China

NUS
National University of Singapore

Bias Issues and Solutions in Recommender System

Jiawei Chen, Xiang Wang, Fuli Feng, Xiangnan He

cjwustc@ustc.edu.cn

slides will be available at: https://github.com/jiawei-chen/RecDebiasing

A literature survey based on this tutorial is available at: https://arxiv.org/pdf/2010.03240.pdf

.

• About Us

Jiawei Chen

Postdoc Researcher

University of Science
and Technology of China

cjwustc@ustc.edu.cn

Xiang Wang

Postdoc Researcher

National University of
Singapore

xiangwang@u.nus.edu

Fuli Feng

Postdoc Researcher

National University of
Singapore

fulifeng93@gmail.com

Xiangnan He

Professor

University of Science
and Technology of China

xiangnanhe@gmail.com

• Information Seeking

Information explosion problem?

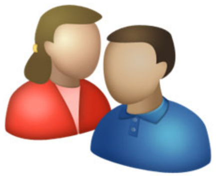
- Information seeking requirements
  - ➢ E-commerce (Taobao/PDD/Amazon)
  - ➢ Social networking (Facebook/Weibo/Wechat)
  - ➢ Content sharing platforms (Tiktok/Kwai/Pinterest)

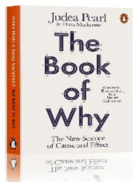
12 million items in Amazon

2.8 billion users in Facebook

720,000 hours videos uploaded per day in Youtube

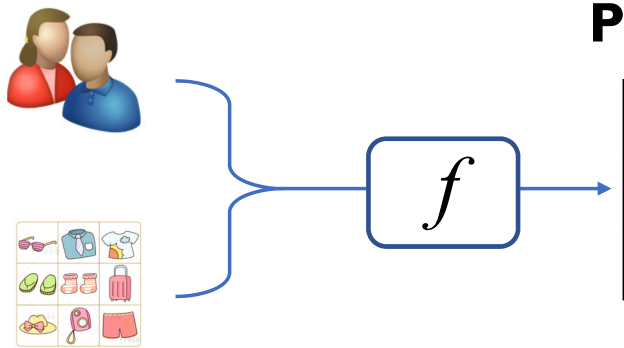
Recommender system has been recognized as a powerful tool to address information overload.





• Classical Problem Setting

- Given:
  - A set of users $U = \{u_1, u_2, \ldots, u_n\}$
  - A set of items $I = \{i_1, i_2, \ldots, i_m\}$
  - Users history feedback on items: $R^o \subseteq \mathbb{R}^{n \times m}$
- To learn a model to predict preference for each user-item pair: $\hat{R} = f(U, I \mid \theta)$
- Minimizing the difference between the prediction and the observed feedback



Predicted Score

3	2	1	4
2	2	3	5
2	4	5	5

$\approx$

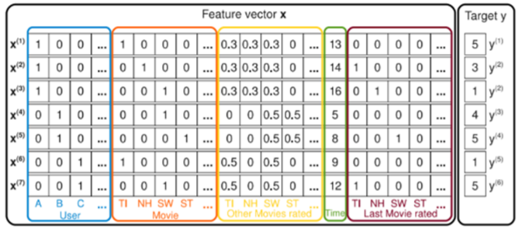
History feedback

3		1	
2	3		5
2		5	

• Mainstream Models

➢ Collaborative filtering

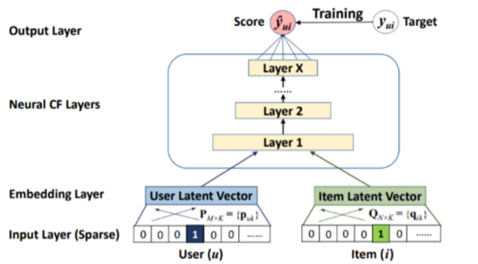
- Matrix factorization & factorization machines



Factorization Machines

➢ Deep learning approaches

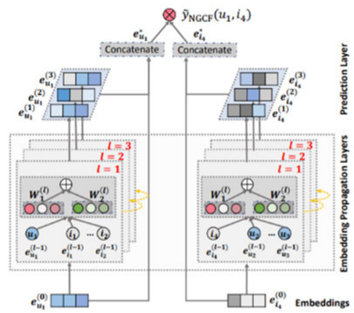
- Neural factorization machines & deep interest networks



Neural Collaborative Filtering

➢ Graph-based approaches

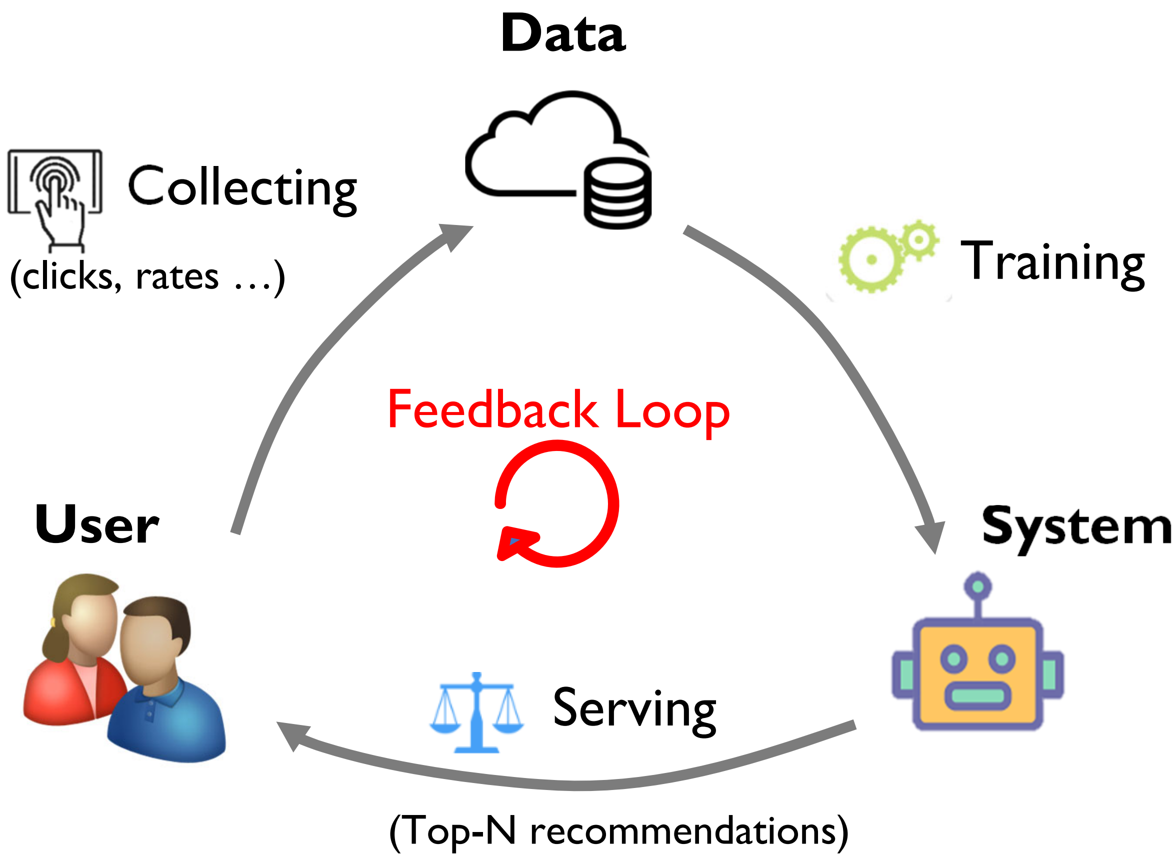
- Leveraging user-item interaction graphs & knowledge graph



Neural Graph Collaborative Filtering

• Ecosystem of Recsys

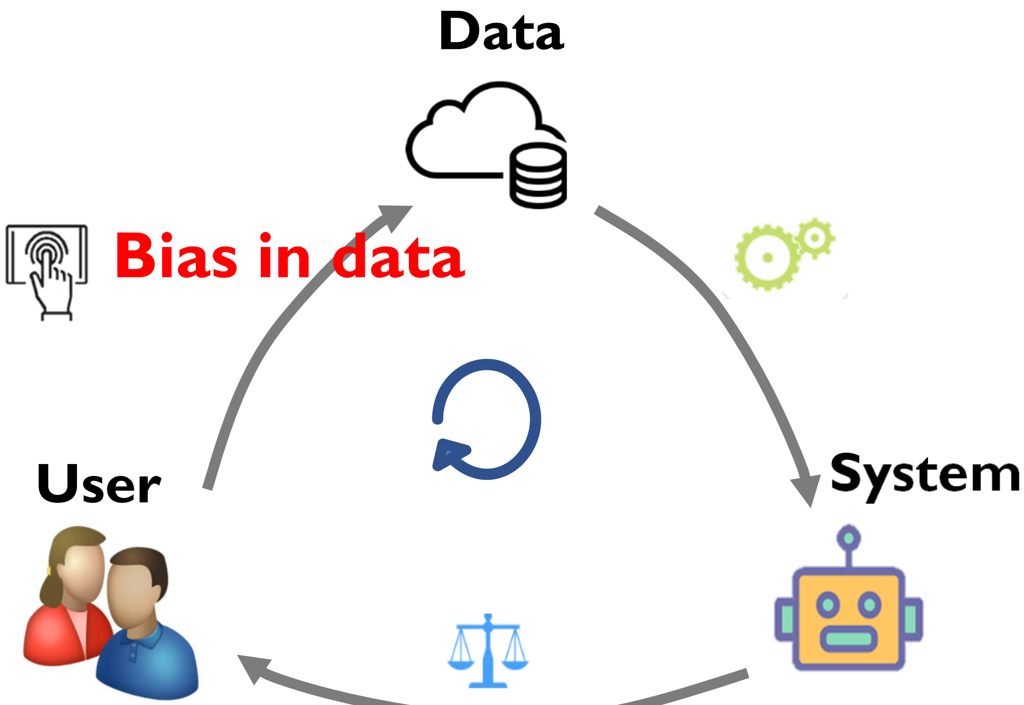
- Workflow of RS
  - **Training**: RS is trained/updated on observed user-item interaction data.
  - **Serving**: RS infers user preference over items and exposes top-n items.
  - **Collecting**: User actions on exposed items are merged into the training data.
- Forming a Feedback Loop



• Where Bias Comes?

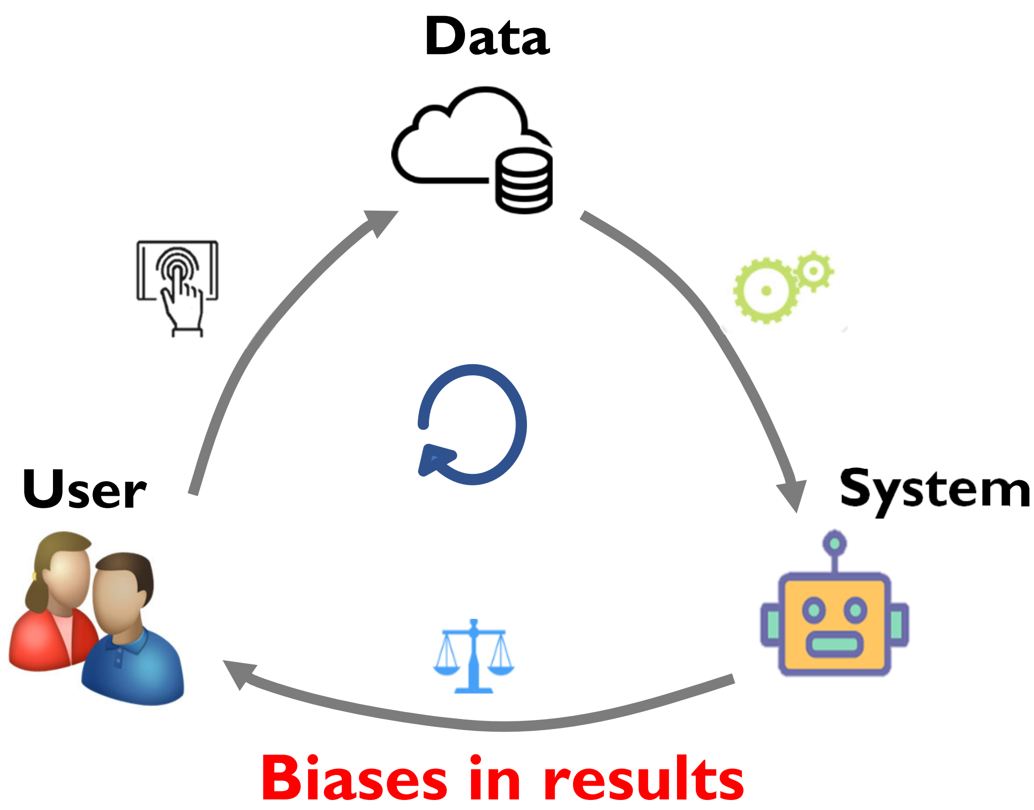
- Bias in data (Collecting):
    - Data is observational rather than experimental (i.e., missing-not-at-random)
    - Affected by many factors:
        - The exposure mechanism
        - Public opinions
        - Display position

      ……
    - The collected data deviates from user true preference.



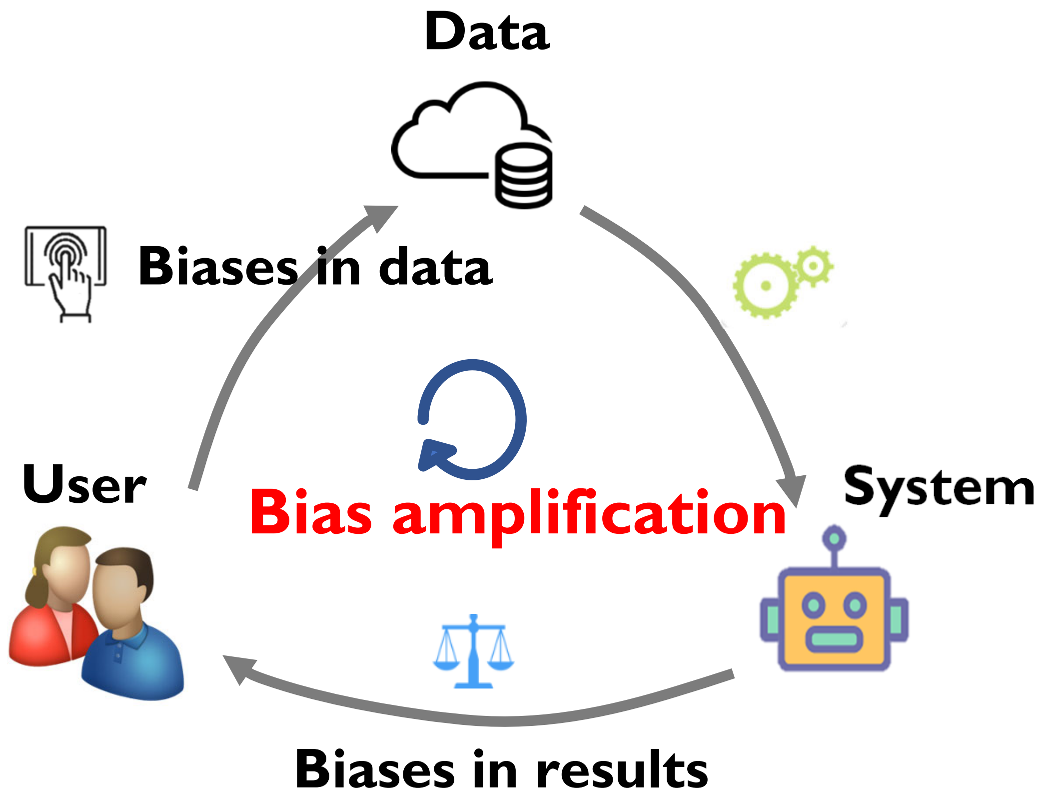
• Where Bias Comes?

- Bias in results (Serving):
  - Unbalanced training data
  - Recommendations are in favor of some item groups
  - E.g., popularity bias, category-aware unfairness
  - Hurting user experience and satisfaction



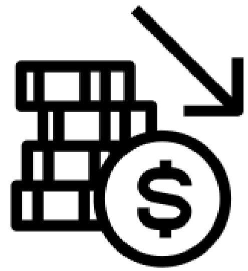
• Matthew Effect: Bias + Loop

- Biases amplification along the loop:
  - Biases would be circled back into the collected data
  - Resulting in "Matthew effect" issue: the rich gets richer
  - Damaging the ecosystem of RS



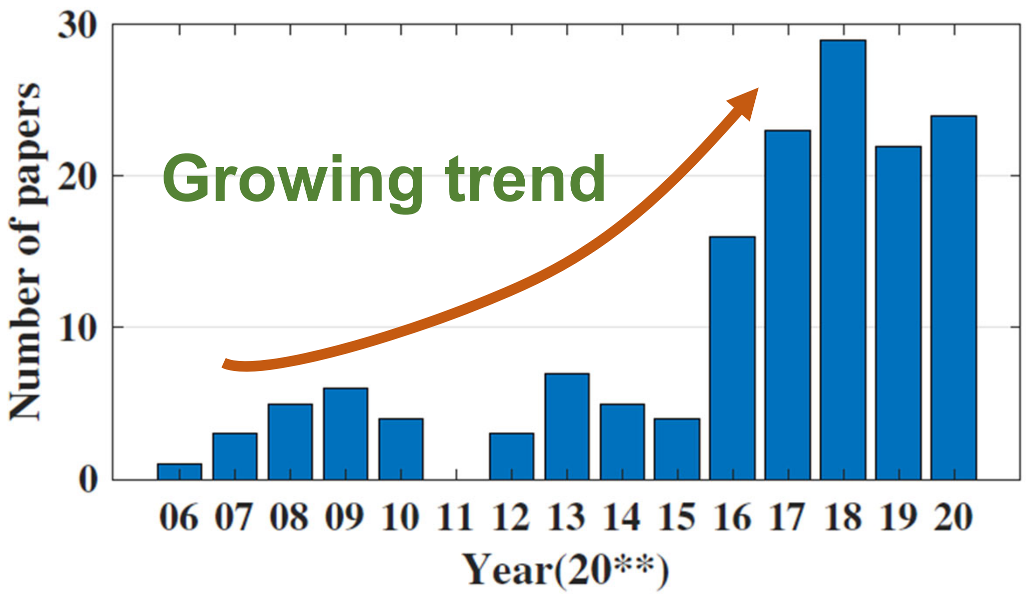
• Bias is Evil

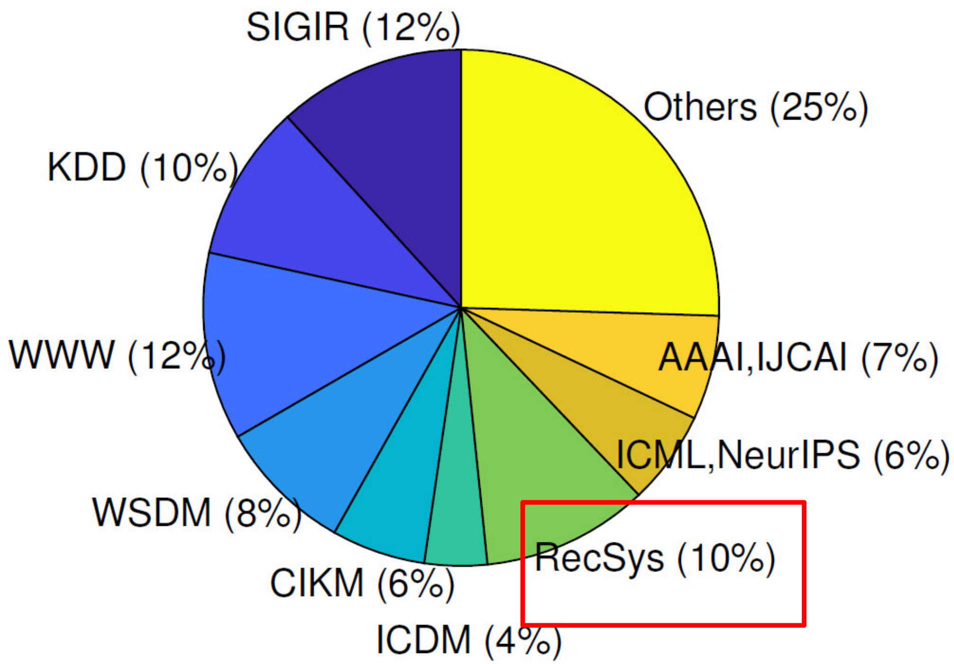
- Economic
  - Bias affects recommendation accuracy
  - Bias hurts user experience, causing the losses of users
  - Unfairness incurs the losses of item providers



- Society
  - Bias can reinforce discrimination of certain user's groups
  - Bias decreases the diversity and intensify the homogenization of users

• Increasing Research in Recsys Bias





Recommendation debiasing becomes a hot topic in top conference

• Also Best Papers and Challenges

sigir21

Best Paper: Computationally Efficient Optimization of Plackett-Luce Ranking Models for Relevance and Fairness by Harrie Oosterhuis

Best Paper Honorable Mention: Causal Intervention for Leveraging Popularity Bias in Recommendation by Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling and Yongdong Zhang

wsdm 2021

WSDM 2021 Best Paper Award Recipient

38: Unifying Online and Counterfactual Learning to Rank

Harrie Oosterhuis (University of Amsterdam), Maarten de Rijke (University of Amsterdam & Ahold Delhaize).

Alibaba Cloud

Tianchi Academic Competitions — Join the Latest Big Data Competitions and Get Exclusive Awards for University Students

KDD Cup 2020 Challenges for Modern E-Commerce Platform: Debiasing

• Tutorial Outline

- ❑Biases in Data (Jiawei Chen, 60 min)
  - ❑Definition of data biases
  - ❑Categories: Selection bias, Conformity bias, Exposure bias and Position bias
  - ❑Recent solutions for data biases
- ❑ Bias Amplification in Loop and its Solutions (Jiawei Chen, 10 min)
- ❑ Biases in Results
  - ❑Popularity bias: definition, characteristic and solutions (Fuli Feng, 40 min)
  - ❑Unfairness: definition, characteristic and solutions (Xiang Wang, 50 min)

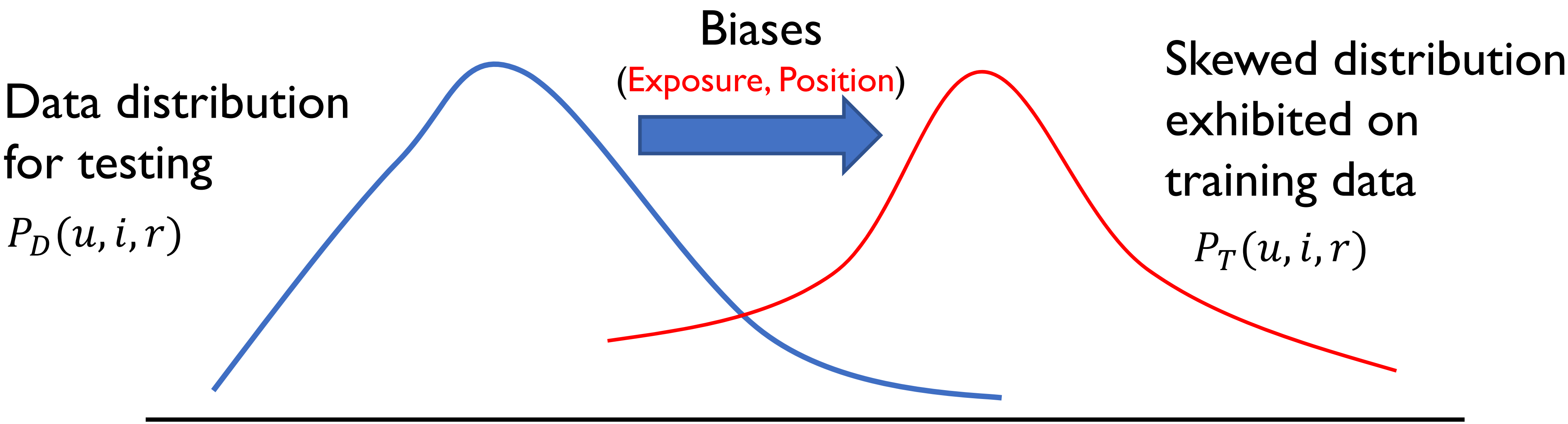
slides will be available at: https://github.com/jiawei-chen/RecDebiasing
A literature survey based on this tutorial is available at: https://arxiv.org/pdf/2010.03240.pdf
.

• What does this section cover?

- What is data bias? The definition of data bias.
- What causes data bias? The taxonomy of data bias.
- How to address data bias? Some typical solutions.
- How does the bias amplify along the loop?

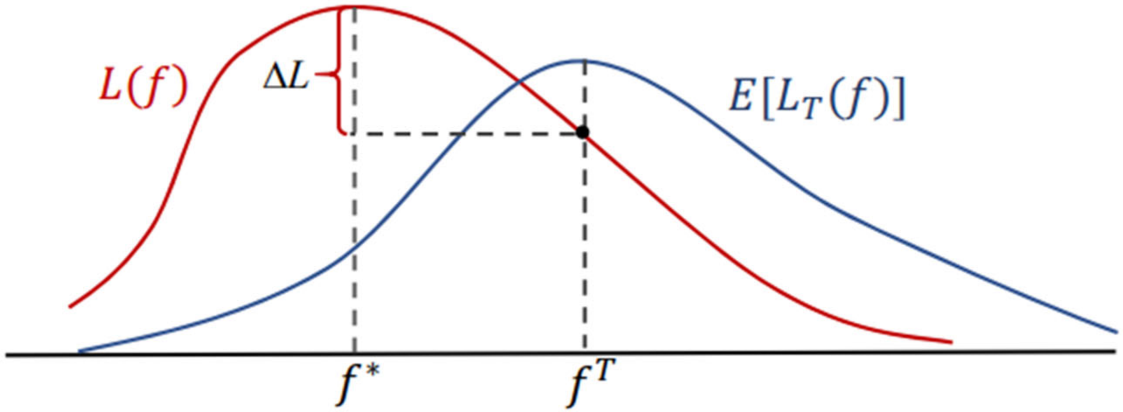
- **What is data bias?**

Data bias: *The distribution for which the training data is collected is different from the ideal data distribution.*



• Impact of Data Bias

- Data bias causes model training towards wrong direction.



Distributional difference between p_T and P_D . → Risk discrepancy between $\hat{L}_T(f)$ and $L(f)$. → Suboptimal results.

$p_T \neq p_D$ $\qquad$ $E_{P_T}[\hat{L}_T(f)] \neq L(f)$ $\qquad$ $f^* \neq f^T$

- True risk. $\quad L(f) = E_{P_D(u,i)P_D(R_{ui}|u,i)}[\delta(f(u,i), R_{ui})]$
- Empirical risk. $\quad \hat{L}_T(f) = \frac{1}{|D_T|} \sum_{(u,i,r_{ui}) \in D_T} \left[\delta\left(f(u,i), r_{ui}\right)\right]$

• Biases in Recommendation Data

Types	Stage in Loop	Data	Cause	Effect
Selection Bias	User→Data	Explicit feedback	Users' self -selection	Skewed observed rating distribution
Exposure Bias	User→Data	Implicit feedback	Users' self-selection; Background; Intervened by systems; Popularity	Unreliable non-positive data
Conformity Bias	User→Data	Both	Conformity	Skewed labels
Position Bias	User→Data	Both	Trust top of lists; Exposed to top of lists	Unreliable positive data

$$P_T(u,i,r) = P_T\ (u,i)P_T(r|u,i)$$

$$P_D(u,i,r) = P_D\ (u,i)P_D(r|u,i)$$

• Selection Bias

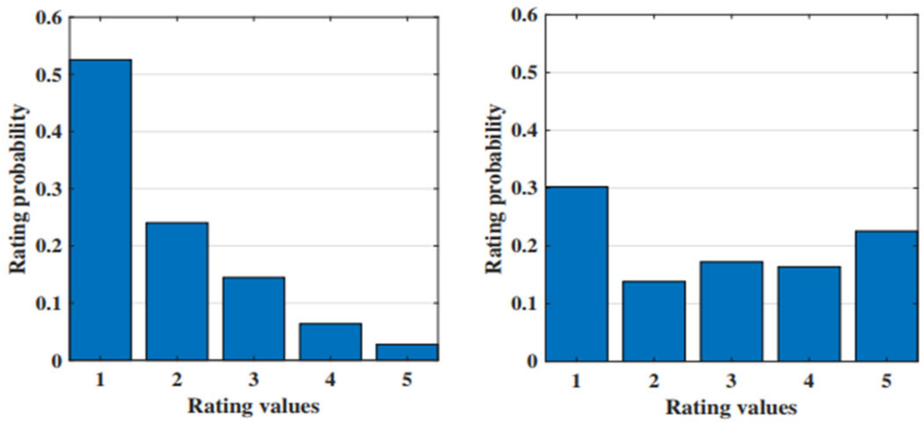
- Definition: *Selection bias happens in explicit feedback data as users are free to choose which items to rate, so that the observed ratings are not a representative sample of all ratings.*

3	4	2	5
1	3	2	5
2	3	4	4

Selection bias →

$$p_T(u,i) \neq p_D(u,i)$$

3	4		5
	3		5
	3	4	4



(a) Random (b) User-selected

[1] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. In ICML.
[2] B. M. Marlin, R. S. Zemel, S. Roweis, and M. Slaney, "Collaborative filtering and the missing at random assumption," in UAI, 2007

• Exposure Bias

- Definition: *Exposure bias happens in implicit feedback data as users are only exposed to a part of specific items.*
- Explanation: A user generates behaviors on exposed items, making the observed user-item distribution $p_T(u,i)$ deviate from the ideal one $p_D(u,i)$.

1	1	0	1
0	1	0	1
0	1	1	1

Exposure bias

$\longrightarrow$

$p_T(u,i) \neq p_D(u,i)$

1	1		
0		0	
0	1	1	1

• Exposure Bias

Unware

Dislike

1	1	0	1
0	1	0	1
0	1	1	1

Exposure bias →

$$p_T(u,i) \neq p_D(u,i)$$

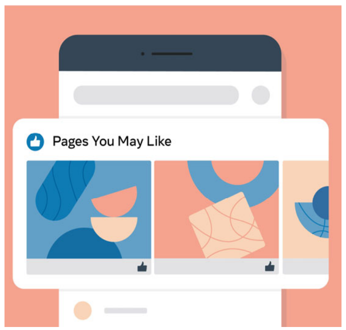
1	1		
0		0	
0	1	1	1

Implicit feedback →

$$p_T(u,i \mid r=0)=0$$

1	1		
	1	1	1

Exposure Policy of RS



User Background

Item Popularity

• Conformity Bias

- Definition: *Conformity bias happens as users tend to behave similarly to the others in a group, even if doing so goes against their own judgment.*

3	4		5
	3		4
	3	4	3

Conformity bias →

$$p_T(r|u,i) \neq p_D(r\,|\,u,i)$$

3	4		5
	3		5
	3	3	4

3	4	3	5

Public opinions

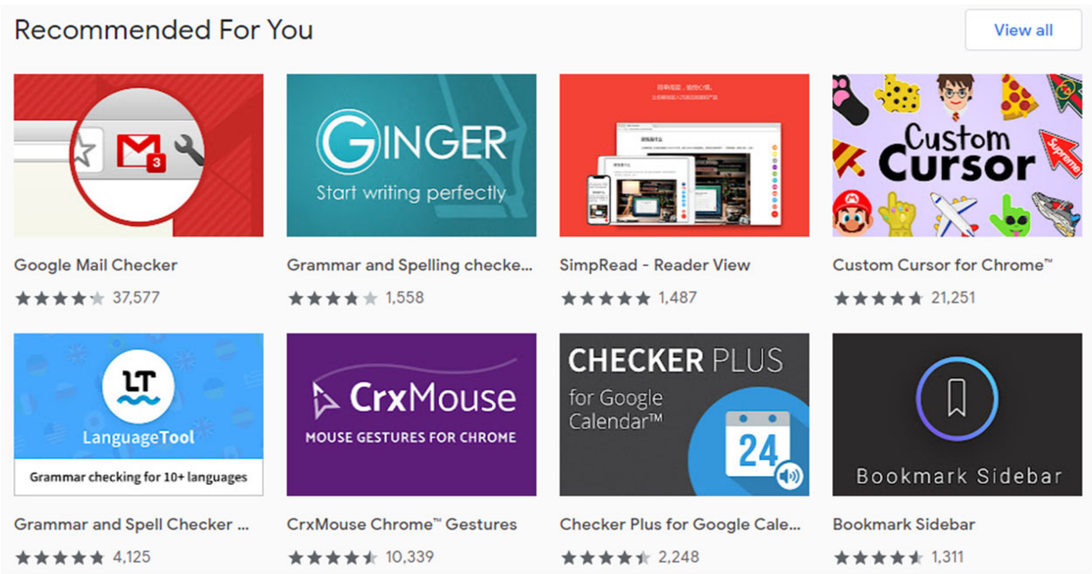
• Position Bias

- Definition: *Position bias happens as users tend to interact with items in higher position of the recommendation list.*

1	1	0	1
0	1	0	1
0	1	1	1

Position bias →

1	1	1	
		1	
		1	1



$$p_T(u,i) \neq p_D(u,i)$$

User exposure will be affected by the position

$$p_T(r \mid u,i) \neq p_D(r \mid u,i)$$

User judgments also will be affected by the position

• Debiasing Strategies Overview

- Re-weighting
  - Giving weights for each instance to re-scale their contributions on model training
- Re-labeling
  - Giving a new pseudo-label for the missing or biased data
- Generative Modeling
  - Assuming the generation process of data and reduces the biases accordingly

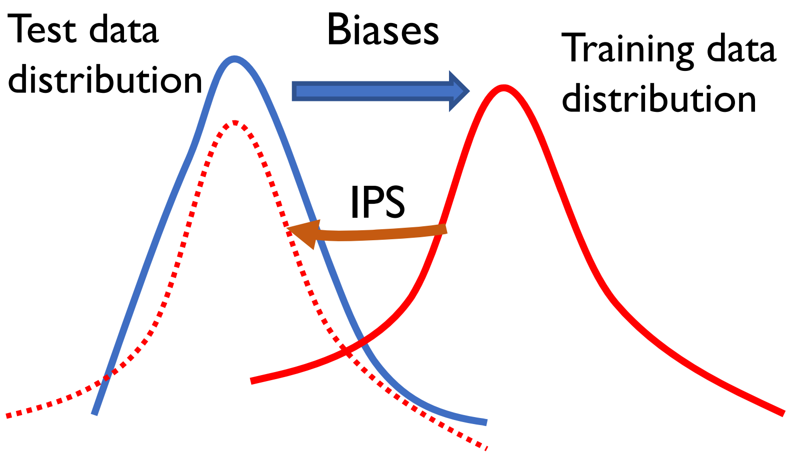
• Re-weighting Strategies

- Basic idea: change data distribution by sample reweighting:

$$L_{ips} = \sum_{(u,i)\in D_T} \frac{1}{\rho_{ui}} \delta(r_{ui}, \hat{r}_{ui})$$

- Mainly addressing the deviation of $p(u,i)$

$$p_T(u,i) \neq p_D(u,i)$$



- Properly defining weights can lead to *unbiased estimator* of the ideal:

$$L(f) = E_{P_D(u,i)P_D(r|u,i)}[\delta(r_{ui}, \hat{r}_{ui})] \neq E[\hat{L}_T(f)] = E_{P_T(u,i)P_T(r|u,i)}[\delta(r_{ui}, \hat{r}_{ui})]$$

$$\boxed{\frac{P_D(u,i)}{P_T(u,i)} = \frac{1}{\rho_{ui}}}$$

Inverse propensity Scores (IPS) $=$

$$E[\hat{L}_{IPS}(f)] = E_{\cancel{P_T(u,i)}P_T(r|u,i)}\left[\boxed{\frac{P_D(u,i)}{\cancel{P_T(u,i)}}}\delta(r_{ui}, \hat{r}_{ui})\right]$$

• Propensity Score for Biases (Reweighting)

3	4	2	5
1	3	2	5
2	3	4	4

Selection bias →

$p_T(u,i) \neq p_D(u,i)$

3	4		5
	3		
2	3	4	4

Reweighting - - - →

3	4		5
	3		
2	3	4	4

Recommendations

q_1 1. The Godfather

q_2 2. The Matrix

q_3 3. The Dark Knight

q_4 4. The Wolf of Wall Street

q_5 5. Titanic

$$L_{IPW}(S,q) = \sum_{x\in\pi_q} \Delta_{IPW}(x,y \mid \pi_q)$$

$$= \sum_{x\in\pi_q, o_q^x=1, y=1} \frac{\Delta(x,y \mid \pi_q)}{\boxed{P(o_q^x = 1 \mid \pi_q)}}$$

Simple and straightforward.

Theoretical soundness.

High Variance.

Difficult to set proper propensity score.

Requires positivity.

Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. In ICML

T. Joachims, A. Swaminathan, and T. Schnabel, "Unbiased learning-to-rank with biased feedback," in WSDM, 2017, pp. 781–789

• How to Set Proper Propensity?

- Intervene the system.
  - Position bias: randomly permutation
  - Selection bias: randomly selection

Intervene the system would harm user satisfactory.

- Inference from the observed data.
  - Training a classifier for selection or exposure.

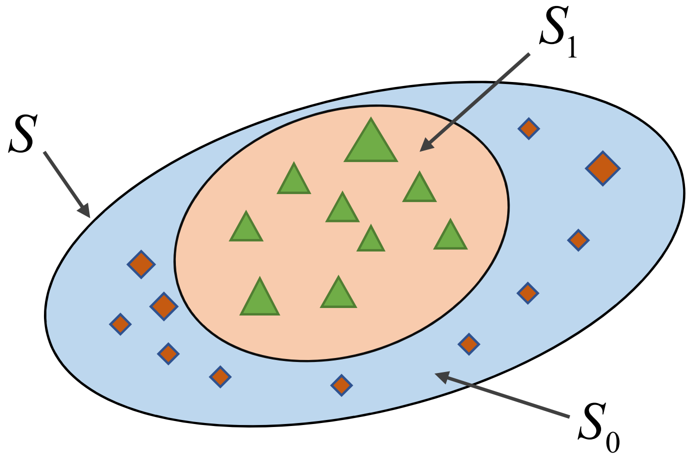
$$P_T(u,i) = Classifier(x_u, x_i, r)$$

Approximation.

[1] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. In ICML
[2] T. Joachims, A. Swaminathan, and T. Schnabel, 2017. Unbiased learning-to-rank with biased feedback. In WSDM
[3] Q. Ai, K. Bi, C. Luo, J. Guo, and W. B. Croft, 2018. Unbiased learning to rank with unbiased propensity estimation. In SIGIR.
[4] Z. Qin, S. J. Chen, D. Metzler, Y. Noh, J. Qin, and X. Wang, 2020. "Attribute-based propensity for unbiased learning in recommender systems: Algorithm and case studies. In KDD

• Limitation of Reweighting: Requiring positivity

- Just leveraging propensity score is insufficient:



$S : \{(u,i,r) : p_U(u,i,r) > 0\}$

$S_0 : \{(u,i,r) : p_U(u,i,r) > 0, p_T(u,i,r) = 0\}$

$S_1 : \{(u,i,r) : p_U(u,i,r) > 0, p_T(u,i,r) > 0\}$

▲ : Training data

◆ : Imputed data

  - Due to the data bias, training data distribution P_T may only provide the partial data knowledge of the region S (S_0 is not included)
  - IPS cannot handle this situation
- Imputing pseudo-data to the region S_0:

$$L_T = \sum_{(u,i)\in D_T} w_{ui}^{(1)} \delta(r_{ui}, \hat{r}_{ui}) + \boxed{\sum_{u\in U, i\in I} w_{ui}^{(2)} \delta(m_{ui}, \hat{r}_{ui})}$$

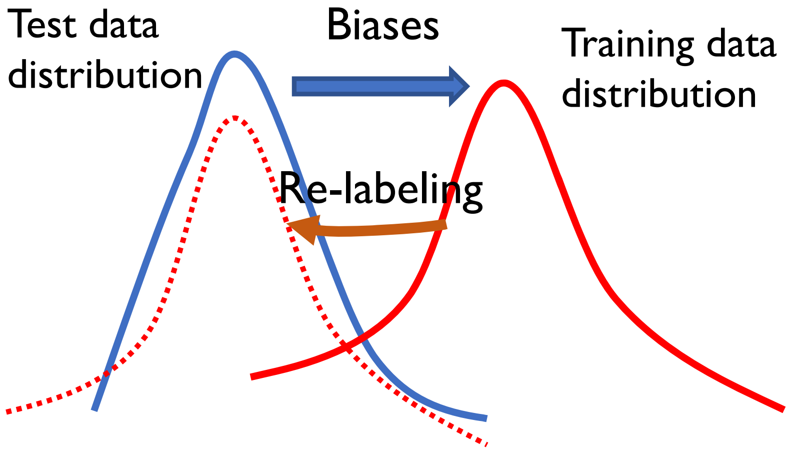
• Debiasing Strategies Overview

- Re-weighting
  - Giving weights for each instance to re-scale their contributions on model training
- Re-labeling
  - Giving a new pseudo-label for the missing or biased instance
- Generative Modeling
  - Assuming the generation process of data and reduces the biases accordingly

• Re-labeling Strategies

- Basic idea: change data distribution by imputing pseudo-labels:

$$L_{DI} = \sum_{(u,i)\in D_T \vee D_m} \delta(r_{ui} \backslash m_{ui}, \hat{r}_{ui})$$



- Could address the deviation of $p(u,i)$ and $p(r|u,i)$

$$p_T(u,i) \neq p_D(u,i) \qquad p_T(r\,|\,u,i) \neq p_D(r\,|\,u,i)$$

- Properly defining pseudo-labels can lead to *unbiased estimator* of the ideal:

For $p_T(r\,|\,u,i) \neq p_D(r\,|\,u,i)$ $\Rightarrow$ $L_{DI} = \sum_{(u,i,r)\in D_T} \delta(m_{ui}, \hat{r}_{ui}), m_{ui} \sim p_D(r\,|\,u,i)$

For $p_T(u,i) \neq p_D(u,i)$ $\Rightarrow$ $L_{DI} = \sum_{(u,i,r)\in D_T} \delta(r_{ui}, \hat{r}_{ui}) + \sum_{(u,i)-D_T} \delta(m_{ui}, \hat{r}_{ui})$

• Data imputation for Selection Bias (Relabeling)

True Preference

3	4	2	5
1	3	2	5
2	3	4	4

Selection bias

$$\xrightarrow{}$$

$$p_T(u,i) \neq p_D(u,i)$$

Training data

3	4		5
	3		5
2	3	4	4

Data imputation

Imputation data

3	4	2	5
2	3	2	5
2	3	4	4

- Relabeling: assigns pseudo-labels for missing data.

$$\arg\min_{\theta} \sum_{u,i} \hat{\delta}\left(\boxed{r_{ui}^{o\&i}}, f(u,i\,|\,\theta)\right) + \mathrm{Reg}(\theta)$$

Simple and straightforward.

Sensitive to the imputation strategy. Imputing proper pseudo-labels is more difficult.

H. Steck, "Training and testing of recommender systems on data missing not at random," in KDD, 2010, pp. 713–722.

X. Wang, R. Zhang, Y. Sun, and J. Qi, "Doubly robust joint learning for recommendation on data missing not at random," in ICML, 2019, pp. 6638–6647

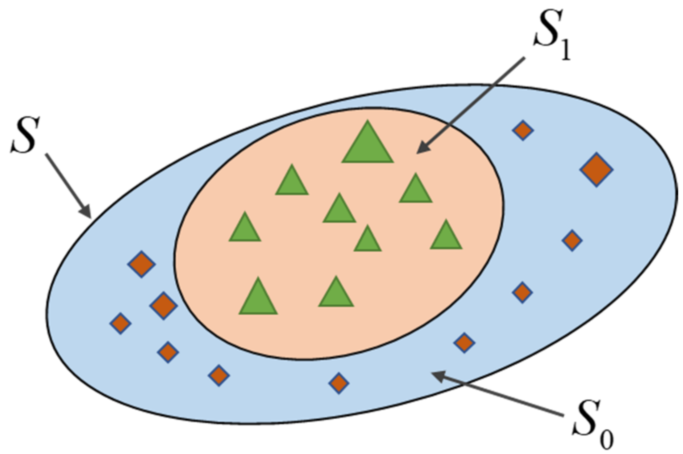
• Relabeling+Reweighting

- Reweighting:
  - Relatively Robust
  - High variance; Requires positivity

+

- Relabeling:
  - General
  - Sensitive to pseudo-labels

$$L_T = \sum_{(u,i)\in D_T} w_{ui}^{(1)}\,\delta(r_{ui}, \hat{r}_{ui}) + \sum_{u\in U, i\in I} w_{ui}^{(2)}\,\delta(m_{ui}, \hat{r}_{ui})$$



• Doubly Robust for Selection Bias (Relabeling+Reweighting)

3	4	2	5
1	3	2	5
2	3	4	4

Selection bias $\longrightarrow$

$p_T(u,i) \neq p_D(u,i)$

3	4		5
	3		
2	3	4	4

Relabeling+ Reweighting $\dashrightarrow$

3	4	2	5
2	3	2	4
2	3	4	4

- Doubly Robust: combines IPS and data imputation for robustness.

$$\hat{L}_{DR} = \sum_{(u,i)\in D_T} \underbrace{\frac{1}{\rho_{ui}}}_{\text{IPS}}\left(\delta\left(\hat{r}_{ui}, r_{ui}\right)\right) + \sum_{u\in U, i\in I} \underbrace{\left(1-\frac{O_{ui}}{\rho_{ui}}\right)\delta\left(\hat{r}_{ui}, m_{ui}\right)}_{\text{Imputation}}$$

$$O_{ui} = \mathbf{I}[(u,i)\in D_T]$$

Low Variance.

Relatively robust to the propensity score and imputation value.

Requires proper imputation or propensity strategy.

Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2019. Doubly robust joint learning for recommendation on data missing not at random. In ICML.

• Relabeling+Reweighting for Exposure Bias

1	1		1
	1		1
	1	1	1

Exposure bias $\longrightarrow$ $p_T(u,i) \neq p_D(u,i)$

1	1		
	1	1	1

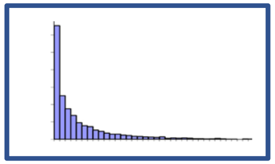
Imputation +Weighting

1	1	0	0
0	0	0	0
0	1	1	1

$$L_w = \sum_{(u,i)\in D_T} \frac{1}{\rho_{ui}} \delta\left(r_{ui}, \hat{r}_{ui}\right) + \sum_{u\in U, i\in I} w_{ui}^{(2)} \delta\left(0, \hat{r}_{ui}\right)$$

- Imputing zero for unobserved data and downweight their contribution.
- $w_{ui}^{(2)}$ reflects how likely the item is exposed to the user.

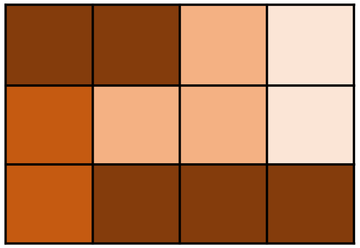
Item popularity



Social network

User community

weighting



• Relabeling+Reweighting for Position Bias

$$\boxed{p_T(r \mid u,i) \neq p_D(r \mid u,i)}$$

User judgments also will be affected by the position

$$\Pr(\tilde{R}=1 \mid E=1,q,d,k)$$
$$=\Pr(\tilde{R}=1 \mid R=1,E=1,k)\Pr(R=1 \mid q,d)$$
$$+\Pr(\tilde{R}=1 \mid R=0,E=1,k)\Pr(R=0 \mid q,d)$$

• **Affine model (Reweighting+Relabeling)**

$$\hat{\Delta}_{\text{affine}}(f)=\frac{1}{N}\sum_{i=1}^{N}\sum_{(d,k)\in y_i}\frac{c_i(d)-\theta_k\epsilon_k^-}{\theta_k\left(\epsilon_k^+-\epsilon_k^-\right)}\cdot\lambda\left(d \mid q_i,f\right)$$

$$\epsilon_k^+=\Pr(\tilde{R}=1 \mid R=1,E=1,k)$$
$$\epsilon_k^-=\Pr(\tilde{R}=1 \mid R=0,E=1,k)$$

Vardasbi, Ali, Harrie Oosterhuis, and Maarten de Rijke. "When Inverse Propensity Scoring does not Work: Affine Corrections for Unbiased Learning to Rank." CIKM, 2019, pp. 1475-1484. 2020.

• A summary of Relabeling+Reweighting

- Optimizes:

$$L_T = \sum_{(u,i)\in D_T} w_{ui}^{(1)}\, \delta(r_{ui}, \hat{r}_{ui}) + \sum_{u\in U, i\in I} w_{ui}^{(2)}\, \delta(m_{ui}, \hat{r}_{ui})$$

- Inherits the merits of Relabeling and Reweighting.
- Depend on proper weights and pseudo-labels.
- Relies on heuristical design.

Lack of Universality.

Lack of adaptivity.

- Is there a universal and adaptive solution?

• AutoDebias: a Universal Solution (Relabeling+Reweighting)

learn from uniform data:

Uniform data provides signal on the effectiveness of debiasing

Meta learning mechanism:

Base learner: optimize rec model with fixed ϕ

$$\theta^*(\phi) = \arg\min_{\theta} \sum_{(u,i)\in D_T} w_{ui}^{(1)} \delta\left(r_{ui}, \hat{r}_{ui}(\theta)\right) + \sum_{u\in U, i\in I} w_{ui}^{(2)} \delta\left(m_{ui}, \hat{r}_{ui}(\theta)\right)$$

Meta learner: optimize debiasing parameters on uniform data

$$\phi^* = \arg\min_{\phi} \sum_{(u,i)\in D_U} \delta\left(r_{ui}, \hat{r}_{ui}(\theta^*)\right)$$

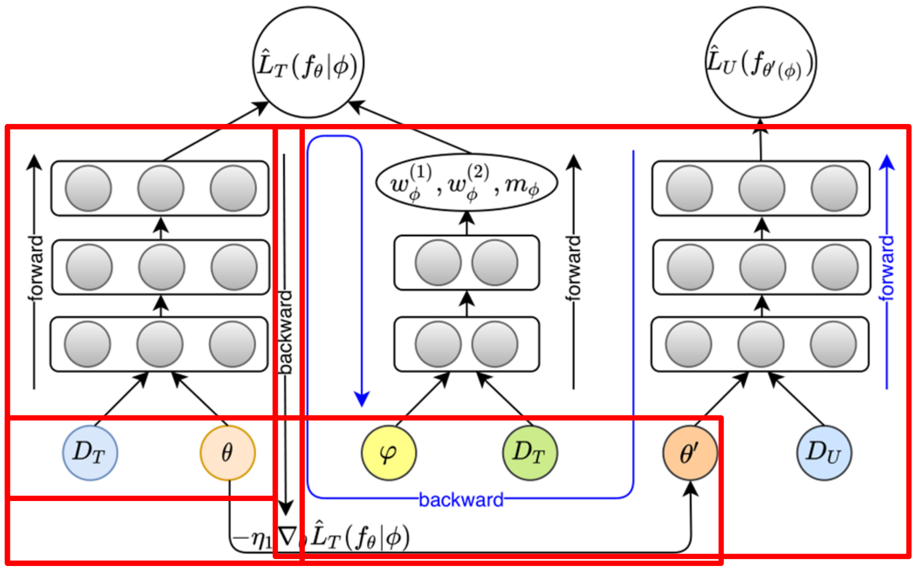
• AutoDebias: a Universal Solution (Relabeling+Reweighting)

- Two challenges:
  - Overfitting: small uniform data but many debiasing parameters ϕ
    - Solution: Introduce a small meta model to generate ϕ, e.g., linear model

$$w_{ui}^{(1)} = \exp(\varphi_1^T[\mathbf{x}_u \circ \mathbf{x}_i \circ \mathbf{e}_{y_{ui}}]), \quad w_{ui}^{(2)} = \exp(\varphi_2^T[\mathbf{x}_u \circ \mathbf{x}_i \circ \mathbf{e}_{O_{ui}}]), \quad m_{ui} = \sigma(\varphi_3^T[\mathbf{e}_{y_{ui}} \circ \mathbf{e}_{O_{ui}}])$$

  - Inefficiency: obtaining optimal ϕ involves nested loops of optimization
    - Solution: Update recsys model and debiasing parameters alternately in a loop

- Step 1: Make a tentative update of θ to θ' with current ϕ
- Step 2: Test θ' on uniform data, which gives feedback to update ϕ
- Step 3: Update θ actually with updated ϕ



• AutoDebias: a Universal Solution (Relabeling+Reweighting)

- Evaluate AutoDebias on two Yahoo!R3 and Coat (Explicit setting with selection bias)

Methods	On Yahoo!R3		On Coat	
	AUC	NDCG@5	AUC	NDCG@5
MF(biased)	0.727	0.550	0.747	0.500
MF(uniform)	0.573	0.449	0.580	0.358
MF(combine)	0.730	0.554	0.750	0.504
IPS	0.723	0.549	0.759	0.509
DR	0.723	0.552	0.765	0.521
CausE	0.731	0.551	0.762	0.500
KD-Label	0.740	0.580	0.748	0.504
AutoDebias-w1	0.733	0.573	0.762	0.510
AutoDebias	**0.741**	**0.645**	**0.766**	**0.522**

- AutoDebias outperforms state-of-the-arts methods
- AutoDebias>AutoDebias-w1: Introducing imputation strategy is effectiveness
- AutoDebias-w1>IPS: learning debiasing parameters from uniform data is superior over simple statistics

• AutoDebias: a Universal Solution (Relabeling+Reweighting)

- Evaluate AutoDebias on Yahoo!R3-im and Coat-im (Implicit setting with exposure bias)
- Evaluate AutoDebias on synthetic dataset (with selection bias + position bias)

Method	Yahoo!R3-Im		Coat-Im	
	AUC	NDCG@5	AUC	NDCG@5
WMF	0.635	0.547	**0.749**	0.521
RI-MF	0.673	0.554	0.696	**0.527**
AWMF	0.675	0.578	0.614	0.505
AutoDebias	**0.730**	**0.635**	0.746	**0.527**

	NLL	AUC	NDCG@5
MF(biased)	-0.712	0.564	0.589
DLA	-0.698	0.567	0.593
HeckE	-0.688	0.587	0.648
AutoDebias	**-0.667**	**0.634**	**0.707**

- AutoDebias consistently outperform state-of-the-art in both addressing exposure bias and bias combinations.

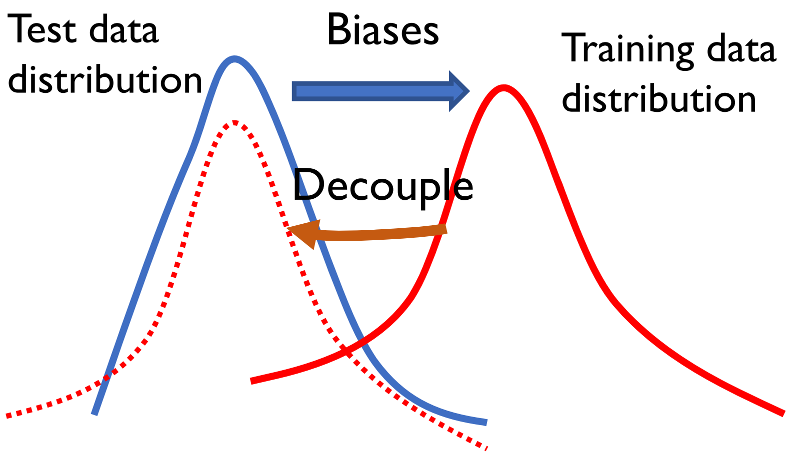
It requires uniform data.
It lacks of explanation.

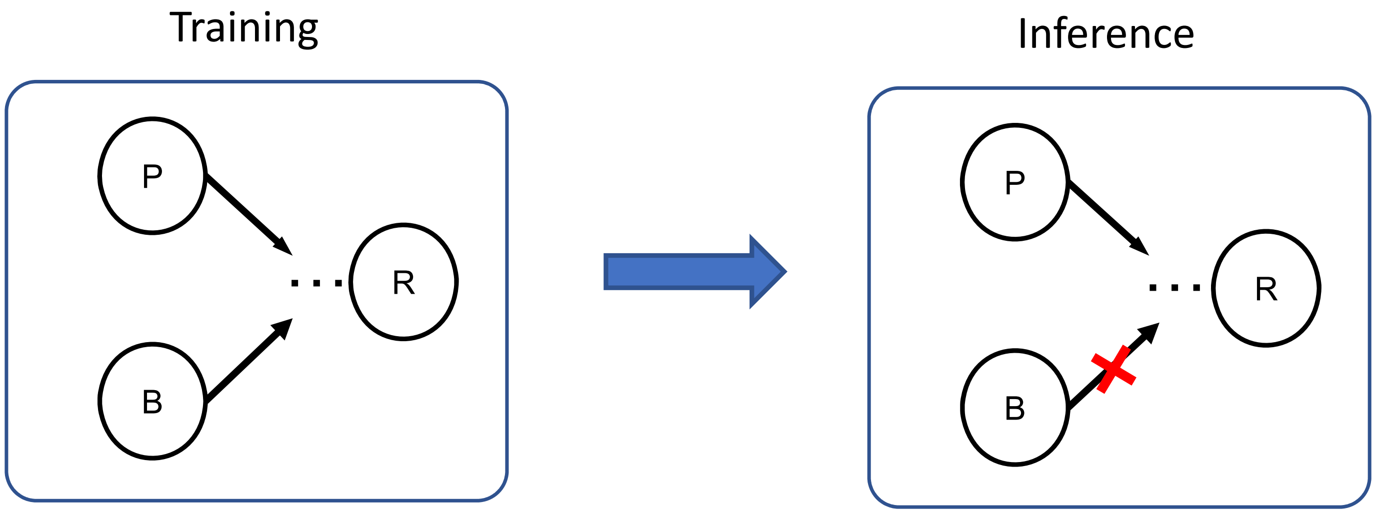
• Debiasing Strategies Overview

- Re-weighting
  - Giving weights for each instance to re-scale their contributions on model training
- Re-labeling
  - Giving a new pseudo-label for the missing or biased instance
- Generative Modeling
  - Assuming the generation process of data and reduces the biases accordingly

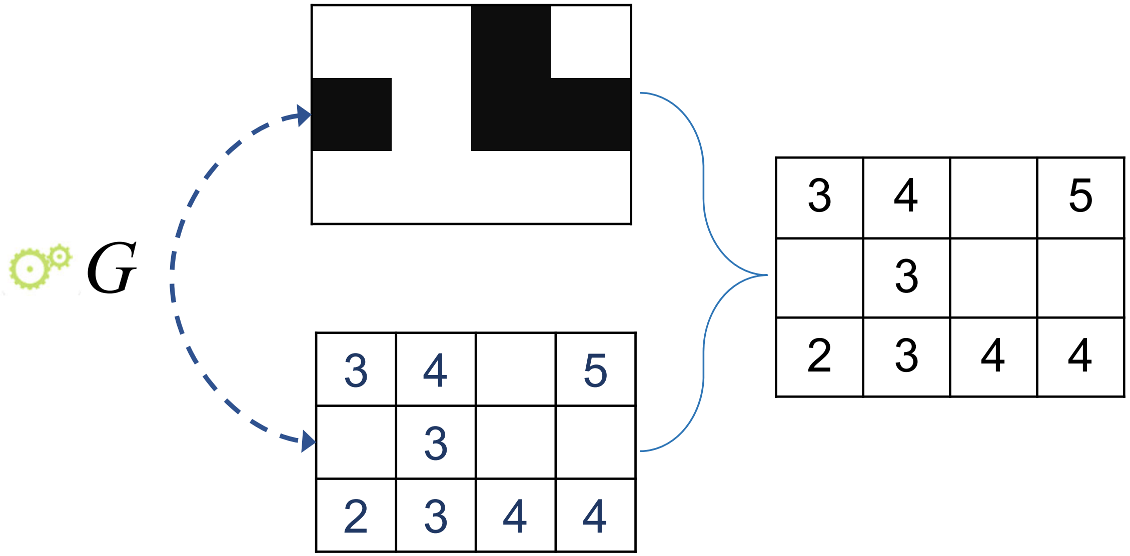
• Generative Modeling

- Basic idea: assuming the generation process of data to decouple the effect of user true preference from the bias.





• Generative Model for Selection Bias



- Generative Model: jointly modeling rating values and user selection.

Explainable.

Complex and sophisticated models.

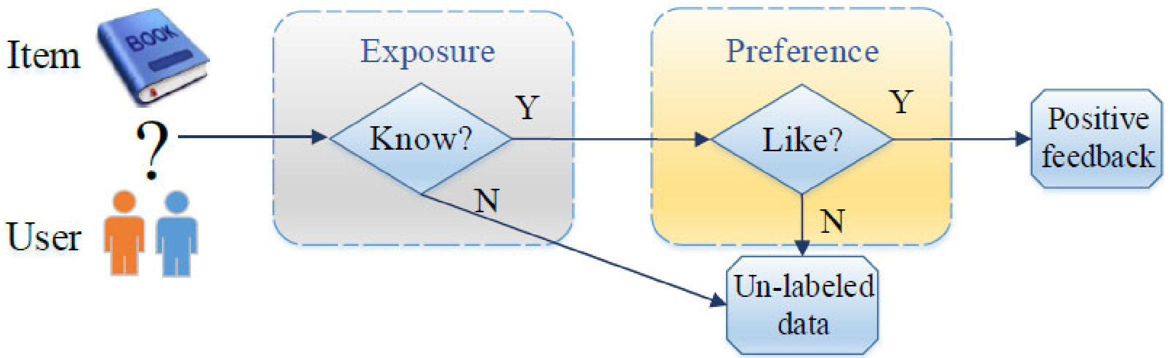
Hard to train.

B. M. Marlin and R. S. Zemel, "Collaborative prediction and ranking with non-random missing data," in RecSys, 2009, pp. 5– 12.

J. M. Hern´andez-Lobato, N. Houlsby, and Z. Ghahramani, "Probabilistic matrix factorization with non-random missing data." in ICML, 2014, pp. 1512–1520.

J. Chen, C. Wang, M. Ester, Q. Shi, Y. Feng, and C. Chen, "Social recommendation with missing not at random data," in ICDM. IEEE, 2018, pp. 29–38.

• Exposure Model for Exposure Bias (Generative modeling)



$$a_{ui} \sim Bernoulli(\eta_{ui})$$
$$(r_{ui} \mid a_{ui} = 1) \sim Bernoulli(f(u,i \mid \theta))$$
$$(r_{ui} \mid a_{ui} = 0) \sim \delta_0$$

$$\underset{\theta,\gamma}{\operatorname{argmin}} \sum_{ui} \gamma_{ui} \delta\left(r_{ui}, f(u,i \mid \theta)\right) + \sum_{ui} g\left(\gamma_{ui}\right) \qquad \gamma_{ui} \approx p(a_{ui} \mid r_{ui})$$

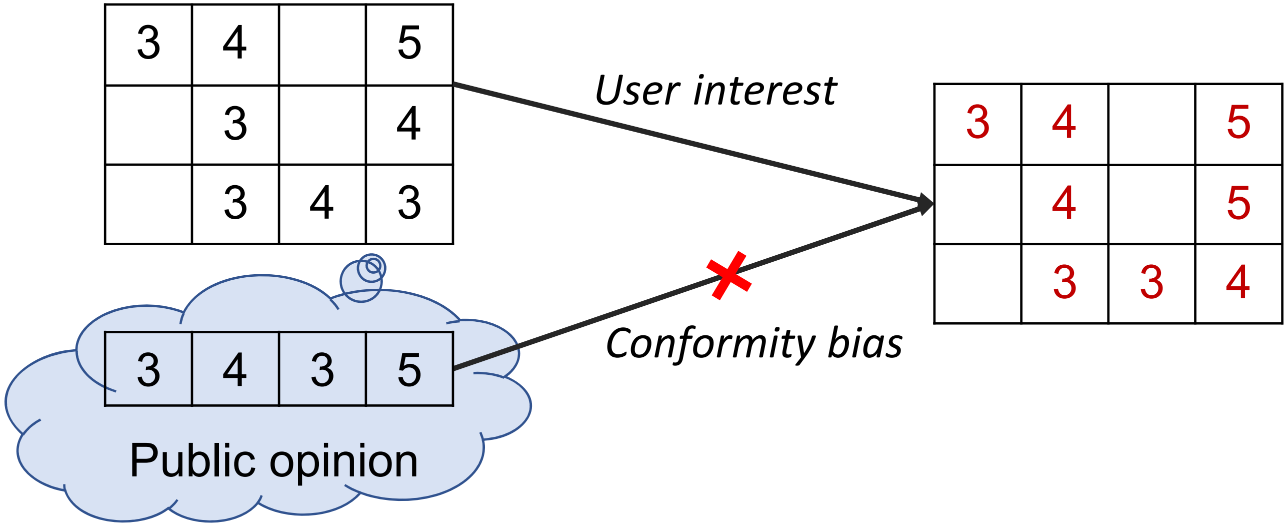
- Generative model: jointly modeling both user exposure and preference.

  Personalized.
  Learnable.

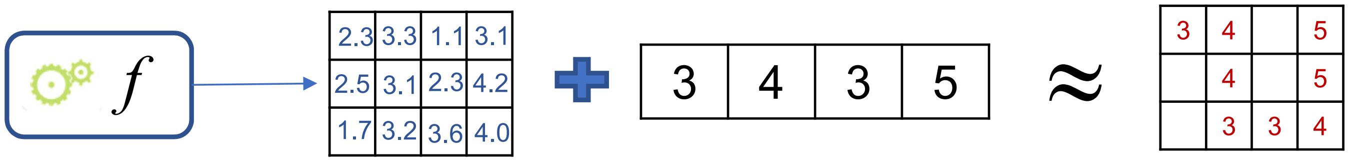
  Hard to train.
  Relying on strong assumptions.

D. Liang, L. Charlin, J. McInerney, and D. M. Blei, "Modeling user exposure in recommendation," in WWW. 2016
J. Chen, C. Wang, S. Zhou, Q. Shi, Y. Feng, and C. Chen, "Samwalker: Social recommendation with informative sampling strategy," in The World Wide Web Conference. ACM, 2019, pp. 228–239.

• Disentangling for Conformity Bias (Generative modeling)



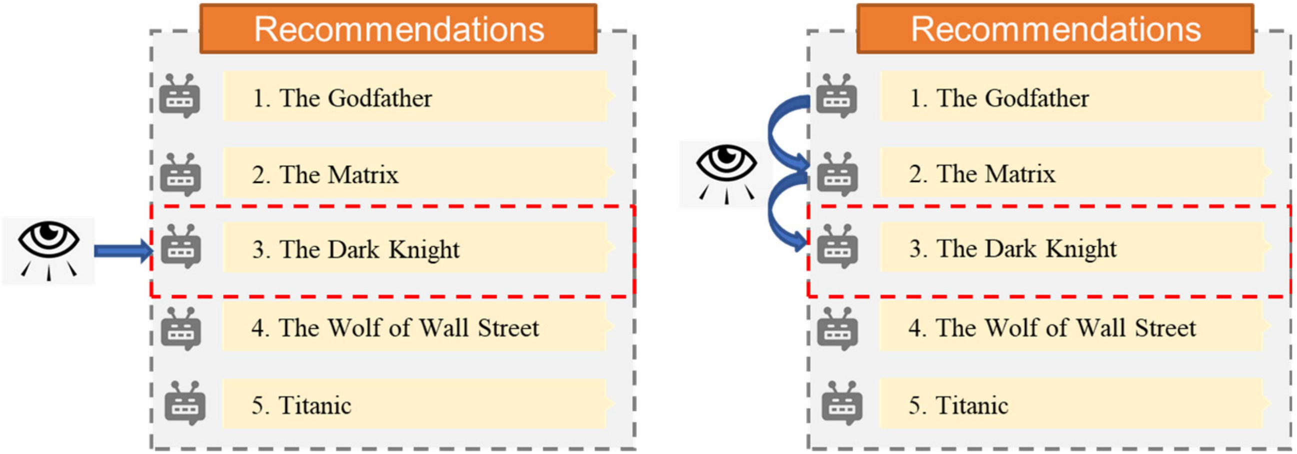
- Disentangling: disentangle the effect of user interest and conformity.



Y. Liu, X. Cao, and Y. Yu, "Are you influenced by others when rating?: Improve rating prediction by conformity modeling," in RecSys. ACM, 2016, pp. 269–272.

A.~J. Chaney, D.~M. Blei, and T.~Eliassi-Rad, ``A probabilistic model for using social networks in personalized item recommendation,'' in RecSys, ACM, 2015

• Click model for Position Bias (Generative modeling)



$$P(C=1 \mid u,i,p) = \underbrace{P(C=1 \mid u,i,E=1)}_{r_{ui}} \cdot \underbrace{P(E=1 \mid p)}_{h_p}$$

$$P\left(E_{p+1}=1 \mid E_p=0\right)=0$$
$$P\left(E_{p+1}=1 \mid E_p=1, C_p\right)=1-C_p$$
$$P\left(C_p=1 \mid E_p=1\right)=r_{u_p,i}$$

- Click model: making hypotheses about user browsing behaviors and learn true preference (or relevant) by optimizing likelihood of the observed clicks .

Explainable.

Requiring a large quantity of clicks.
Requiring strong assumptions.

O. Chapelle and Y. Zhang, "A dynamic bayesian network click model for web search ranking," in WWW, 2009, pp. 1–10.

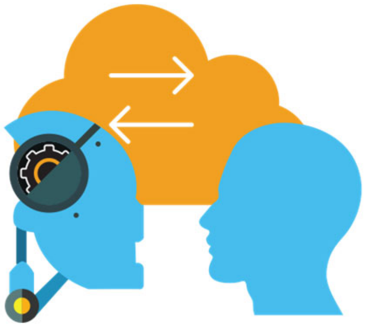
F. Guo, C. Liu, A. Kannan, T. Minka, M. Taylor, Y.-M. Wang, and C. Faloutsos, "Click chain model in web search," in WWW, 2009, pp. 11–20.

Z. A. Zhu, W. Chen, T. Minka, C. Zhu, and Z. Chen, "A novel click model and its applications to online advertising," in WSDM, 2010, pp. 321–330.

• Debiasing Strategies Overview

- Re-weighting
  - Giving weights for each instance to re-scale their contributions on model training
  - Advantages: simple, theoretical soundness, relatively robust to the weights
  - Limitations: high variance, requires positivity, hard to set proper propensity
- Re-labeling
  - Giving a new pseudo-label for the missing or biased instance
  - Advantages: simple, general
  - Limitations: inefficiency, very sensitive to pseudo-label, hard to set pseudo-label
- Generative Modeling
  - Assuming the generation process of data and reduces the biases accordingly
  - Advantages: leveraging human prior knowledge, explainable
  - Limitations: hard to train, strong assumptions

• Debiasing Strategies Overview



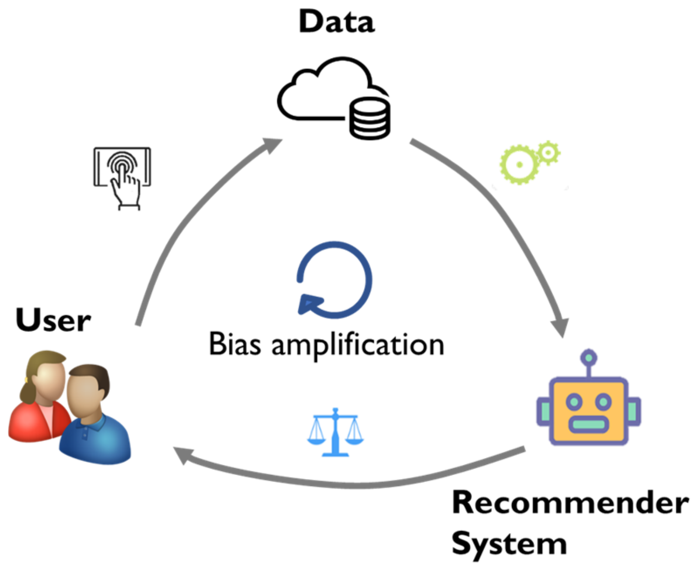
1 For Selection Bias
- Data imputation (Relabeling)
- Propensity score (Reweighting)
- Doubly robust model (Reweighting+Relabeling)
- Generative model (Generative modeling)

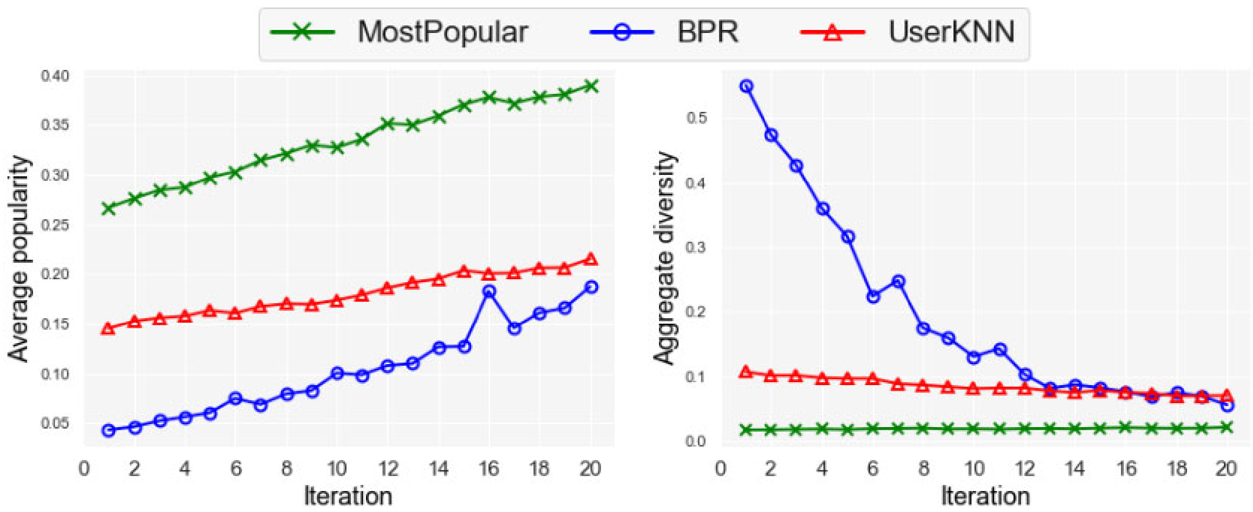
2 For Conformity Bias
- Disentangling (Generative modeling)

3 For Exposure Bias
- Downweighting (Reweighting+Relabeling)
- Exposure-based generative model (Generative modeling)

4 For Position Bias
- Click model (Generative modeling)
- Propensity score (Reweighting)
- Trust-aware model (Reweighting+Relabeling)

• Feedback Loop Amplifies Biases

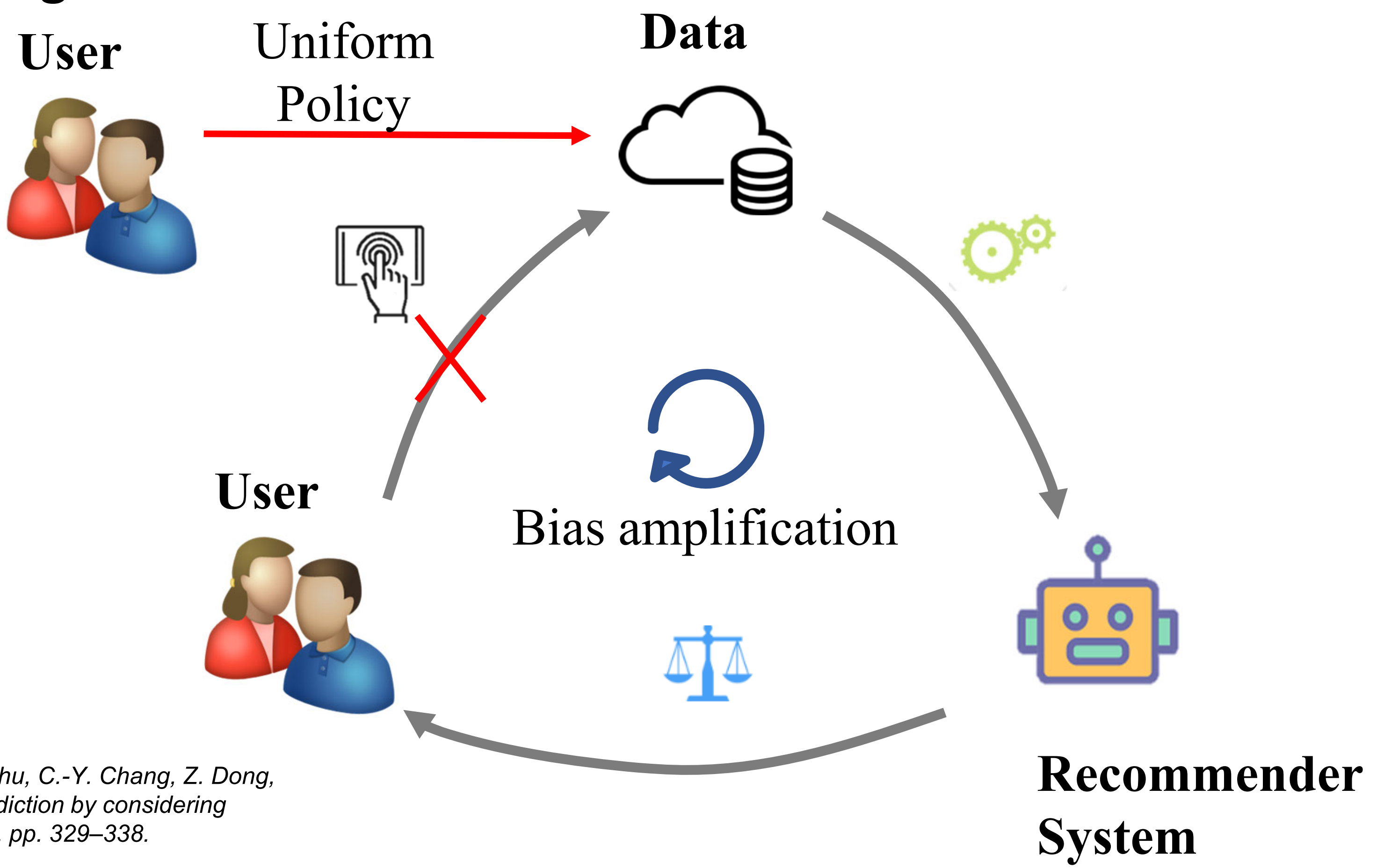




- The average popularity of the recommendation items are increasing while the diversity are decreasing along the feedback loop.

• Solution for Bias Amplification

- Leveraging uniform data.



B. Yuan, J.-Y. Hsia, M.-Y. Yang, H. Zhu, C.-Y. Chang, Z. Dong, and C.-J. Lin, "Improving ad click prediction by considering nondisplayed events," in CIKM, 2019, pp. 329–338.

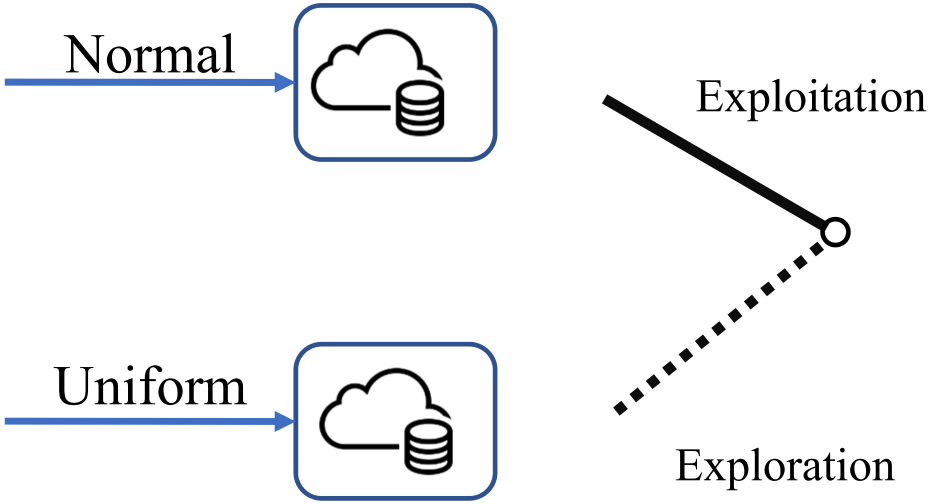
S. Bonner and F. Vasile, "Causal embeddings for recommendation," in RecSys, 2018, pp. 104–112.

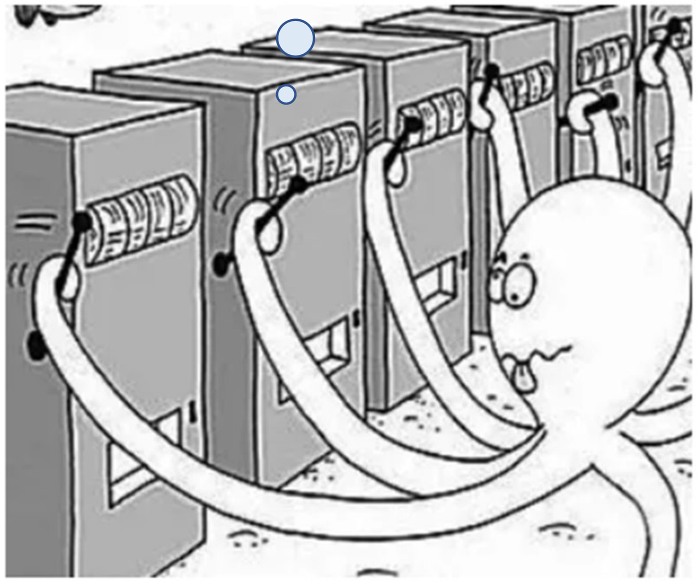
• Solution for Bias Amplification

- Interactive recommendation.

a recommender system can interact with a user and dynamically capture his preference

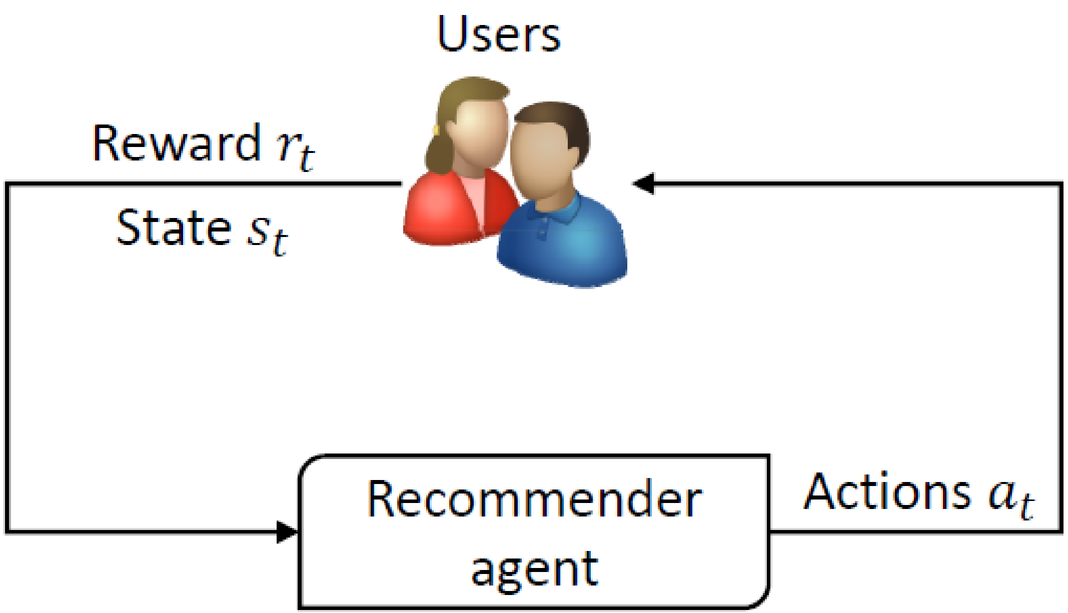
Exploration and Exploitation Balance





Bandit

• Solution for Bias Amplification



RL agent → Recommender system

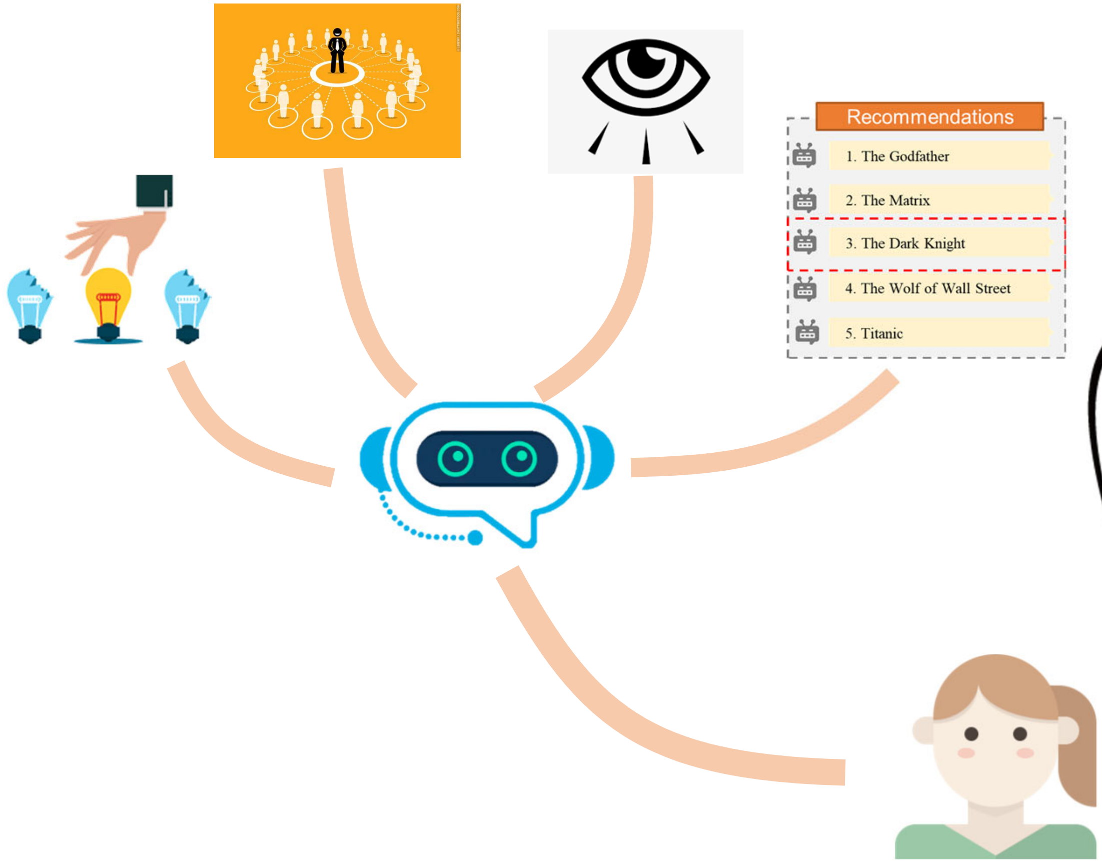
Reward → User feedback

Environment → User

Policy → Which items to be recommended

• Open Problem and Future Direction I

- A learnable universal solution.



How to develop a universal solution that accounts for multiple biases and their combinations?

I hope the system can adaptively adjust debiasing strategy according to the data.

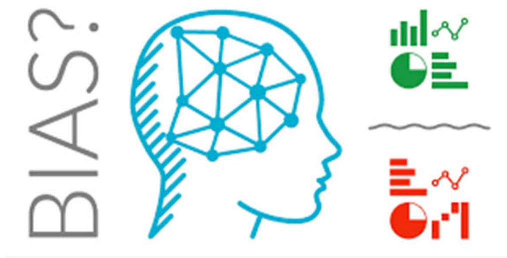
$G \rightarrow G^I$

• Open Problem and Future Direction II

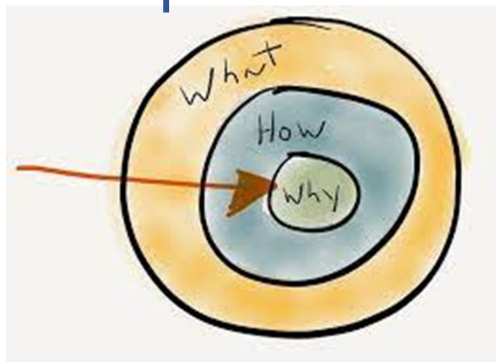
- **Knowledge-enhanced Debiasing**

Knowledge Graph

Address biases



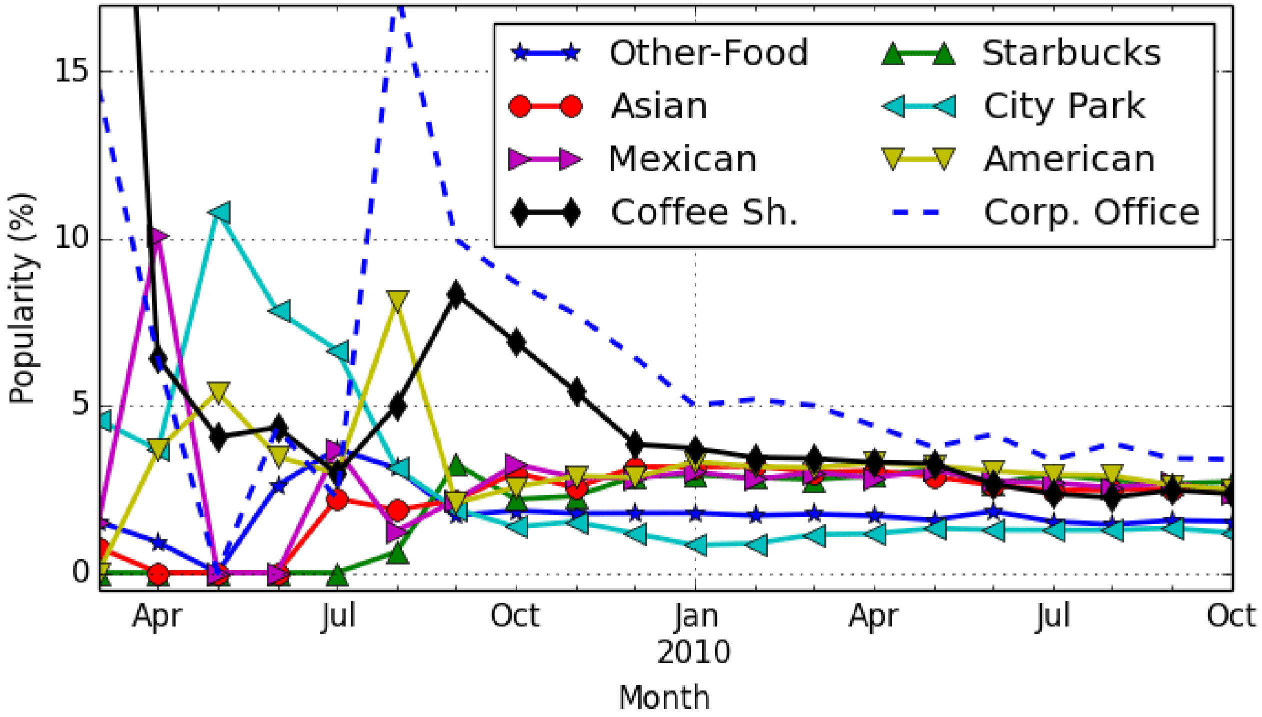
Explainable



- Leveraging human prior knowledge in better discovering biases in data
- Empowering knowledge graph to both address biases and give interpretation

• Open Problem and Future Direction III

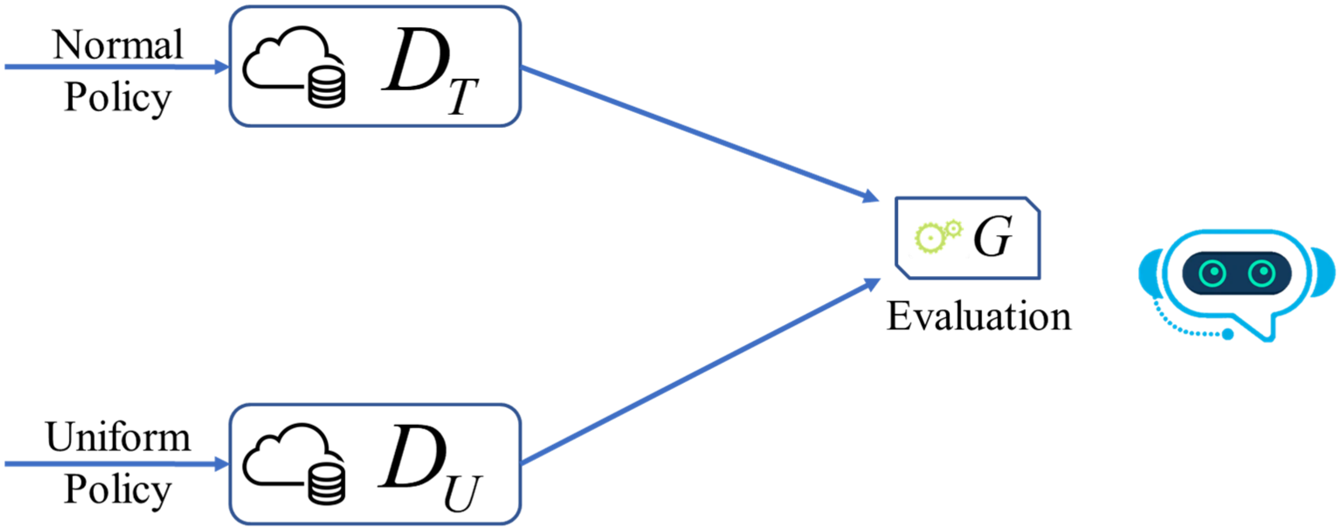
- Dynamic bias.



  - Biases are usually dynamic rather than static.
  - Online updating of debiasing strategies.

• Open Problem and Future Direction IV

- Better evaluation.



- Benchmark datasets and evaluation metrics.

Bias and Debias in Recommender System: A Survey and Future Directions

Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, Xiangnan He

Abstract—While recent years have witnessed a rapid growth of research papers on recommender system (RS), most of the papers focus on inventing machine learning models to better fit user behavior data. However, user behavior data is observational rather than experimental. This makes various biases widely exist in the data, including but not limited to selection bias, position bias, exposure bias, and popularity bias. Blindly fitting the data without considering the inherent biases will result in many serious issues, e.g., the discrepancy between offline evaluation and online metrics, hurting user satisfaction and trust on the recommendation service, etc. To transform the large volume of research models into practical improvements, it is highly urgent to explore the impacts of the biases and perform debiasing when necessary. When reviewing the papers that consider biases in RS, we find that, to our surprise, the studies are rather fragmented and lack a systematic organization. The terminology "bias" is widely used in the literature, but its definition is usually vague and even inconsistent across papers. This motivates us to provide a systematic survey of existing work on RS biases. In this paper, we first summarize seven types of biases in recommendation, along with their definitions and characteristics. We then provide a taxonomy to position and organize the existing work on recommendation debiasing. Finally, we identify some open challenges and envision some future directions, with the hope of inspiring more research work on this important yet less investigated topic.

Index Terms—Recommendation, Recommender System, Collaborative Filtering, Survey, Bias, Debias, Fairness

https://arxiv.org/pdf/2010.03240.pdf

paper/code link

Papers	Taxonomy 1	Taxonomy 2	Taxonomy 3	Date	Conference	Code
Collaborative filtering and the missing at random assumption	Bias in data	Bias in explicit feedback data	Selection Bias	2007	UAI	Python
Probabilistic matrix factorization with non-random missing data	Bias in data	Bias in explicit feedback data	Selection Bias	2014	PMLR	Python
Evaluation of recommendations: rating-prediction and ranking	Bias in data	Bias in explicit feedback data	Selection Bias	2013	RecSys	
Why amazon's ratings might mislead you: The story of herding effects	Bias in data	Bias in explicit feedback data	Conformity Bias	2014	Big data Volume: 2 Issue 4: December 15, 2014	
Are you influenced by others when rating?: Improve rating prediction by conformity modeling	Bias in data	Bias in explicit feedback data	Conformity Bias	2016	RecSys	
A methodology for learning, analyzing, and mitigating social influence bias in recommender systems	Bias in data	Bias in explicit feedback data	Conformity Bias	2014	RecSys	Python

https://github.com/jiawei-chen/RecDebiasing

cjwustc@ustc.edu.cn

• Tutorial Outline

- ❑ Biases in Data (Jiawei Chen, 60 min)
  - ❑ Definition of data biases
  - ❑ Categories: Selection bias, Conformity bias, Exposure bias and Position bias
  - ❑ Recent solutions for data biases
- ❑ Bias Amplification in Loop and its Solutions (Jiawei Chen, 10 min)
- ❑ Biases in Results
  - ❑ Popularity bias: definition, characteristic and solutions (Fuli Feng, 40 min)
  - ❑ Unfairness: definition, characteristic and solutions (Xiang Wang, 50 min)

slides will be available at: https://github.com/jiawei-chen/RecDebiasing
A literature survey based on this tutorial is available at: https://arxiv.org/pdf/2010.03240.pdf
.

Popularity Bias

➢ Definitions [1]:

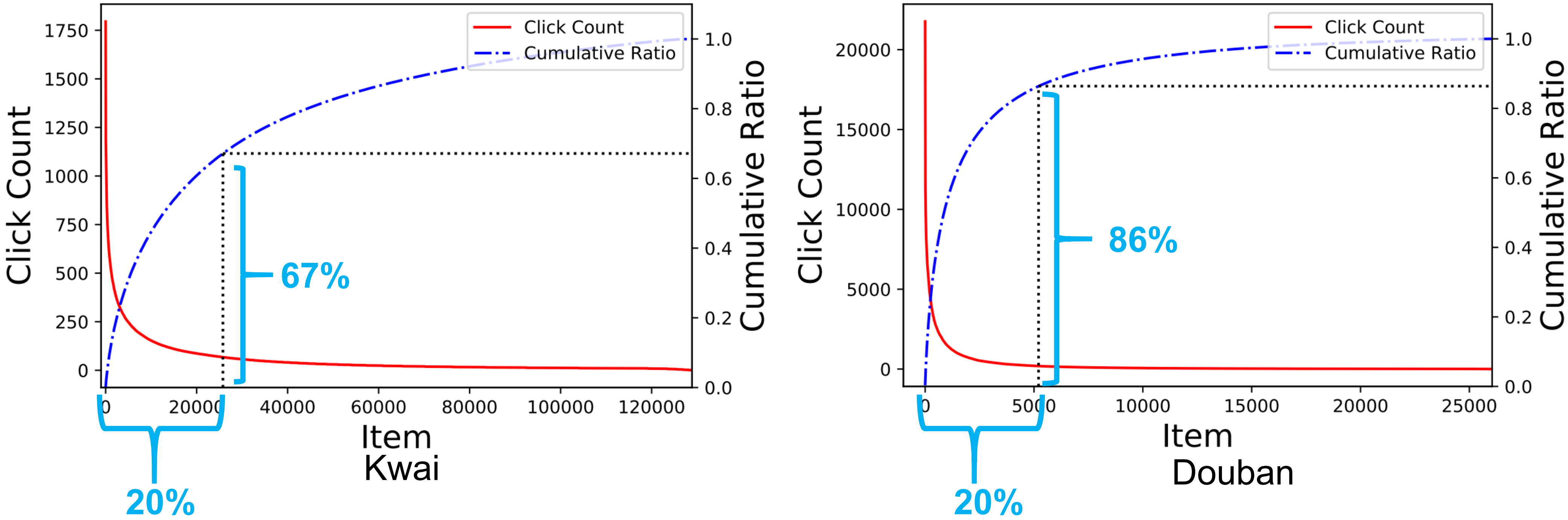
- Popularity bias refers to the problem where the recommendation algorithm **favors a few popular items** while not giving deserved attention to the majority of other items.

- Popularity bias is a well-known phenomenon in recommender systems where popular items are recommended even more frequently than their popularity would warrant, **amplifying** long-tail effects already present in many recommendation domains.

[1] Abdollahpouri, Himan. *Popularity Bias in Recommendation: A Multi-stakeholder Perspective*. Diss. University of Colorado, 2020.

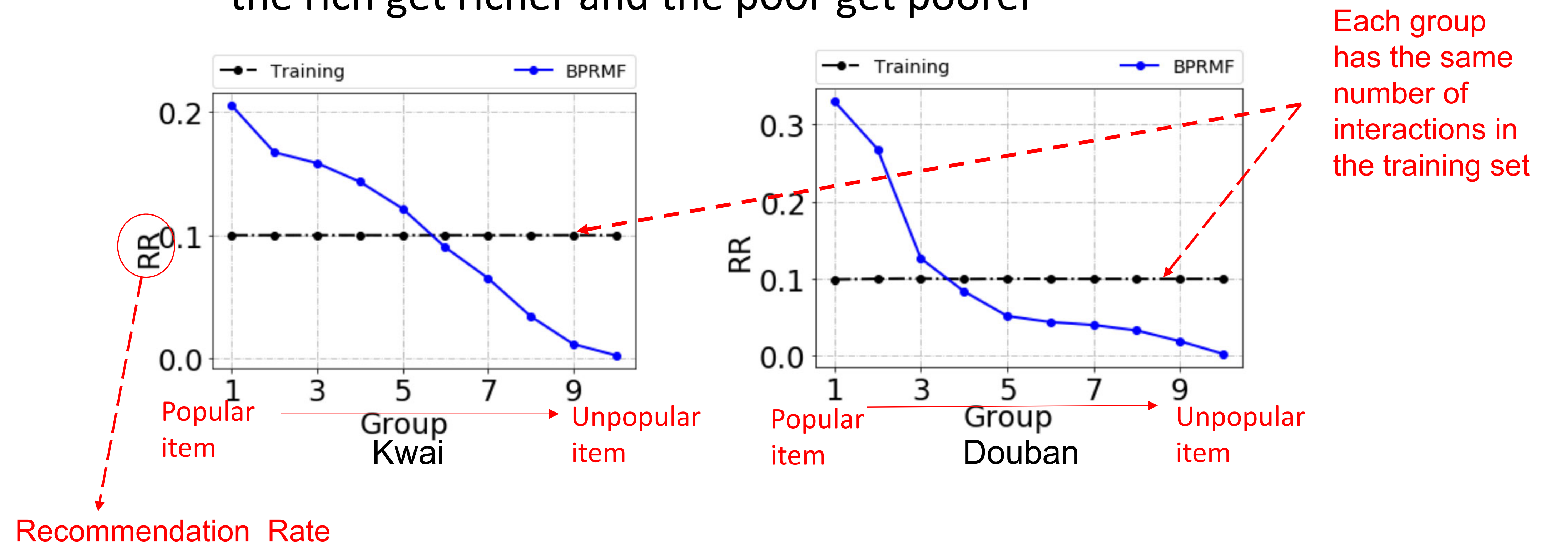
Source of Popularity Bias

- The Underlying Data
  - Few popular items which take up the majority of rating interactions while the majority of the items receive small attention from the users.



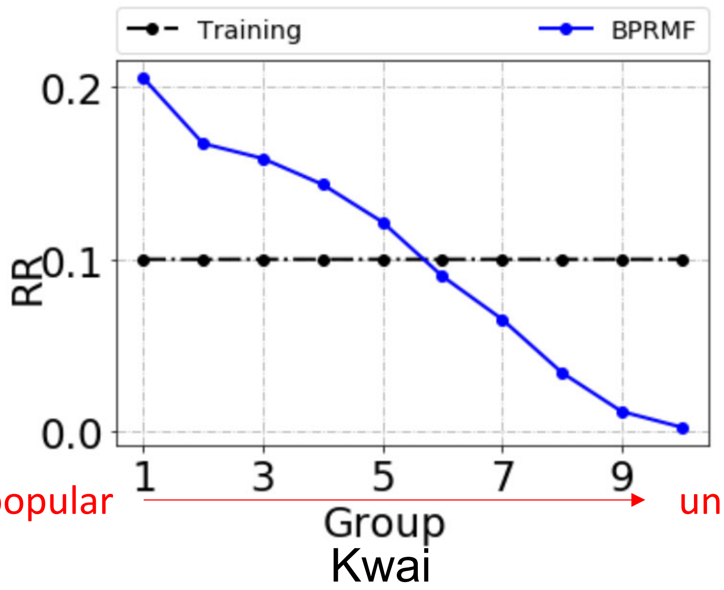
Source of Popularity Bias

- ➢ Algorithmic Bias
  - ☐ Not only inherit bias from data, but also amplify the bias.
  - —— the rich get richer and the poor get poorer



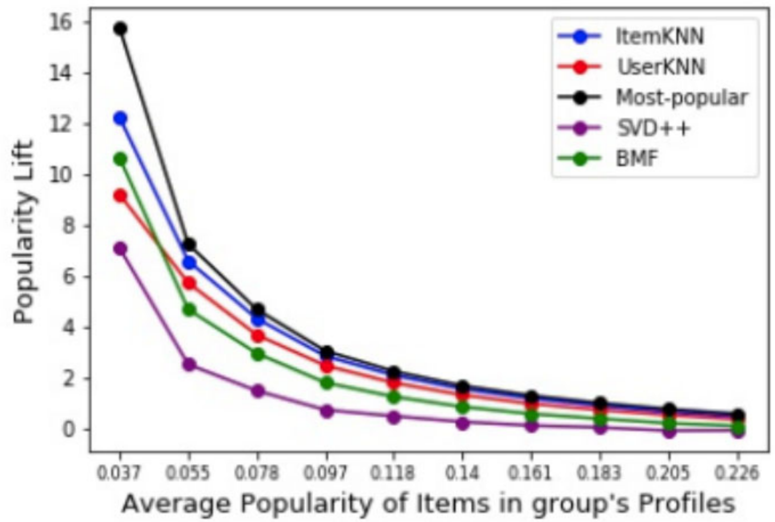
Impacts of Popularity Bias

➤ Item-side



popular → unpopular

User-side [1]



(a) MovieLens

→ Like popular items

Matthew effect; Amplified interests for popular items; Unfairness for both users and items

[1]. Abdollahpouri, Himan, et al. "The Connection Between Popularity Bias, Calibration, and Fairness in Recommendation." Fourteenth ACM Conference on Recommender Systems. 2020.

Methods for Popularity Bias

- **Ranking Adjustment --- balance the recommendation lists**
  - Regularization
  - Re-ranking
- Causal Embedding --- utilize causal-specific data
  - Disentanglement
- Causal Inference --- control the causal-effect of popularity
  - Inverse Propensity Score
  - Intervention
  - Counterfactual

Ranking Adjustment

➢ Regularization

Key: push the model towards 'balanced' recommendation lists by regularization

$$\min_{\{P,Q\}} \boxed{L_{acc}(P,Q)} + \boxed{\lambda L_{pop\_reg}(P,Q)}$$

Recommendation Loss

Regularization term for adjusting recommendation list

- L_{pop_reg}
  - ✓ **Fairness-aware** [1] : $tr(Q^{\top} L_D Q) \Rightarrow \min_{L_u} \frac{1}{N(N-1)} \sum_{ij \in R_u} d_{ij}$

    where R_u is recommendation list, and $D_{ij} = d_{ij} = \begin{cases} 0, & pop(i) \neq pop(j) \\ 1, & pop(i) = pop(j) \end{cases}$
  - ✓ **Decorrelation** [2] : $PCC\left(\hat{R}, pop(I)\right)^2$

    where $\hat{R} = P^{\top} Q$, pop(I) is the popularity of I,

    and PCC is Pearson Correlation Coefficient

[1]. Abdollahpouri, Himan et.al. "Controlling popularity bias in learning-to-rank recommendation." In RecSys 2017.
[2]. Ziwei zhu et.al. "Popularity-Opportunity Bias in Collaborative Filtering." In WSDM 2021.

Ranking Adjustment

➢ Re-ranking

Key: Modify the ranking score to adjust the ranking list

$$argmax_i \; \underbrace{\boxed{\hat{R}_{int}(u,i)}}_{\text{model score}} + \underbrace{\boxed{\lambda \, \hat{R}_{pop}(u,i)}}_{\text{adjusting score}}$$

- ☐ $\hat{R}_{pop}$
  - ✓ **Popularity Compensation** [1] : $\mathrm{C_{u,i}} * \frac{n_u}{\mathrm{m_u}}$

    Where $\mathrm{C_{u,i}} = \frac{1}{pop(i)}(\hat{R}_{int}(u,i)\beta + 1 - \beta)$, $\frac{n_u}{\mathrm{m_u}}$ is the re-scaling coefficient
  - ✓ **List smoothing [2]** : $\sum_{c\in\{F,F'\}} P(c|u)p(i|c)\prod_{j\in S}(1 - \mathrm{P}(j|c,S))$

    F, F': popular or unpopular $P(c|u)$: user interests for the popular (unpopular)

    $p(i|c)$: category of item i $\prod_{j\in S}(1 - \mathrm{P}(j|c,S))$: list state regarding popularity

[1] Ziwei zhu et.al. “Popularity-Opportunity Bias in Collaborative Filtering.” In WSDM 2021.
[2] Abdollahpouri et.al. "Managing popularity bias in recommender systems with personalized re-ranking.“ In FLAIRS 2019.

Methods for Popularity Bias

- ☐ Ranking Adjustment --- balance the recommendation lists
  - Regularization
  - Re-ranking
- ☐ **Causal Embedding --- utilize causal-specific data**
  - Disentanglement
- ☐ Causal Inference --- control the causal-effect of popularity
  - Inverse Propensity Score
  - Intervention
  - Counterfactual

Causal Embedding

➢ Bias-free uniform data

Key: utilizing causal-specific data to guide model learning [1]

- Even data(CausalE):

On even data $\qquad$ On biased data

$$\min_{\mathcal{W}_c,\mathcal{W}_t} \underbrace{\frac{1}{|S_c|}\sum_{(i,j)\in S_c}\ell\left(y_{ij},\hat{y}_{ij}^c\right)}_{\text{On even data}} + \underbrace{\frac{1}{|S_t|}\sum_{(i,j)\in S_t}\ell\left(y_{ij},\hat{y}_{ij}^t\right)}_{\text{On biased data}} +$$

$$\lambda_c R\left(\mathcal{W}_c\right) + \lambda_t R\left(\mathcal{W}_t\right) + \underbrace{\lambda_{tc}^{CausE}\left\|\mathcal{W}_t - \mathcal{W}_c\right\|_F^2}_{\text{Guiding term}},$$

Guiding term

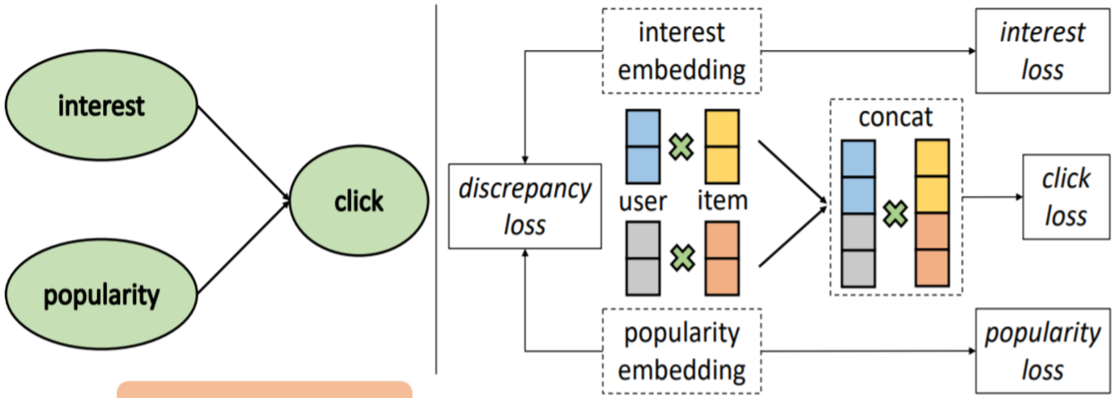
[1] Bonner, Stephen et.al. "Causal embeddings for recommendation." In RecSys 2018.

[2] Liu, Dugang, et al. "A general knowledge distillation framework for counterfactual recommendation via uniform data." In SIGIR 2020.

Causal Embedding

➢ Pairwise causal-specific data —— DICE

Key: **Disentangle** user interest and item popularity:



(a) causal graph

(b) causal embedding

$$L_{\text{interest}} = \sum_{(u,i,j)\in\mathcal{O}_1} \mathrm{BPR}(\langle u^{(\text{int})}, i^{(\text{int})}\rangle, \langle u^{(\text{int})}, j^{(\text{int})}\rangle).$$

$$L_{\text{click}} = \sum_{(u,i,j)\in\mathcal{O}} \mathrm{BPR}(\langle u^{(\text{int})} \| u^{(\text{pop})}, i^{(\text{int})} \| i^{(\text{pop})}\rangle, \langle u^{(\text{int})} \| u^{(\text{pop})}, j^{(\text{int})} \| j^{(\text{pop})}\rangle)$$

$$L^{(1)}_{\text{popularity}} = \sum_{(u,i,j)\in\mathcal{O}_1} -\mathrm{BPR}(\langle u^{(\text{pop})}, i^{(\text{pop})}\rangle, \langle u^{(\text{pop})}, j^{(\text{pop})}\rangle),$$

$$L^{(2)}_{\text{popularity}} = \sum_{(u,i,j)\in\mathcal{O}_2} \mathrm{BPR}(\langle u^{(\text{pop})}, i^{(\text{pop})}\rangle, \langle u^{(\text{pop})}, j^{(\text{pop})},\rangle),$$

$$L_{\text{popularity}} = L^{(1)}_{\text{popularity}} + L^{(2)}_{\text{popularity}}.$$

A click record reflects one or both aspects:
- the item's characteristics match the user's interest
- The item's popularity matches the user's conformity

•$\mathcal{O}$: whole training set of triplets (u, i, j): user, pos item, neg item

•$\mathcal{O}_1$: positive samples are less popular than negative samples

•$\mathcal{O}_2$: positive samples are more popular than negative samples

$$\mathcal{O} = \mathcal{O}_1 + \mathcal{O}_2$$

Yu Zheng et.al. "Disentangling user interest and conformity bias for recommendation with causal embedding." In www 2021.

Methods for Popularity Bias

- Ranking Adjustment --- balance the recommendation lists
  - Regularization
  - Re-ranking
- Causal Embedding --- utilize causal-specific data
  - Disentanglement
- Causal Inference --- control the causal-effect of popularity
  - Inverse Propensity Score
  - Intervention
  - Counterfactual

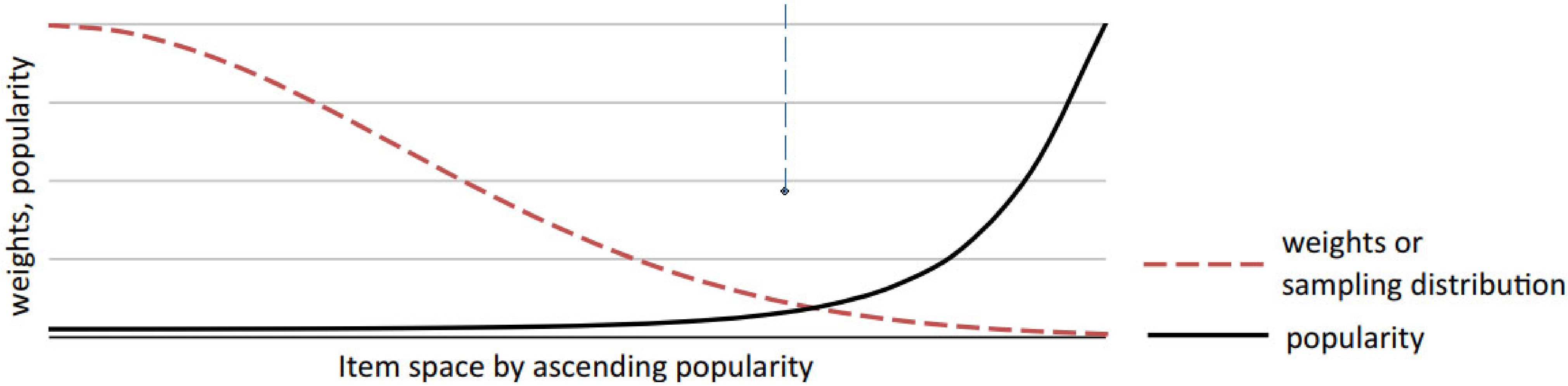
Causal Inference

- Inverse Propensity Scoring (IPS)[1,2]

Key: adjust the distribution of training data

$$Loss = \frac{1}{N}\sum \boxed{\frac{1}{ps(i)}} \delta(u,i)$$

Impose lower weights on popular items, and boost unpopular items



[1] Jannach, Dietmar, et al. "What recommenders recommend: an analysis of recommendation biases and possible countermeasures." User Modeling and User-Adapted Interaction 25.5 (2015): 427-491.

[2] Schnabel, Tobias, et al. "Recommendations as treatments: Debiasing learning and evaluation." *international conference on machine learning*. PMLR, 2016.

Causal Inference

➢ Basic Concepts in Causal Theory [1]

- **Causal Graph:**

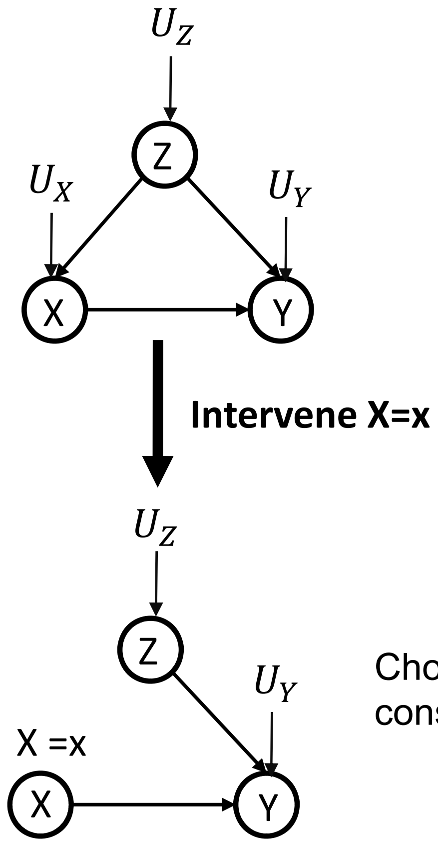
  Graphical models used to encode assumptions about the data-generating process.

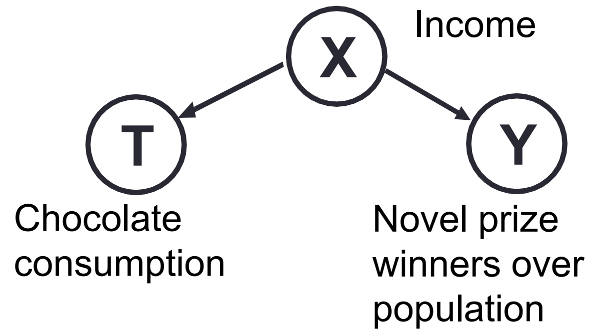
- **Intervention on X [term: do(X=x)]**

  Study specific causal relationships between X and the target variable.

  Randomized controlled trial.

  In graph: Cut off the paths that point into X





[1]. Pearl, Judea, Madelyn Glymour, and Nicholas P. Jewell. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016.

Causal Inference

➢ Basic Concepts in Causal Theory [1]

☐ Causal Effect:

P(Y| do(X=x)) – P(Y | do(X=x_{ref}))

measures the expected increase in Y as the treatment changes from $X = x$ to X=x_{ref}

General causal effect: P(Y| do(X=x))

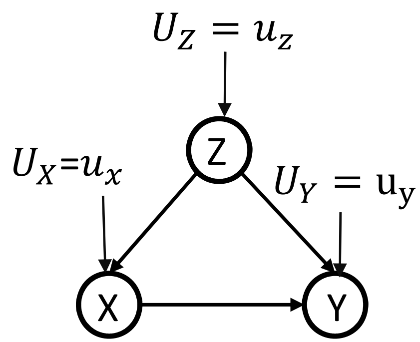
Others: NIE, NDE, TIE …

☐ Counterfactual

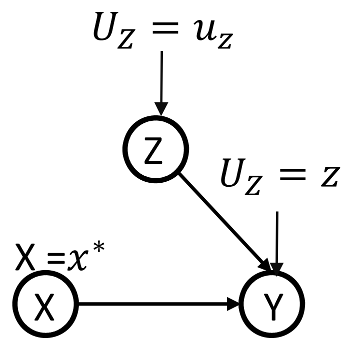
Imagine a world that does not really existed, given existed information.

Observed Y=y_1, assume the X is x^*, what will the Y is?

- **Abduction**: Based on Y=y_1, inference $U_Y = u_y, U_z = u_z$
- **Action**: Let X=x^*
- **Prediction**: $Z = f_z(u_z),\ X = x^*, Y = f_Y(f_z(u_z), x^*, u_y)$



Factual



Counterfactual

[1]. Pearl, Judea, Madelyn Glymour, and Nicholas P. Jewell. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016.

Causal Inference

➢ Counterfactual Inference —— MACR

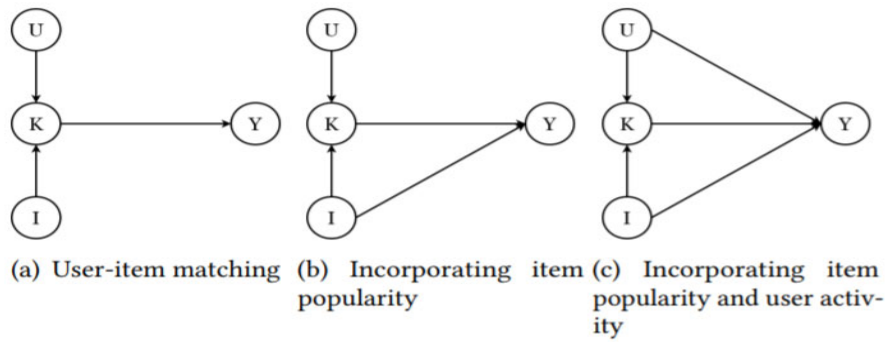
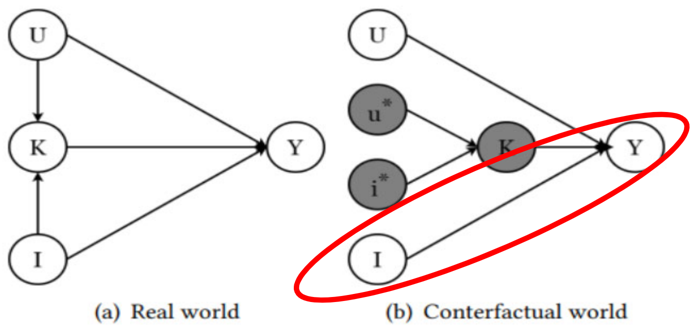


Figure 2: Causal graph for (a) user-item matching; (b) incorporating item popularity; and (c) incorporating user activity. I: item. U: user. K: matching features between user and item. Y: ranking score (e.g., the probability of interaction).

- A causal view of the popularity bias in recommendation.
  - The direct edge from I to R represents popularity bias.
  - The direct edge from U to R represents to what extent the user is sensitive to popularity.



- Counterfactual inference:

$$\underbrace{\hat{y}_k * \sigma(\hat{y}_i) * \sigma(\hat{y}_u)}_{\text{Factual prediction}} - \underbrace{c * \sigma(\hat{y}_i) * \sigma(\hat{y}_u)}_{\text{Counterfactual prediction}},$$

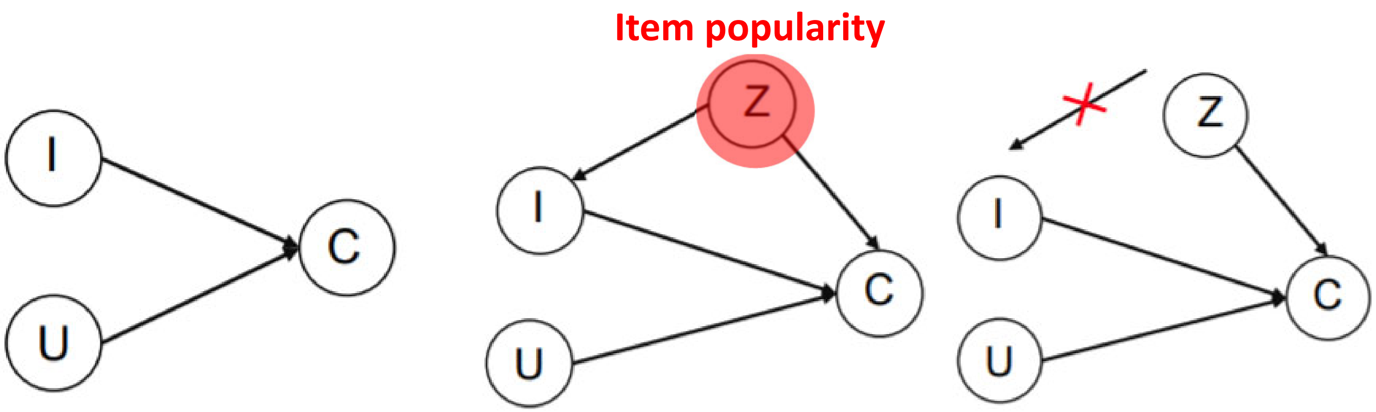
Wei, Tianxin, et al. "Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System." SIGKDD, 2021

Causal Inference

- De-confounding —— Popularity De-confounding(PD) and Adjusting (PDA)

Key: item popularity is a confounder, both bad and good effect of popularity exist.

Leverage popularity bias instead of blindly removing.



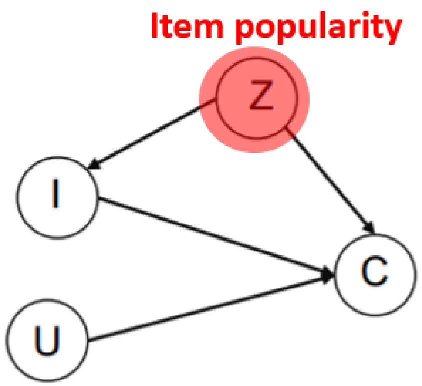
(a) Causal graph of traditional methods. (b) Causal graph that considers item popularity. (c) We cut off $Z \rightarrow I$ for model training

We estimate the user-item matching as $P(C|do(U, I))$ based on figure (c)

“Causal Intervention for Leveraging Popularity Bias in Recommendation.” SIGIR, 2021

Causal Inference

➢ PD --- Popularity De-confounding



Causality:
$\boldsymbol{P(C|do(U,I)) = \sum_Z P(C|U,I,Z)P(Z)}$

vs

Correlation:
$P(C|U,I) = \sum_Z P(C|U,I,Z)P(Z|U,I)$

◻ De-confounding --- estimate $\boldsymbol{P(C|do(U,I))}$:

- **Step 1**. estimate P(C|U,I,Z)
  - $P_\Theta(c=1|u,i,m_i^t) = ELU'(f_\Theta(u,i)) \times (m_i^t)^\gamma$
  - m_i^t the popularity of item i in timestamp t
  - $f_\Theta(u,i)$: user-item matching, such as MF
  - Learning this component from data
- **Step 2**. computing $P(\mathrm{C}|\mathrm{do}(\mathrm{U},\mathrm{I}))$
  - $\sum_Z P(C|U,I,Z)P(Z) \propto ELU'(f_\Theta(u,i))$
  - ranking with this term

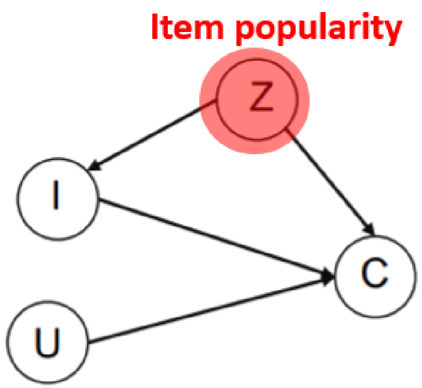
✓ In pursuit of real interests instead of even state! Higher popularity because of better quality.

“Causal Intervention for Leveraging Popularity Bias in Recommendation.” SIGIR, 2021

Causal Inference

➢ PDA --- Popularity De-confounding and Adjusting

- We have estimated $P(C|do(U,I))$, which does not chase the even state but the real interests.
- Is it enough?
  - No… In some time, we need inject some desired popularity.
  - Such as we can recommend more item that will be popular if we can know the trends of popularity.



Introducing popularity bias by intervention:

$P\left(C \middle| do(U,I), do(Z=\tilde{Z})\right)$ = P(C|U,I, $\tilde{Z}$)

P(C|U,I, $\tilde{Z}$) = $ELU'\left(f_\Theta(u,i)\right) \times \left(\tilde{Z}_i\right)^{\gamma}$

$\tilde{Z}$: predicted by the trends of item popularity.

“Causal Intervention for Leveraging Popularity Bias in Recommendation.” SIGIR, 2021

Causal Inference

➢ Experimental Setting

- **Datasets:**

Dataset	#User	#Item	#Interaction	#Sparsity	#type
Kwai	37,663	128,879	7,658,510	0.158%	Click
Douban	47,890	26,047	7,174,218	0.575%	Review
Tencent	80,339	27,070	1,816,046	0.084%	Like

- **Data Splitting:**

  Temporal splitting --- split each into 10 time stages according to timestamp.
  0-8th stages: training, 9th stage: validation & testing.

- **Evaluation Setting:**

  PD: directly test
  PDA: Most recent stages can be utilized to predict future popularity.

- **Baselines:**

  PD: MostPop, BPRMF, xQuad(2019FLAIRS), BPR-PC(2021WSDM), DICE(2021WWW)
  PDA: MostRecent(2020SIGIR), BPRMF(t)-pop(2017RecTemp@ RecSys), BPRMF-A, DICE-A

“Causal Intervention for Leveraging Popularity Bias in Recommendation.” SIGIR, 2021

Causal Inference

➢ Results for PD

Dataset	Methods	Top 20 Recall	Top 20 Precision	Top 20 HR	Top 20 NDCG	Top 20 RI	Top 50 Recall	Top 50 Precision	Top 50 HR	Top 50 NDCG	Top 50 RI
Kwai	MostPop	0.0014	0.0019	0.0341	0.0030	632.4%	0.0040	0.0021	0.0802	0.0036	480.9%
	BPRMF	0.0054	0.0057	0.0943	0.0067	146.3%	0.0125	0.0053	0.1866	0.0089	121.0%
	xQuad	0.0054	0.0057	0.0948	0.0068	145.0%	0.0125	0.0053	0.1867	0.0090	120.3%
	BPR-PC	0.0070	0.0056	0.0992	0.0072	125.0%	0.0137	0.0046	0.1813	0.0092	123.7%
	DICE	0.0053	0.0056	0.0957	0.0067	147.8%	0.0130	0.0052	0.1872	0.0090	119.0%
	PD	**0.0143**	**0.0138**	**0.2018**	**0.0177**	-	**0.0293**	**0.0118**	**0.3397**	**0.0218**	-
Douban	MostPop	0.0218	0.0297	0.2373	0.0349	75.4%	0.0490	0.0256	0.3737	0.0406	55.9%
	BPRMF	0.0274	0.0336	0.2888	0.0405	47.0%	0.0581	0.0291	0.4280	0.0475	34.3%
	xQuad	0.0274	0.0336	0.2895	0.0391	48.3%	0.0581	0.0291	0.4281	0.0473	34.4%
	BPR-PC	0.0282	0.0307	0.2863	0.0381	51.6%	0.0582	0.0271	0.4260	0.0457	38.0%
	DICE	0.0273	0.0336	0.2845	0.0421	46.2%	0.0513	0.0273	0.4000	0.0460	44.5%
	PD	**0.0453**	**0.0454**	**0.3970**	**0.0607**	-	**0.0843**	**0.0362**	**0.5271**	**0.0686**	-
Tencent	MostPop	0.0145	0.0043	0.0684	0.0093	340.8%	0.0282	0.0035	0.1181	0.0135	345.8%
	BPRMF	0.0553	0.0153	0.2005	0.0328	27.1%	0.1130	0.0129	0.3303	0.0497	25.3%
	xQuad	0.0552	0.0153	0.2007	0.0326	27.3%	0.1130	0.0129	0.3302	0.0497	25.3%
	BPR-PC	0.0556	0.0153	0.2018	0.0331	26.5%	0.1141	0.0128	0.3322	0.0500	24.9%
	DICE	0.0516	0.0149	0.1948	0.0312	32.8%	0.1010	0.0132	0.3312	0.0486	29.0%
	PD	**0.0715**	**0.0195**	**0.2421**	**0.0429**	-	**0.1436**	**0.0165**	**0.3875**	**0.0641**	-

The power of de-confounded estimation !!

"Causal Intervention for Leveraging Popularity Bias in Recommendation." SIGIR, 2021

Causal Inference

➢ PD —— Recommendation Analysis.

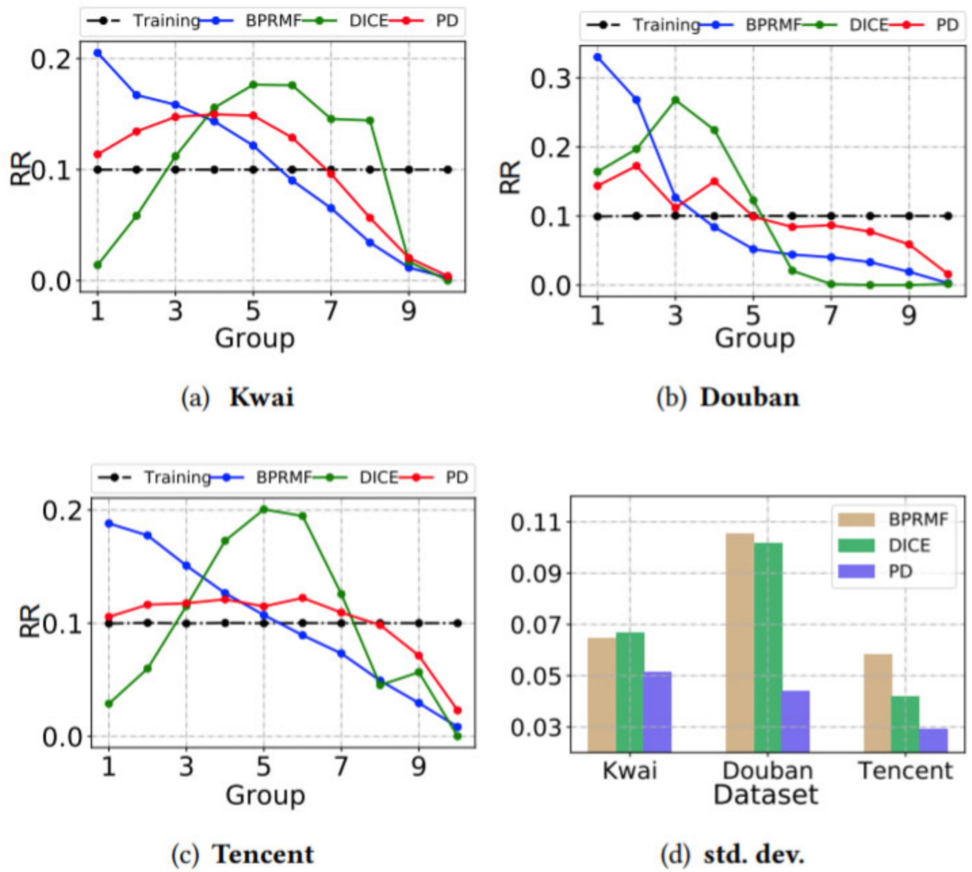


Figure 4: Recommendation rate(RR) over item groups.

- Less amplification for most popular groups compared with BPRMF
- Do not **over-suppress** the most popular groups compared with DICE
- More flat lines and standard deviations over different groups
  --- **relative fair recommendation opportunities** for different group (refer to training set)
- Better performance
  --- remove bad effect but keep good effect of popularity bias

Causal Inference

➢ Results for PDA

Table 2: Top-K recommendation performance with popularity adjusting on Kwai, Douban, and Tencent Datasets.

Datasets		Kwai				Douban				Tencent			
Methods		top 20		top 50		top 20		top 50		top 20		top 50	
		Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG
MostRecent		0.0074	0.0096	0.0139	0.011	0.0398	0.0582	0.0711	0.0615	0.0360	0.0222	0.0849	0.0359
BPRMF(t)-pop		0.0188	0.0241	0.0372	0.0286	0.0495	0.0682	0.0929	0.0760	0.1150	0.0726	0.2082	0.1001
BPRMF-A	(a)	0.0191	0.0249	0.0372	0.0292	0.0482	0.0666	0.0898	0.0744	0.1021	0.0676	0.1805	0.0905
	(b)	0.0201	0.0265	0.0387	0.0306	0.0486	0.0667	0.0901	0.0746	0.1072	0.0719	0.1886	0.0953
DICE-A	(a)	0.0242	0.0315	0.0454	0.0363	0.0494	0.0681	0.0890	0.0736	0.1227	0.0807	0.2161	0.1081
	(b)	0.0245	0.0323	0.0462	0.0370	0.0494	0.0680	0.0882	0.0734	0.1249	0.0839	0.2209	0.1116
PDA	(a)	0.0279	0.0352	0.0531	0.0413	0.0564	**0.0746**	**0.1066**	**0.0845**	0.1357	0.0873	0.2378	0.1173
	(b)	**0.0288**	**0.3364**	**0.054**	**0.0429**	**0.0565**	0.0745	**0.1066**	0.0843	**0.1398**	**0.0912**	**0.2418**	**0.1210**

- Introducing desired popularity bias can improve the recommendation performance.
- Our method achieves the best performance.

"Causal Intervention for Leveraging Popularity Bias in Recommendation." SIGIR'21

Conclusion & Future Work

- Conclusion
  - Heuristic methods -- Not best
  - Uniform/unbiased data -- Hard to get these data
  - Causal perspective
    - IPS -- Hard to estimate Propensity Scores
    - Counterfactual & Intervention -- Extra assumption of causal graph
  - Eliminate the bad effect of bias, leverage the good effect of bias.
- Potential directions
  - Considering popularity bias at finer-grain [1,2].
  - Considering multiple behaviors.
  - Popularity bias with features of users and items.
  - Accurate estimation of causal effect.

[1] Zhao, Zihao, et al. "Popularity Bias Is Not Always Evil: Disentangling Benign and Harmful Bias for Recommendation." Under Review.
[2] Wang, Wenjie, et al. "Deconfounded Recommendation for Alleviating Bias Amplification." SIGKDD, 2021.

Thanks!

• Tutorial Outline

- ❑Biases in Data (Jiawei Chen, 60 min)
  - ❑Definition of data biases
  - ❑Categories: Selection bias, Conformity bias, Exposure bias and Position bias
  - ❑Recent solutions for data biases
- ❑ Bias Amplification in Loop and its Solutions (Jiawei Chen, 10 min)
- ❑ Biases in Results
  - ❑Popularity bias: definition, characteristic and solutions (Fuli Feng, 40 min)
  - ❑Unfairness: definition, characteristic and solutions (Xiang Wang, 50 min)

slides will be available at: https://github.com/jiawei-chen/RecDebiasing
A literature survey based on this tutorial is available at: https://arxiv.org/pdf/2010.03240.pdf
.

• Sensitive Attributes in Fairness

User Data: U

Item Data: I

Interaction Data: Y

Data

$g: (U, I, Y) \rightarrow Z_U, Z_I$

Representation Learning

$f: Z_U, Z_I \rightarrow Y$

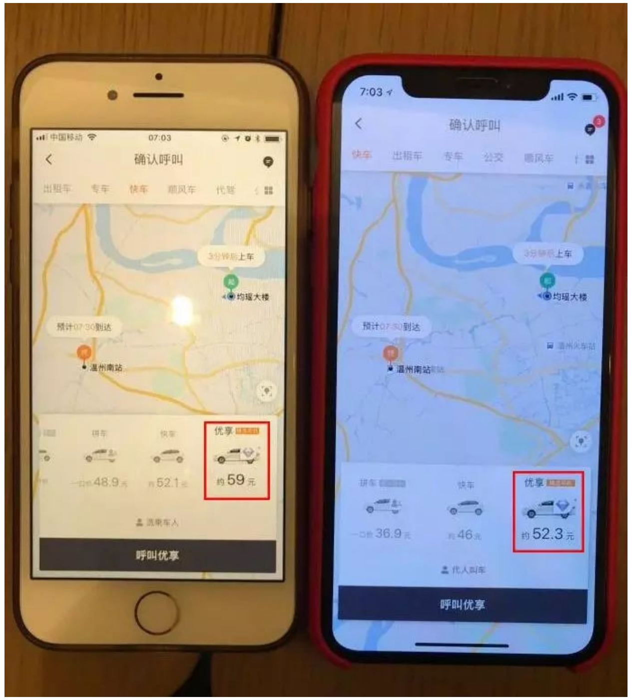
Interaction Modeling

User Side
- Age
- Gender
- Race
- Color
- Religion
- Disability status
- Marriage status
- … …

Sensitive Attribute (protected variable)

User experience differs:
- in terms of **service effectiveness** (results' relevance, user satisfaction)
- **resulting outcomes** (exposure to lower-paying job offers)
- **participation costs** (privacy risks)

Discrimination



• Unfairness Leads to Discrimination

Individual Fairness

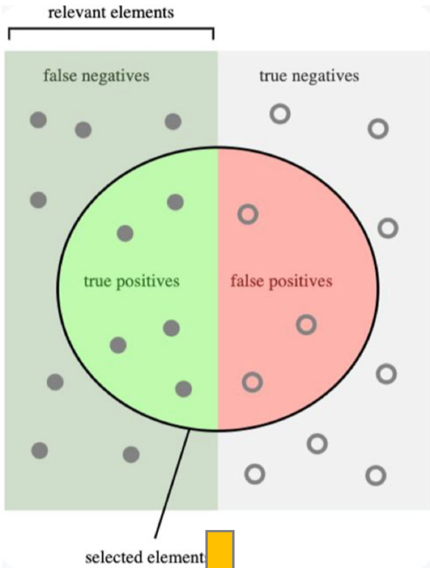
"Similar individuals treated similarly"

Basketball

Ping-pong ball

Group Fairness

"Similar Classifier Statistics Across Groups"



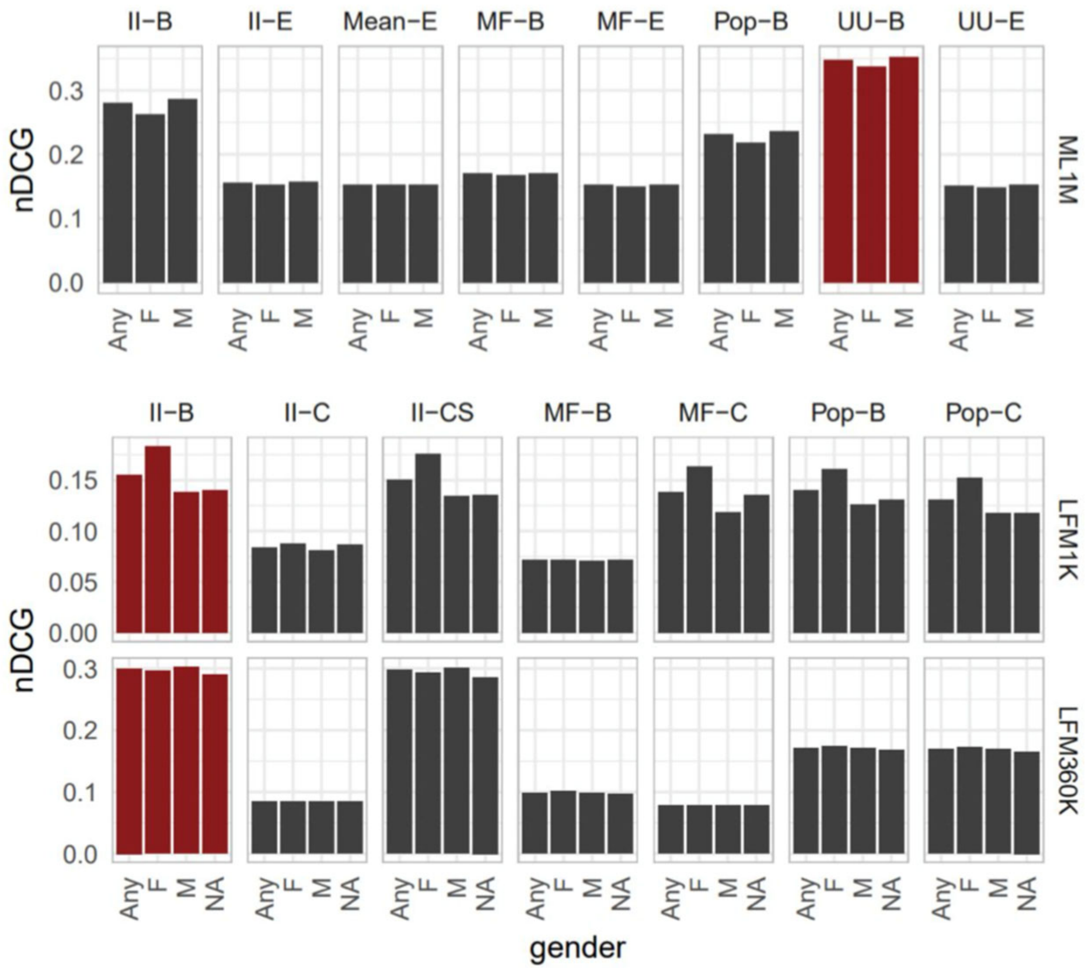
Individual Discrimination

A model gives unfairly different predictions to similar individuals

Group Discrimination

A model systematically treats individuals who belong to different groups unfairly

• Case 1 in Recommendation



Motivation

Investigating whether **demographic groups obtain different utility** from recommender systems in LastFM and Movielens 1M datasets

Findings

- Movielens 1M & LastFM 1K have statistically-significant differences between **gender groups**
- LastFM 360K has significant differences between **age brackets**

[Ekstrand et al., All The Cool Kids, How Do They Fit In?: Popularity and Demographic Biases in Recommender Evaluation and Effectiveness. FAT 2018: 172-186]

• Case 2 in Recommendation

User group	UserItemAvg	UserKNN	UserKNNAvg	NMF
LowMS	42.991***	49.813***	46.631***	**38.515*** **
MedMS	33.934	42.527	37.623	**30.555**
HighMS	40.727	46.036	43.284	**37.305**
All	38.599	45.678	41.927	**34.895**

Table 1. MAE results (the lower, the better) for four personalized recommendation algorithms and our three user groups. The worst (i.e., highest) results are always given for the LowMS user group (statistically significant according to a t-test with $p < .005$ as indicated by ***). Across the algorithms, the best (i.e., lowest) results are provided by NMF (indicated by bold numbers).

Motivation

Investigating **three user groups** from Last.fm based on how much their listening preferences deviate from the most popular music:

- low-mainstream users
- medium-mainstream users
- high-mainstream users

Findings

- Different user groups are treated differently
- **Low-mainstream user group** significantly receives the **worst** recommendations

[Dominik et al.: The Unfairness of Popularity Bias in Music Recommendation: A Reproducibility Study. ECIR 2020]

• Definitions of Fairness

Fairness through Unawareness

A model is fair if **any sensitive attribute is not explicitly used** in the decision-making process

Individual Fairness

a model is fair if it gives similar predictions to **similar individuals**

$$\hat{Y}(X(i), A(i)) \approx \hat{Y}(X(j), A(j)), if\ |X(i) - X(j)| \leq \varepsilon$$

Equal Opportunity

A model is fair if the groups have **equal true positive rates**

$$P(\hat{Y} = 1 | A = 0, Y = 1) = P(\hat{Y} = 1 | A = 1, Y = 1)$$

Equalized Odds

A model is fair if the groups have **equal rates for true positives and false positives**

$$P(\hat{Y} = 1 | A = 0) = P(\hat{Y} = 1 | A = 1)$$

Demographic Parity

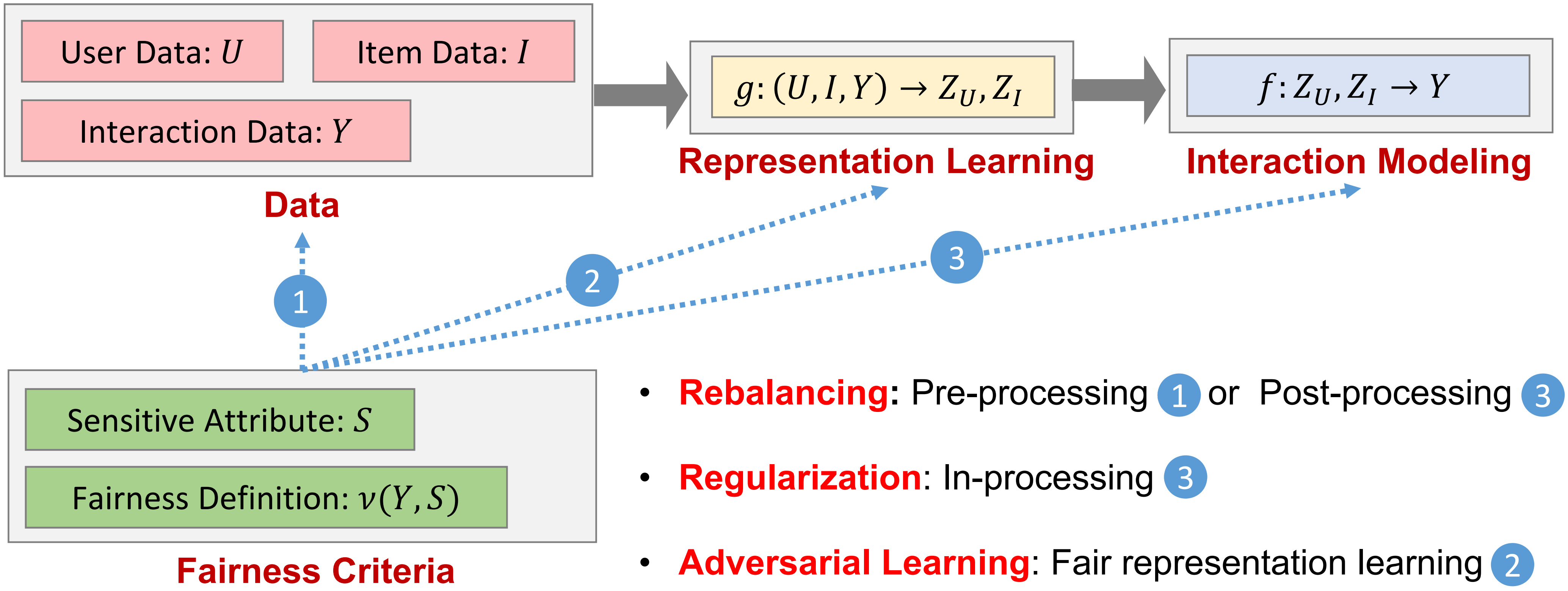
A model is fair if **the likelihood of a positive outcome** should be the same regardless of the group $P(\hat{Y} | A = 0) = P(\hat{Y} | A = 1)$

Counterfactual Fairness

A model is is fair towards an individual if it is the same **in both the actual world and a counterfactual world** where the individual belonged to a different demographic group

$$P(\hat{Y} | X = x, do(A) = 0) = P(\hat{Y} | X = x, do(A) = 1)$$

• Four Research Lines towards Fairness

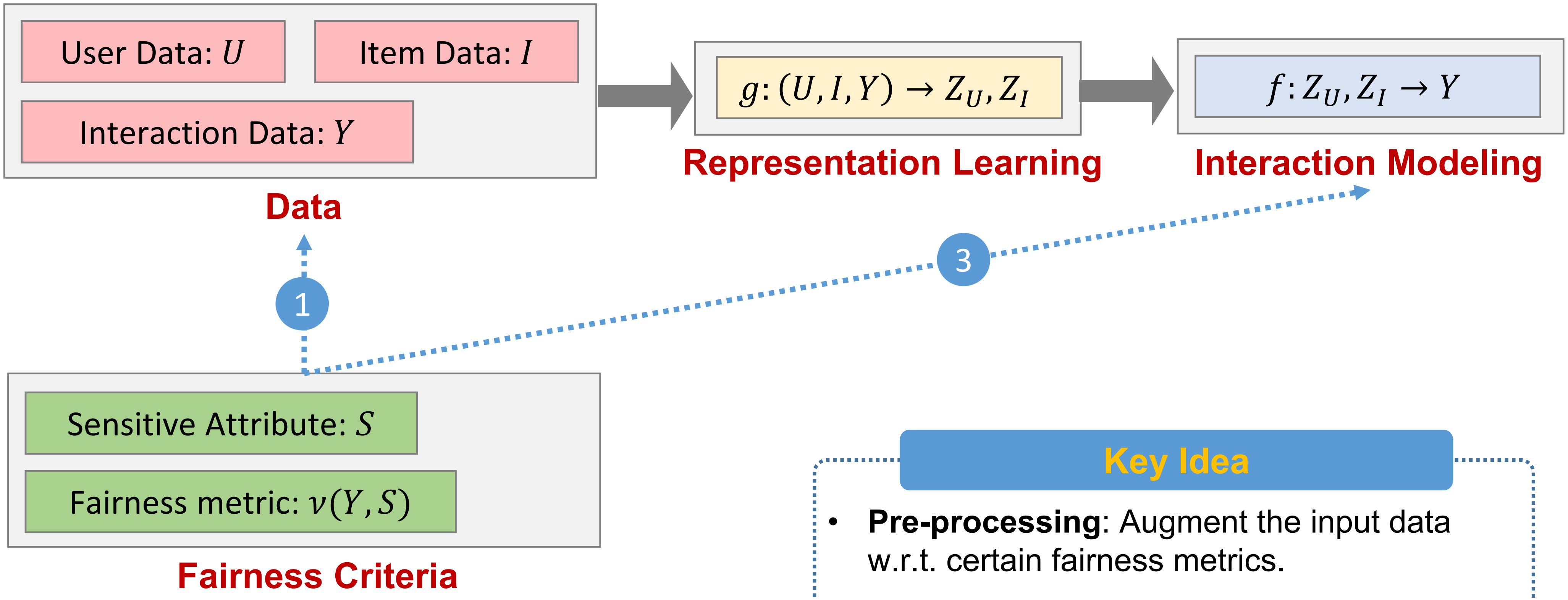


- **Rebalancing**: Pre-processing 1 or Post-processing 3
- **Regularization**: In-processing 3
- **Adversarial Learning**: Fair representation learning 2

• Four Research Lines towards Fairness

- ☐ Rebalancing
- ☐ Regularization
- ☐ Fair Representation Learning

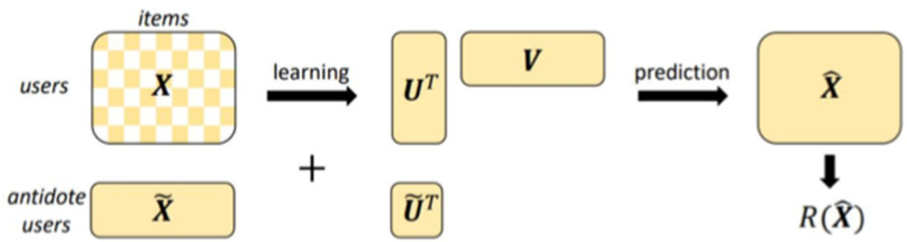
• Line 1: Rebalancing

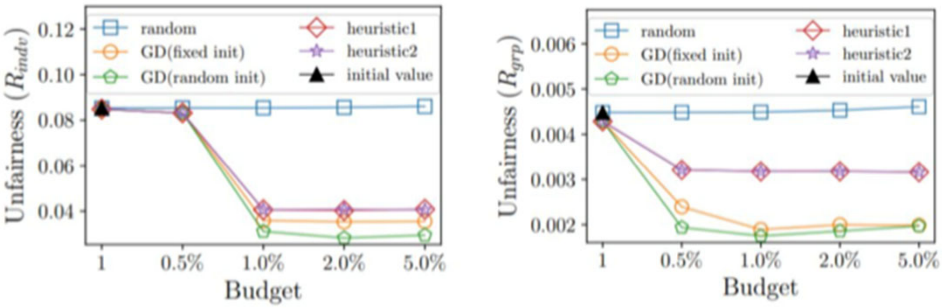


Key Idea

- **Pre-processing**: Augment the input data w.r.t. certain fairness metrics.
- **Post-processing**: Balance the results w.r.t. certain fairness metrics.

• **Example 1: Pre-processing** — Using Antidote Data





(a) Individual fairness **(b) Group fairness**

Figure 3: Improving fairness.

Idea

Augmenting the input with **additional "antidote" data** can improve the social desirability of recommendations

Algorithms

MF family of algorithms

Findings

- The small amounts of antidote data (typically on the order of 1% new users) can generate a dramatic improvement (on the order of 50%) in the polarization or the fairness of the system's recommendations

[Rastegarpanah et al.: Fighting Fire with Fire: Using Antidote Data to Improve Polarization and Fairness of Recommender Systems. WSDM 2019]

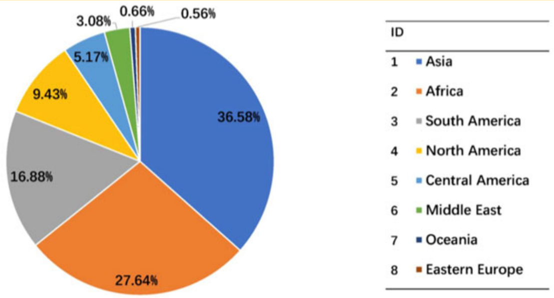
• Example 2: Post-processing — Fairness-Aware Re-ranking

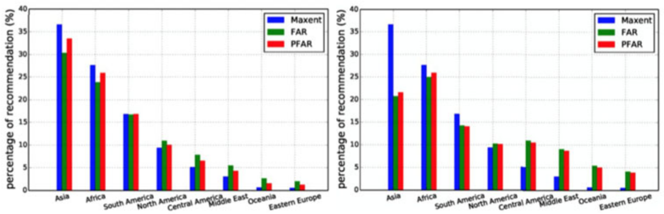
Personalization score determined by the base recommender

Importance of the group with attribute c

$$\max_{v\in R(u)} \underbrace{(1-\lambda)P(v|u)}_{\text{personalization}} + \underbrace{\lambda\tau_u \sum_c P(\mathcal{V}_c)\mathbb{1}_{\{v\in\mathcal{V}_c\}} \prod_{i\in S(u)} \mathbb{1}_{\{i\notin\mathcal{V}_c\}}}_{\text{personalized fairness}},$$

coverage of Vc for the current generated re-ranked list S(u)





(a) $\lambda = 0.1$, $\mathbf{nDCG_{FAR}} = 0.0962$, $\mathbf{nDCG_{PFAR}} = 0.0983$.

(b) $\lambda = 0.99$, $\mathbf{nDCG_{FAR}} = 0.0709$, $\mathbf{nDCG_{PFAR}} = 0.0756$.

Figure 4: Recommendation percentage of each region.

[Liu et al.: Personalized fairness-aware re-ranking for microlending. RecSys 2019)]

Idea

Combing a **personalization-induced term** & a **fairness-induced term** to promote the loans of currently uncovered borrower groups

Algorithms

RankSGD, UserKNN, WRMF, Maxent

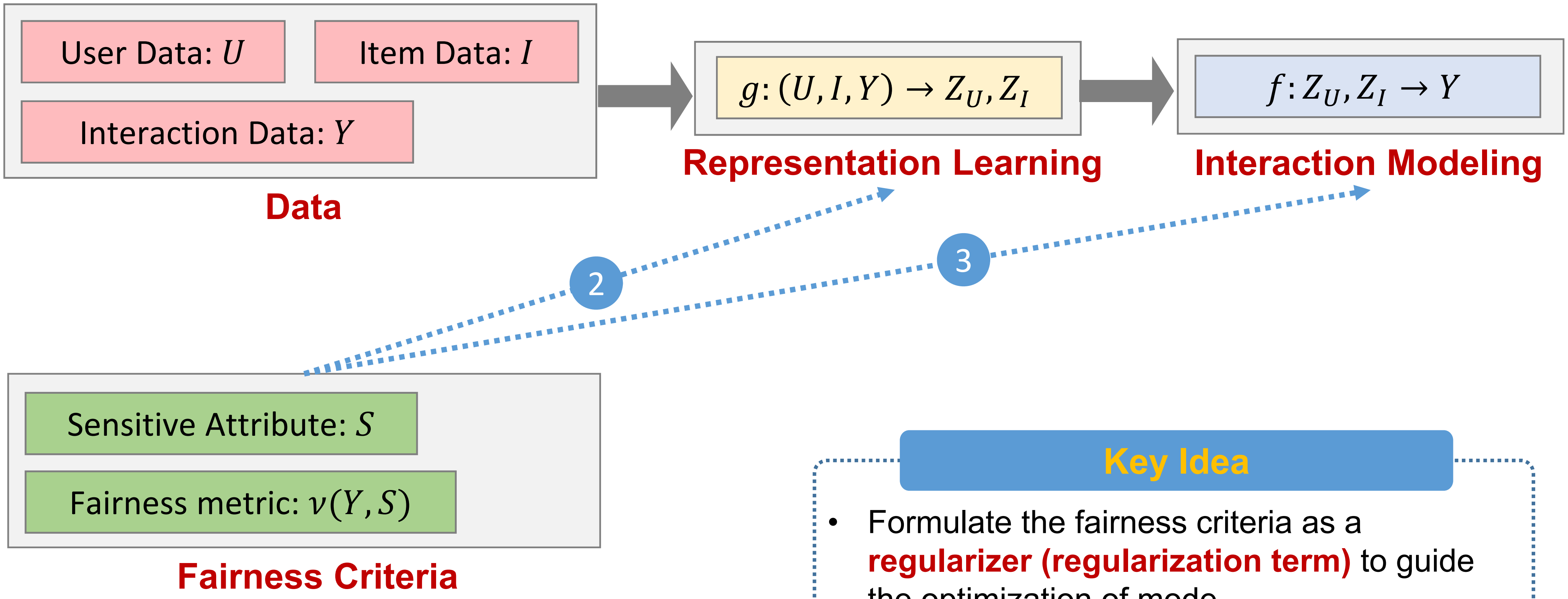
Findings

- A balance between the two terms
- **Recommendation accuracy** (nDCG) remains at a high level after the re-ranking
- **Recommendation fairness** is significantly improved → loans belonging to less-popular groups are promoted.

• Four Research Lines towards Fairness

- ☐ Rebalancing
- ☐ Regularization
- ☐ Fair Representation Learning

• Line 2: Regularization



Key Idea

- Formulate the fairness criteria as a **regularizer (regularization term)** to guide the optimization of mode

• **Example 1:** Learned Fair Representation (LFR)

Reconstruction loss between input data X and representations R

Prediction error in generating prediction Y from R

$$\min \mathcal{L} = \alpha C(X,R) + \beta D(R,A) + \gamma E(Y,R)$$

Regularization term that measures the dependence between R and sensitive attribute A

Fairness criteria (e.g., demographic parity)

$$D(R,A) = |\mathbb{E}_R P(R|A=1) - \mathbb{E}_R P(R|A=0)|$$

Distance of representation R and the centroid representation of the group where A = 1

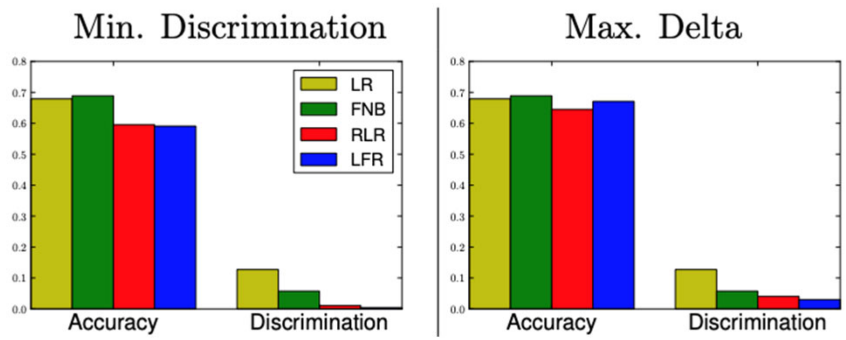
Idea

Representation Learning

- **encode insensitive attributes** of data
- **remove any information about sensitive attributes** w.r.t. the protected subgroup

Findings

- pushing the discrimination to very low values
- while maintaining fairly high accuracy



[Zemel et al.: Learning Fair Representations. ICML 2013]

• Example 2: Neutrality-Enhanced Recommendation

Loss of predicting ratings (e.g., squared error)

L2 regularization on model parameters

$$\mathcal{L}(\mathcal{D}) = \sum_{(x_i, y_i, s_i, v_i) \in \mathcal{D}} (s_i - \hat{s}(x_i, y_i, v_i))^2 + \eta\, I(\hat{s}; v) + \lambda\, R$$

Neutrality function to quantify the degree of the information neutrality from a viewpoint variable

Independence between the predictions & sensitive attributes → negative mutual information

$$-I(\hat{s}; v) = \sum_{v \in \{0,1\}} \int \Pr[\hat{s}, v] \log \frac{\Pr[\hat{s}|v]}{\Pr[\hat{s}]} d\hat{s}$$

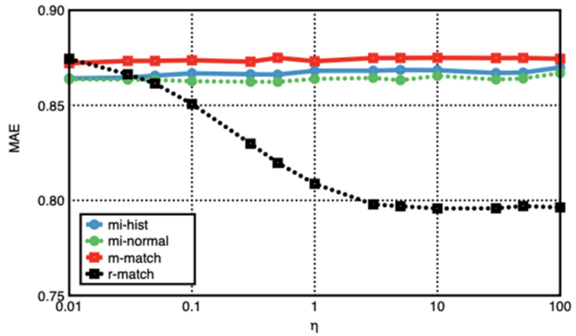
Idea

Regularization term

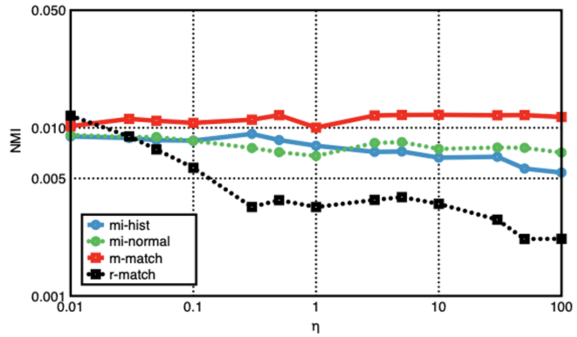
- **Negative mutual information** between sensitive attribute A and prediction Y

Findings

- enhances the independence toward the specified sensitive attribute



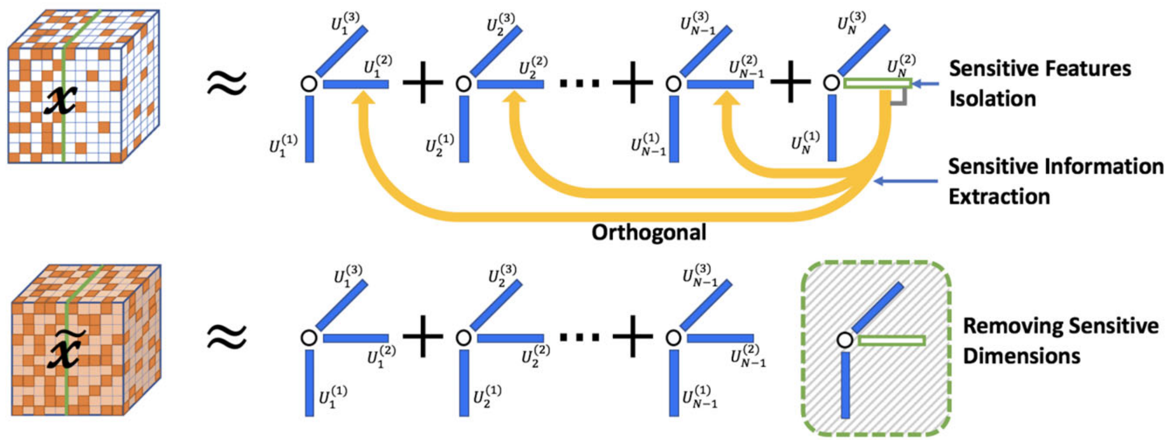
(c) Prediction error (MAE) for Gender data



(d) Degree of neutrality (NMI) for Gender data

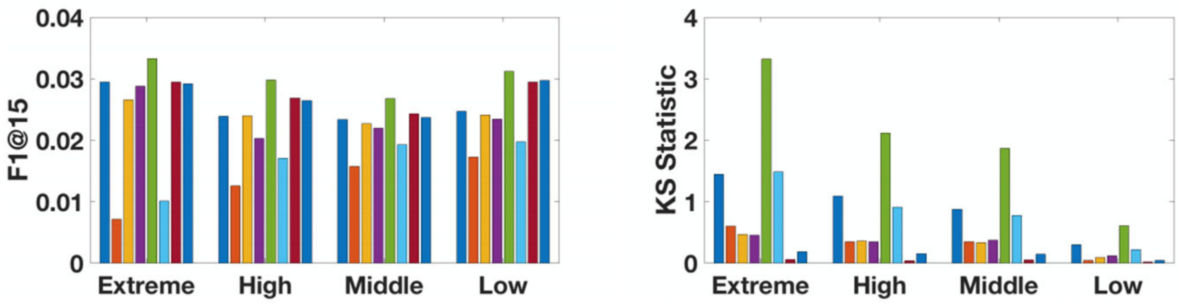
[Kamishima et al.: Neutrality-Enhanced Recommendation. RecSys 2013]

• **Example 3:** Fairness-Aware Tensor-based Rec (FATR)



Idea

- Use **sensitive latent factor matrix** to isolate sensitive features
- Use a regularizer to **extract sensitive information which taints other factors**.



(a) Recommendation quality: F1@15. (b) Fairness: KS Statistic.

Findings

- Eliminate sensitive information & provides fair recommendation with respect to the sensitive attribute.
- Maintain recommendation quality

[Zhu et al.: Fairness-Aware Tensor-Based Recommendation. CIKM 2018]

• Some tradeoffs when comparing these fairness approaches

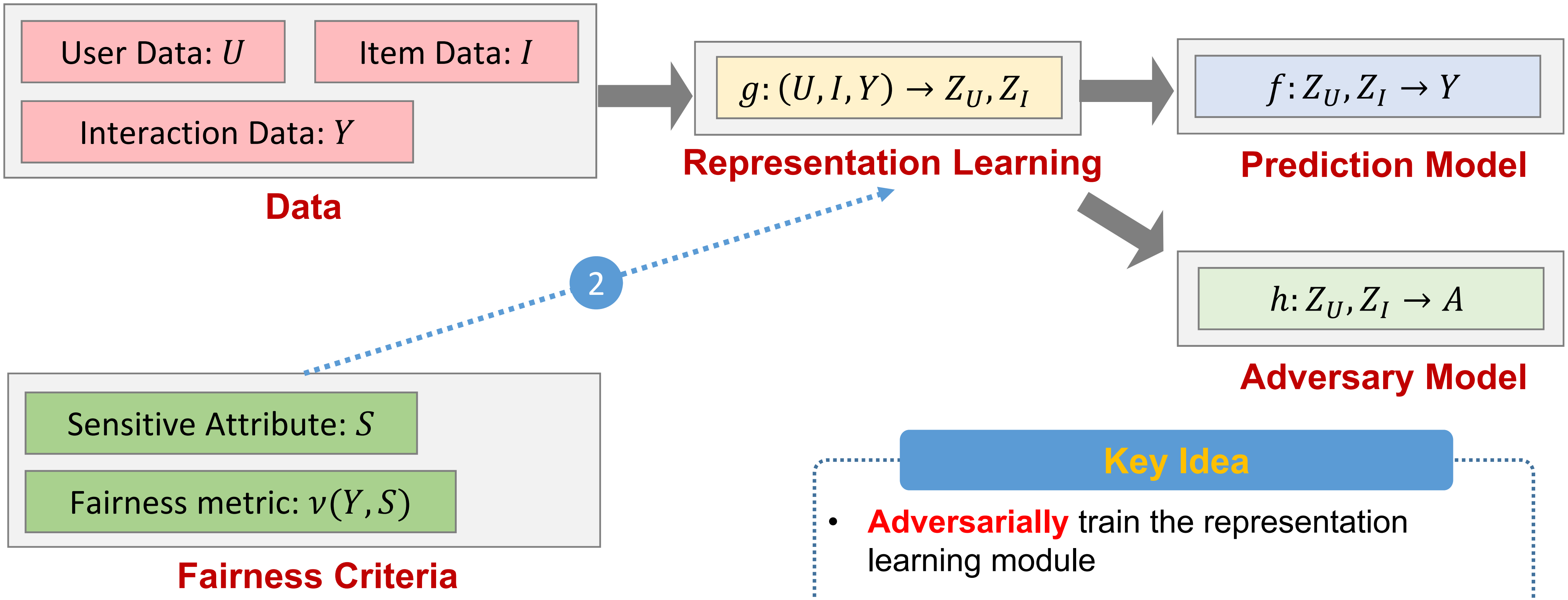
	Ease of implementation and (re-)use	Scalability	Ease of auditing	Fairness / Performance tradeoff	Generalization
Pre-processing, e.g., representation learning	✖	✖	✖		✖
In-processing, i.e., joint learning and fairness regulation			✖	✖	✖
Post-processing, e.g., threshold adjustment		✖	✖		

[Moustapha Cisse & Sanmi Koyejo; Fairness and Representation Learning]

• Four Research Lines towards Fairness

- ☐ Rebalancing
- ☐ Regularization
- ☐ Fair Representation Learning

• Line 3: Adversarial Learning → Fair Representation Learning



Key Idea

- **Adversarially** train the representation learning module
- Based on the representations, the prediction & adversary models paly a min-max game

• **Example 1:** Adversarial Learned Fair Representation (ALFR)

Reconstruction loss between input data X and representations R

Prediction error in generating prediction Y from R

$$\max_{\phi}\min_{\theta}\mathcal{L} = \alpha C_{\theta}(X,R) + \beta D_{\theta,\phi}(R,A) + \gamma E_{\theta}(Y,R)$$

Training an adversary model to encourage the independence between the representation R and the sensitive attributes A, **rather than a regularization term**

Predicting sensitive attributes from the representations R

$$D = \mathbb{E}_{X,A} A \cdot \log(f(R)) + (1-A)\cdot\log(1-f(R))$$

Cross entropy for binary sensitive attribute

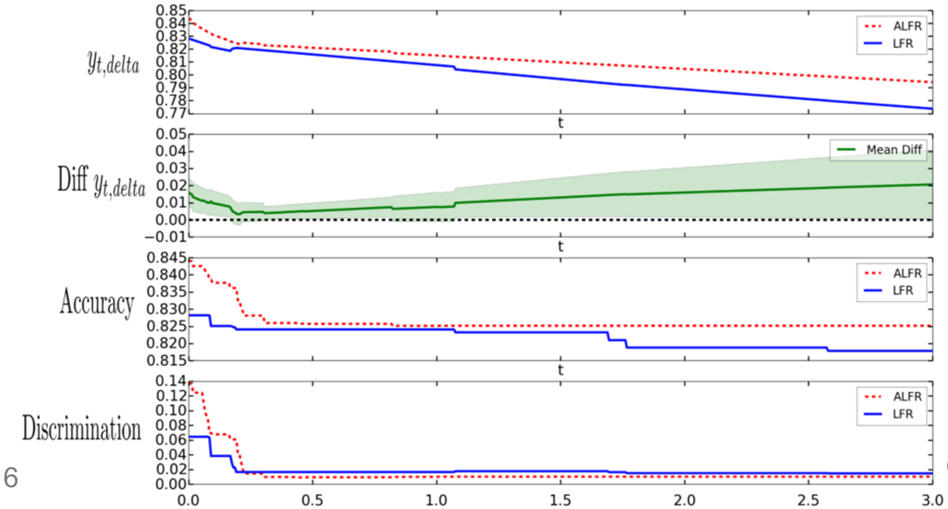
Idea

Adversarial Representation Learning

- **encode insensitive attributes** of data
- **remove any information about sensitive attributes**

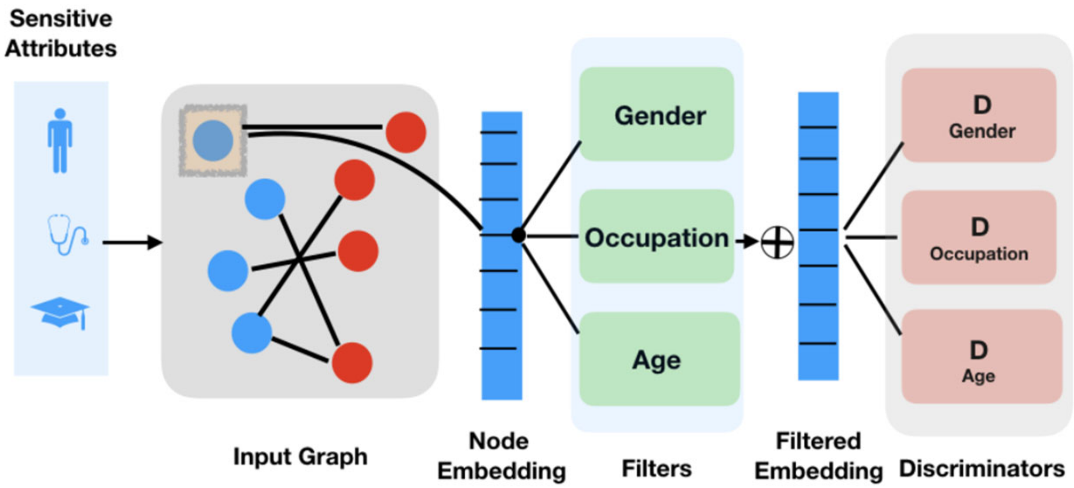
Findings

- Achieve better performance & fairness than LFR (regularization)



[Edwards et al.: CENSORING REPRESENTATIONS WITH AN ADVERSARY. ICLR 2016

• **Example 2:** Compositional Fairness Constraints for Graph Embeddings



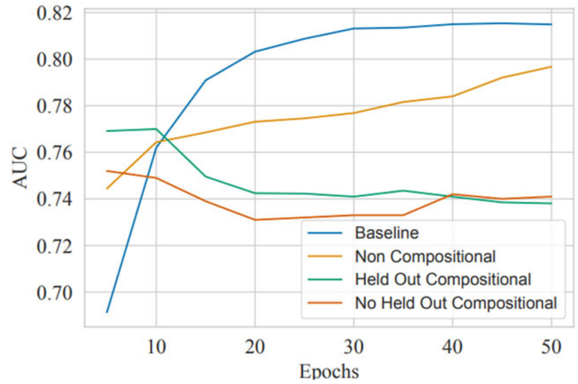


Figure 3. Performance on the edge prediction (i.e., recommendation) task on the Reddit data. Evaluation is using the AUC score, since there is only one edge/relation type.

Idea

Based on ALFR
- Focusing on **graph structured data**
- Flexibly accommodate **different combinations of fairness constraints** → **compositional fairness**

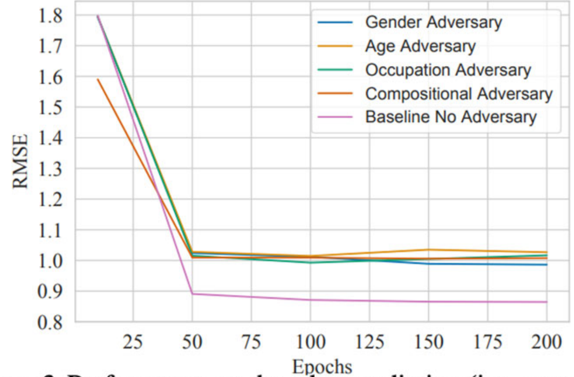
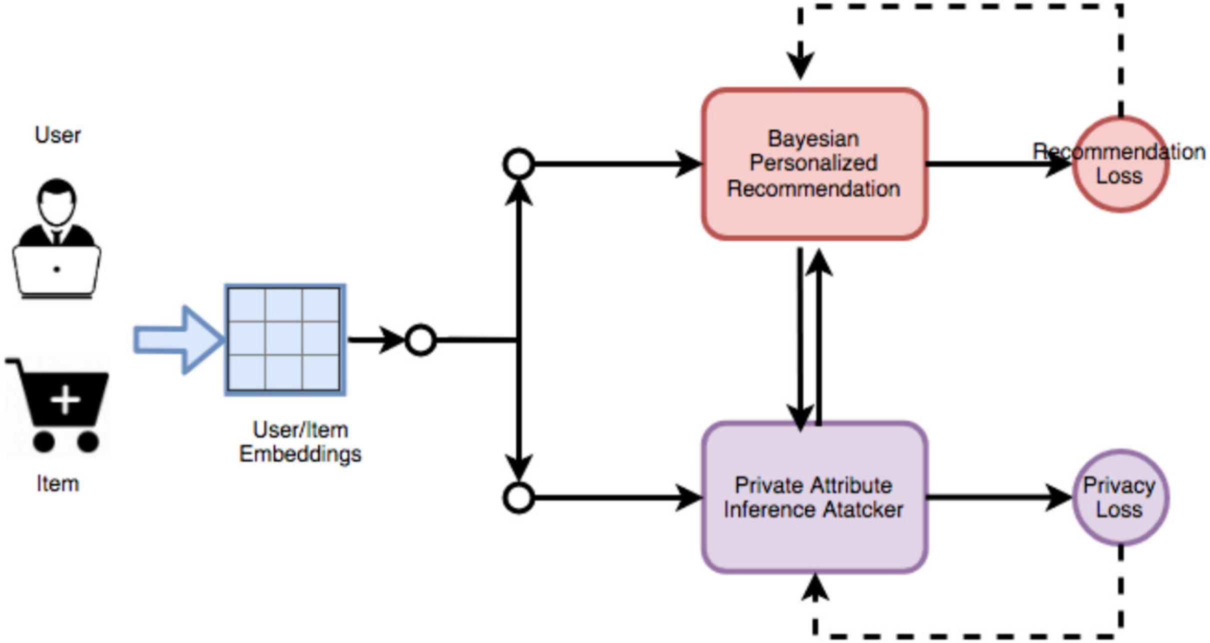


Figure 2. Performance on the edge prediction (i.e., recommendation) task on MovieLens, using RMSE as in Berg et al. (2017).

[Bose & Hamilton et al.: Compositional Fairness Constraints for Graph Embeddings. ICML 2019]

• **Example 3:** Recommendation with Attribute Protection (RAP)

$$\min_{\theta_R} \frac{1}{N}\sum_{h=1}^{N}\left[\sum_{(h,j,k)\in\mathcal{D}_h} -\ln\sigma\big((\hat{y}_{hj}(\theta_R)-\hat{y}_{hk}(\theta_R)).\,g(h,j,k)\big) - \alpha\left[\frac{1}{T}\sum_{t=1}^{T}\mathcal{L}_{D_P^t}(\hat{p}_{h,t},p_{h,t})\right]\right]+\lambda\Omega(\theta)$$



$$\min_{\{\theta_P^t\}^{T}\_\{t=1\}} \frac{1}{N}\sum_{h=1}^{N}\left[\frac{1}{T}\sum_{t=1}^{T}\mathcal{L}_{D_P^t}(\hat{p}_{h,t},p_{h,t})\right]$$

Idea

Based on ALFR
- Focusing on **recommendation scenarios**
- **Prediction model** → BPR
- **Adversary model** → Private attribute inference attacker

Model	Gen	Age	35 Occ	*P@K*	*R@K*
ORIGINAL	0.7662	0.7050	0.8332	0.156	0.156
LDP-SH	0.6587	0.6875	0.8076	0.071	0.071
BLURME	0.6266	0.6177	0.7614	0.118	0.118
RAP	**0.6039**	**0.5397**	**0.7319**	**0.152**	**0.152**

[Beigi et al.: Privacy-Aware Recommendation with Private-Attribute Protection using Adversarial Learning. WSDM 2020]

• Summary

Pros:

- Representation learning can centralize fairness constraints
- Representation learning can simplify and centralize the task of fairness auditing
- Learned representations can be constructed to satisfy multiple fairness measures simultaneously
- Learned representations can simplify the task of evaluating the fairness/performance tradeoff, e.g., using performance bounds

Cons:

- Less precise control of fairness/performance tradeoff, than joint learning …
- May lead to fairness overconfidence …

[Moustapha Cisse & Sanmi Koyejo; Fairness and Representation Learning]

THANK YOU!