Disentangling User Interest and Conformity for Recommendation with Causal Embedding

***Yu Zheng*[1]**, *Chen Gao*[1], *Xiang Li*[2], *Xiangnan He*[3], *Depeng Jin*[1], *Yong Li*[1]

Tsinghua University[1]

University of Hong Kong[2]

University of Science and Technology of China[3]

liyong07@tsinghua.edu.cn

Background

- What are the **causes** behind each user-item interaction?

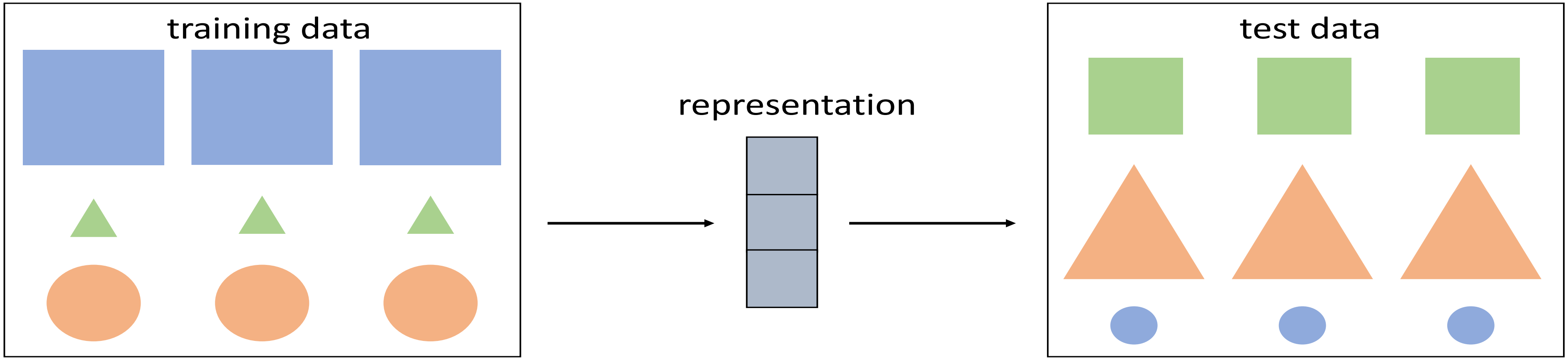
There are two main causes:

- **Interest**
- **Conformity**
  - How users tend to follow other people

Goal: Learn disentangled representations for interest and conformity

Motivation

- Why learning disentangled representations?
  - Causal recommendation under **non-IID situations**!
  - IID: independent and identically distributed



- **Robustness**
  - Recommenders are trained and updated in real-time
  - Training data and test data are not IID
- **Interpretability**
  - Improve user-friendliness
  - Facilitates algorithm developing

Causal Recommendation

- Inverse Propensity Scoring (IPS)[1]

$$\hat{R}_{\mathrm{IPS}}(\hat{Z}|P) = \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{1}{|\mathcal{S}_u|}\sum_{i\in\mathcal{S}_u^*}\frac{c(\hat{Z}_{u,i})}{P_{u,i}}$$

$$= \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{1}{|\mathcal{S}_u|}\sum_{i\in\mathcal{S}_u}\frac{c(\hat{Z}_{u,i})}{\boxed{P_{u,i}}}\cdot O_{u,i}$$

propensity score

- Propensity score is estimated from item popularity
- Intuition: impose lower weights on popular items, and boost unpopular items
- Interest and popularity are bundled as one unified representation

Two factors are **entangled**!

[1] Yang, L., Cui, Y., Xuan, Y., Wang, C., Belongie, S., & Estrin, D. (2018, September). Unbiased offline recommender evaluation for missing-not-at-random implicit feedback. In Proceedings of the 12th ACM Conference on Recommender Systems (pp. 279-287).

Causal Recommendation

- Causal Embeddings (CausE)[1]

MF on large biased data

$$L_{CausE}^{prod} = \underbrace{L(U\Theta_t, Y_t) + \Omega(\Theta_t)}_{treatment\ task\ loss} + \underbrace{L(U\Theta_c, Y_c) + \Omega(\Theta_c)}_{control\ task\ loss} + \underbrace{\Omega(\Theta_t - \Theta_c)}_{regularizer\ between\ tasks}$$

regularization on two MF

MF on small unbiased data

- Require a large fraction of **biased** data and a small fraction of **unbiased** data
- Perform two MF on biased and unbiased data, respectively
- Impose L1/L2 regularization on two MF

Still **entangled** representations!

[1] Bonner, S., & Vasile, F. (2018, September). Causal embeddings for recommendation. In Proceedings of the 12th ACM conference on recommender systems (pp. 104-112).

Disentangling interest and conformity

- Variety of conformity
- Conformity depends on **both users and items**
- One user's conformity varies on **different items**, and conformity towards one item varies for **different users**

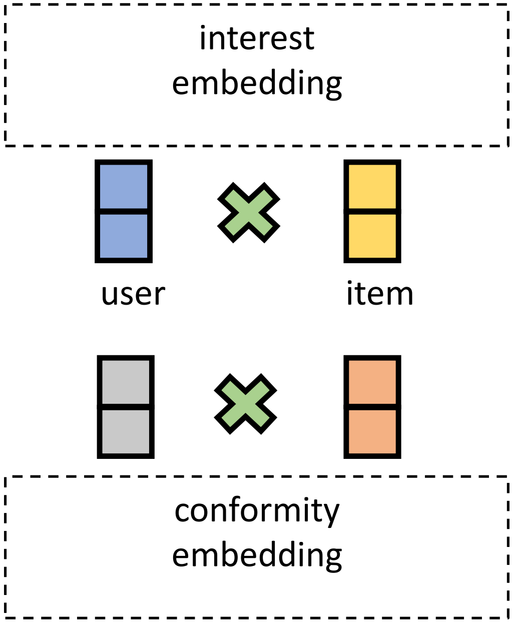
- Learning disentangled representations is intrinsically hard
- Only observational data is accessible.
- **No ground-truth** for user interest.

- An interaction can come from one or both factor
- Careful designs are needed for **combining the two factors** to make recommendations.

Methodology: Our DICE Model

- Disentangling Interest and Conformity with Causal Embedding (DICE)

- **Challenge 1**: Variety of conformity
- **Our proposal**: Adopt separate embeddings of interest and conformity for users and items
  - **Benefit 1**: Embedding proximity in high dimensional space can express the variety of conformity (**challenge 1 addressed**)
  - **Benefit 2**: Independent modeling of interest and conformity

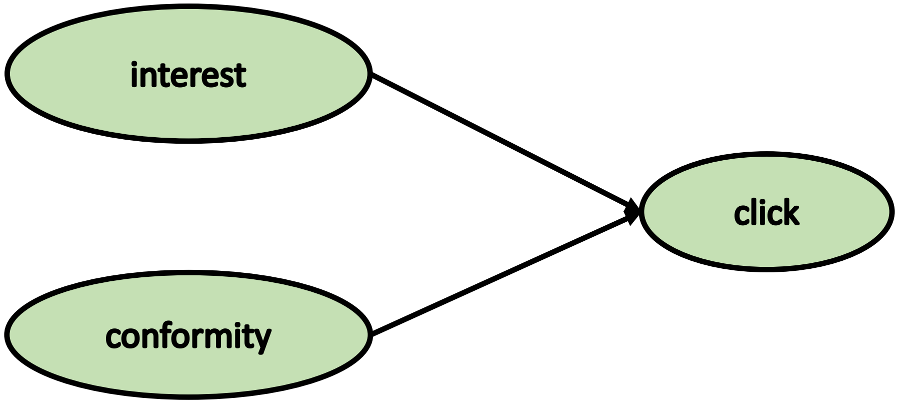


Methodology: Our DICE Model

- Disentangling Interest and Conformity with Causal Embedding (DICE)

- **Challenge 2**: Learning disentangled representations is intrinsically hard
- **Our proposal**: Utilize the colliding effect from causal inference to obtain cause-specific data.

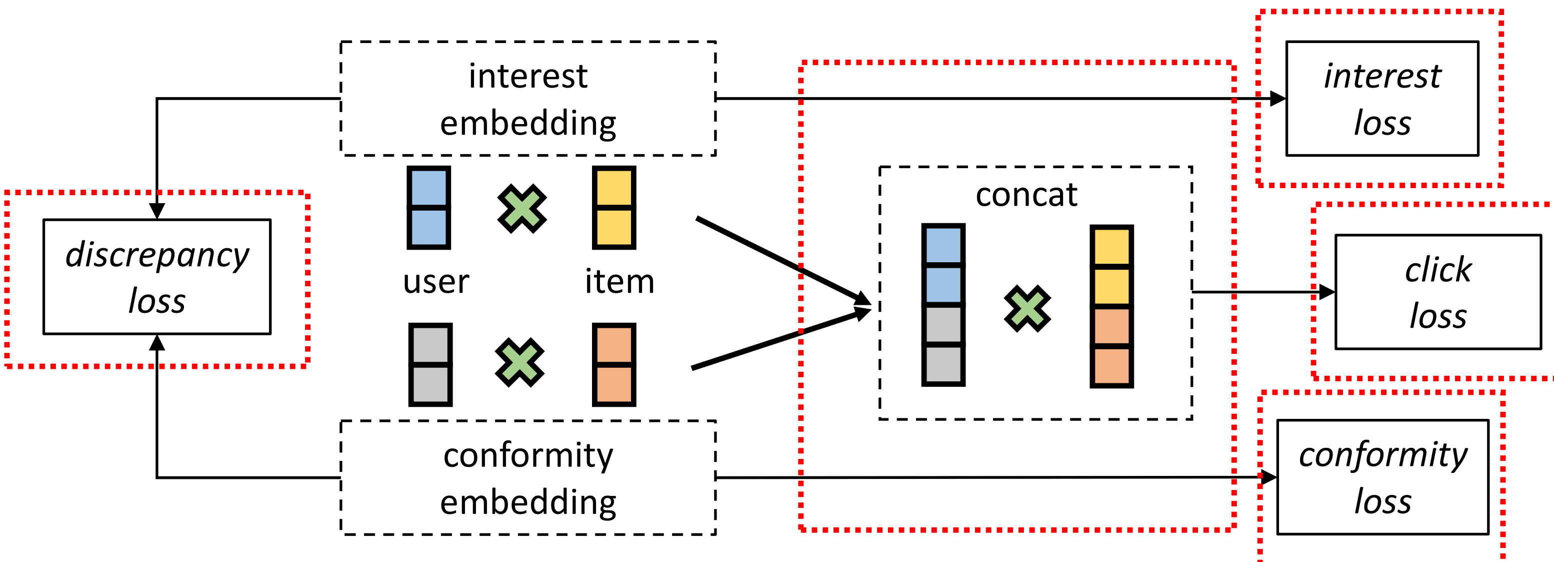
Intuition:
Train interest/conformity embeddings with interactions that are caused by interest/conformity



Methodology: Our DICE Model

- Disentangling Interest and Conformity with Causal Embedding (DICE)

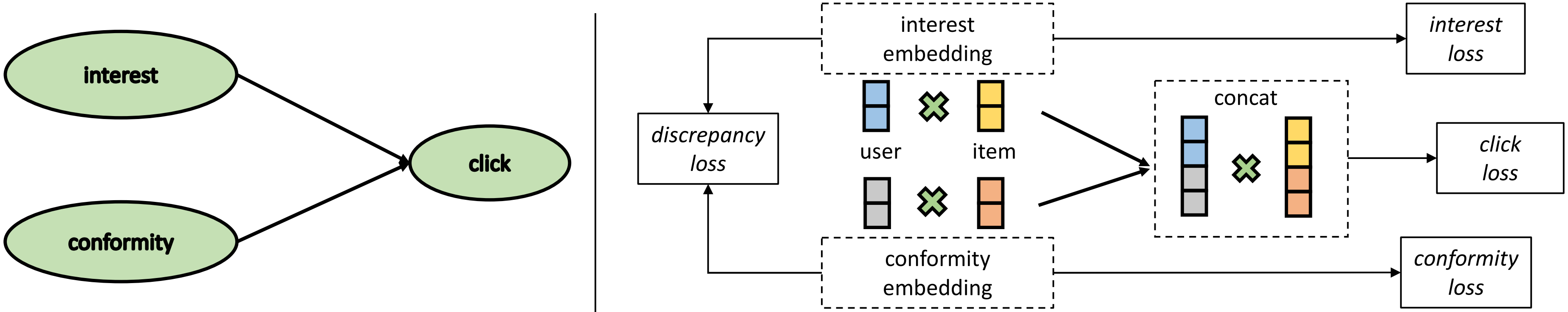
- **Challenge 3**: Aggregation of the two factors is complicated
- **Our proposal**: Leverage multi-task curriculum learning to combine the two causes.



Methodology: Our DICE Model

- Disentangling Interest and Conformity with Causal Embedding (DICE)

- Causal Embedding
- Disentangled Representation Learning
- Multi-task Curriculum Learning



(a) Causal Graph

(b) Causal Embedding

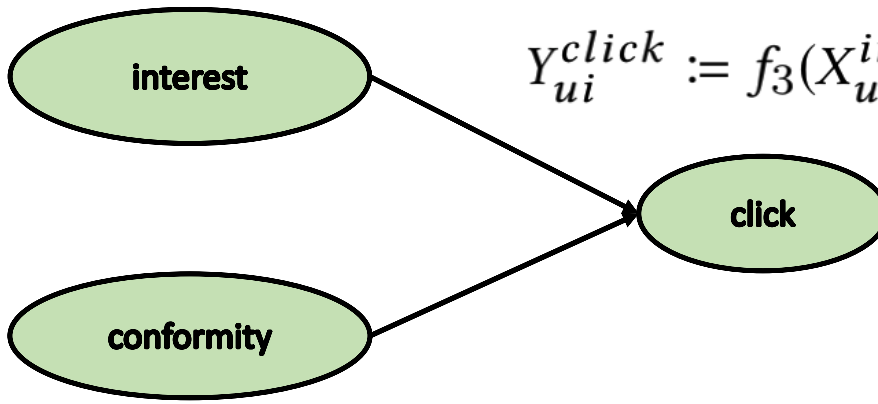
Methodology: Our DICE Model

- Causal graph and Structural Causal Model (SCM)

causal graph

$$X_{ui}^{int} := f_1(u, i, N^{int}),$$

$$Y_{ui}^{click} := f_3(X_{ui}^{int}, X_{ui}^{con}, N^{click}),$$



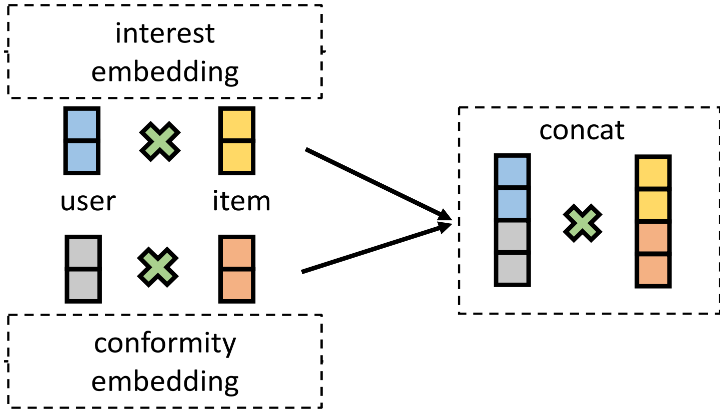
$$X_{ui}^{con} := f_2(u, i, N^{con}),$$

SCM

$$\boxed{\begin{aligned} X_{ui}^{int} &:= f_1(u, i, N^{int}), \\ X_{ui}^{con} &:= f_2(u, i, N^{con}), \\ Y_{ui}^{click} &:= f_3(X_{ui}^{int}, X_{ui}^{con}, N^{click}), \end{aligned}}$$

Methodology: Our DICE Model

- Causal embedding
- **Separate embeddings** for interest and conformity
  - User: $u^{(int)}, u^{(\mathrm{con})}$
  - Item: $i^{(int)}, i^{(\mathrm{con})}$
- Use **inner product** to compute matching score
- Predict click by combining two causes

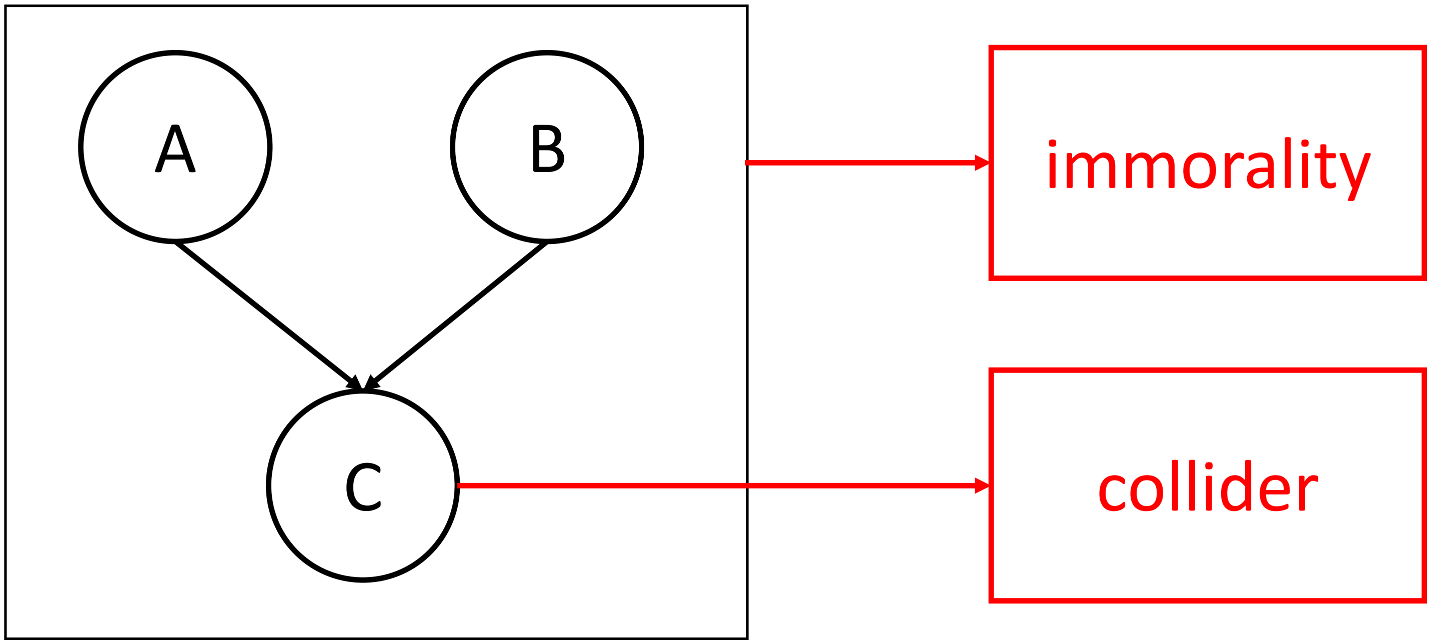


$$s_{ui}^{int} = \langle \boldsymbol{u}^{(\mathrm{int})}, \boldsymbol{i}^{(\mathrm{int})} \rangle, \quad s_{ui}^{con} = \langle \boldsymbol{u}^{(\mathrm{con})}, \boldsymbol{i}^{(\mathrm{con})} \rangle,$$

$$s_{ui}^{click} = s_{ui}^{int} + s_{ui}^{con},$$

Methodology: Our DICE Model

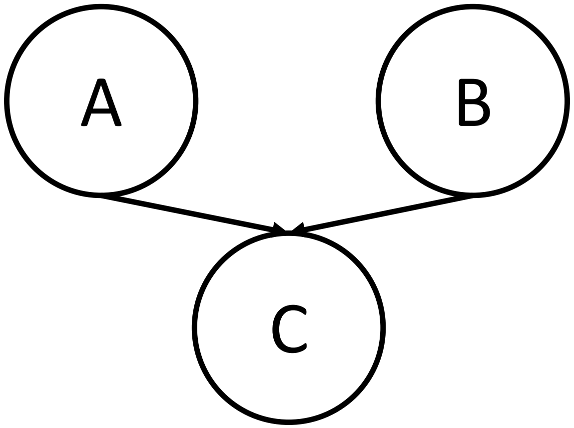
- Mining **cause-specific** data with **causal inference**
- Immorality and collider



- **Colliding effect**
- A and B are independent
- A and B are **NOT** independent when **conditioned on C**

Methodology: Our DICE Model

- Mining **cause-specific** data with **causal inference**
- e.g.
  - A: whether a student is talented
  - B: whether a student is hard-working
  - C: whether a student passes an exam



- **Bob passes the exam, and Bob is not talented**

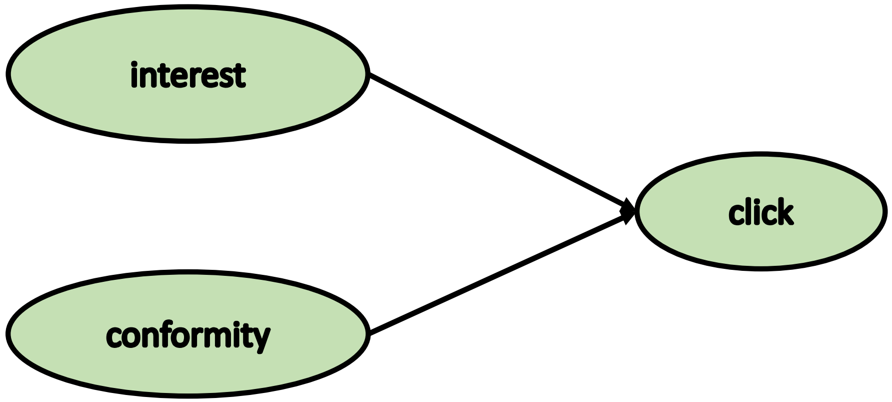
  ➡ He is hard-working with high probability

- **Alice doesn't pass the exam, and Alice is talented**

  ➡ She is most likely not hard-working

Methodology: Our DICE Model

- Mining **cause-specific** data with **causal inference**
- The **colliding effect** can come to help!
- Click is the collider of interest and conformity!



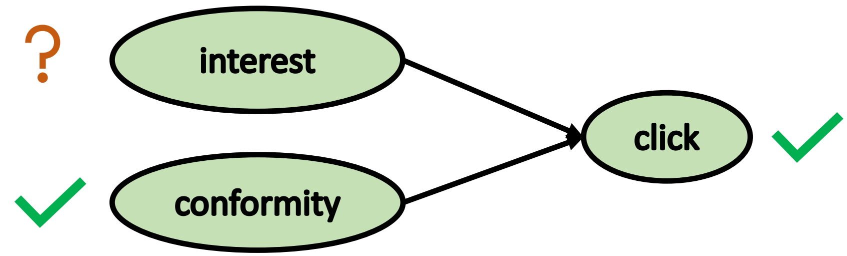
- Use popularity as a **proxy** for conformity
- A clicked item with low popularity
  ➡ high interest
- An unclicked item with high popularity
  ➡ low interest

Methodology: Our DICE Model

- Notation
- M^I : interest matching probability matrix
- M^C : conformity matching probability matrix

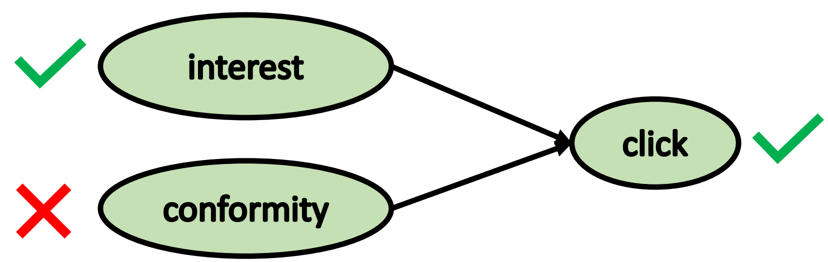
Case 1: u clicks a popular item a, doesn't click an unpopular item b

$$M^C_{ua} > M^C_{ub},$$
$$M^I_{ua} + M^C_{ua} > M^I_{ub} + M^C_{ub}.$$



Case 2: u clicks an unpopular item c, doesn't click a popular item d

$$M^I_{uc} > M^I_{ud}, M^C_{uc} < M^C_{ud},$$
$$M^I_{uc} + M^C_{uc} > M^I_{ud} + M^C_{ud}.$$



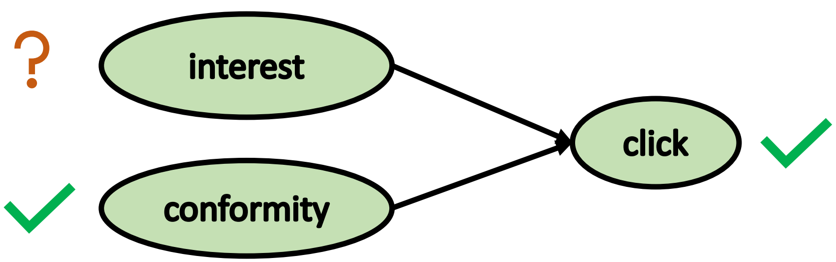
Methodology: Our DICE Model

- $\mathcal{O}$: whole training set (u, i, j): user, pos item, neg item
- $\mathcal{O}_1$: negative samples **more popular** than positive samples
- $\mathcal{O}_2$: negative samples **less popular** than positive samples

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

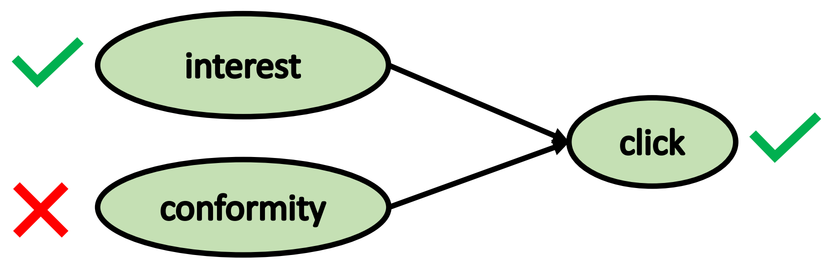
$\mathcal{O}_1$

$$M^C_{ua} > M^C_{ub},$$
$$M^I_{ua} + M^C_{ua} > M^I_{ub} + M^C_{ub}.$$



$\mathcal{O}_2$

$$M^I_{uc} > M^I_{ud}, M^C_{uc} < M^C_{ud},$$
$$M^I_{uc} + M^C_{uc} > M^I_{ud} + M^C_{ud}.$$



Solution: train different embeddings with different **cause-specific** data

Methodology: Our DICE Model

- Main task: estimating clicks

$$\mathcal{O}_1$$

$$M^C_{ua} > M^C_{ub},$$

$$\boxed{M^I_{ua} + M^C_{ua} > M^I_{ub} + M^C_{ub}.}$$

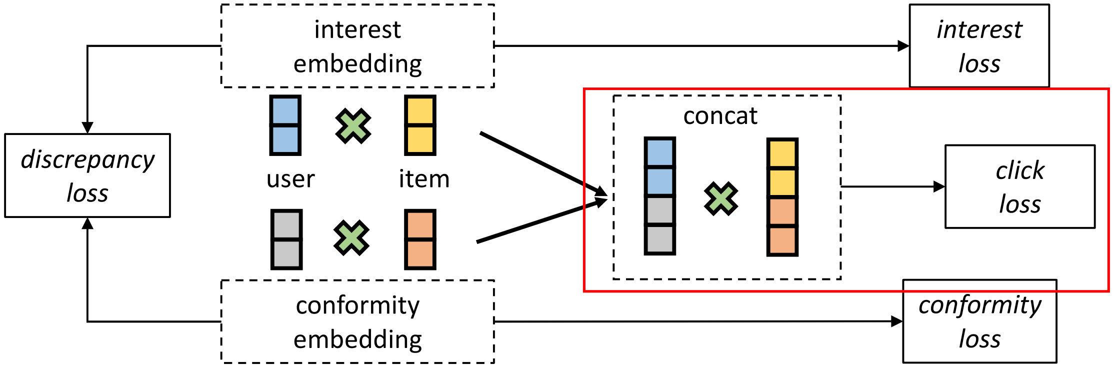
$$\mathcal{O}_2$$

$$M^I_{uc} > M^I_{ud}, M^C_{uc} < M^C_{ud},$$

$$\boxed{M^I_{uc} + M^C_{uc} > M^I_{ud} + M^C_{ud}.}$$

$$\boldsymbol{u}^t = \boldsymbol{u}^{(\mathrm{int})} \| \boldsymbol{u}^{(\mathrm{con})}, \boldsymbol{i}^t = \boldsymbol{i}^{(\mathrm{int})} \| \boldsymbol{i}^{(\mathrm{con})}, \boldsymbol{j}^t = \boldsymbol{j}^{(\mathrm{int})} \| \boldsymbol{j}^{(\mathrm{con})},$$

$$L^{\mathcal{O}_1+\mathcal{O}_2}_{\mathrm{click}} = \sum_{(u,i,j)\in\mathcal{O}} \mathrm{BPR}(\langle \boldsymbol{u}^t, \boldsymbol{i}^t \rangle, \langle \boldsymbol{u}^t, \boldsymbol{j}^t \rangle).$$



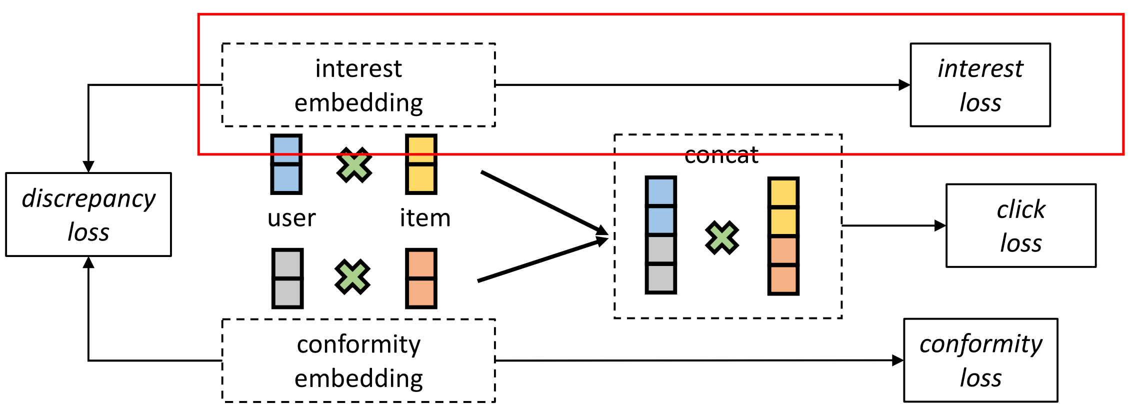
Methodology: Our DICE Model

- Interest modeling
  - Only use interest embedding

$$\mathcal{O}_2$$

$$\boxed{M^I_{uc} > M^I_{ud},} M^C_{uc} < M^C_{ud},$$
$$M^I_{uc} + M^C_{uc} > M^I_{ud} + M^C_{ud}.$$

$$L^{\mathcal{O}_2}_{\text{interest}} = \sum_{(u,i,j)\in\mathcal{O}_2} \mathrm{BPR}(\langle \boldsymbol{u}^{(\text{int})}, \boldsymbol{i}^{(\text{int})}\rangle, \langle \boldsymbol{u}^{(\text{int})}, \boldsymbol{j}^{(\text{int})}\rangle).$$



Methodology: Our DICE Model

- Conformity modeling
  - Only use conformity embedding

$\mathcal{O}_1$

$$\boxed{M^C_{ua} > M^C_{ub},}$$
$$M^I_{ua} + M^C_{ua} > M^I_{ub} + M^C_{ub}.$$

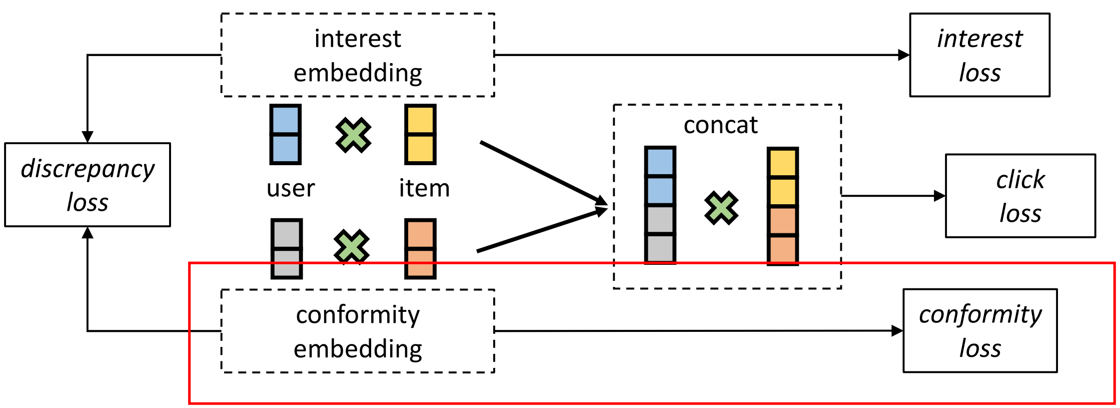
$$L^{O_1}_{\text{conformity}} = \sum_{(u,i,j)\in O_1} \text{BPR}(\langle \boldsymbol{u}^{(\text{con})}, \boldsymbol{i}^{(\text{con})}\rangle, \langle \boldsymbol{u}^{(\text{con})}, \boldsymbol{j}^{(\text{con})}\rangle),$$

$\mathcal{O}_2$

$$M^I_{uc} > M^I_{ud}, \boxed{M^C_{uc} < M^C_{ud},}$$
$$M^I_{uc} + M^C_{uc} > M^I_{ud} + M^C_{ud}.$$

$$L^{O_2}_{\text{conformity}} = \sum_{(u,i,j)\in O_2} -\text{BPR}(\langle \boldsymbol{u}^{(\text{con})}, \boldsymbol{i}^{(\text{con})}\rangle, \langle \boldsymbol{u}^{(\text{con})}, \boldsymbol{j}^{(\text{con})}\rangle),$$

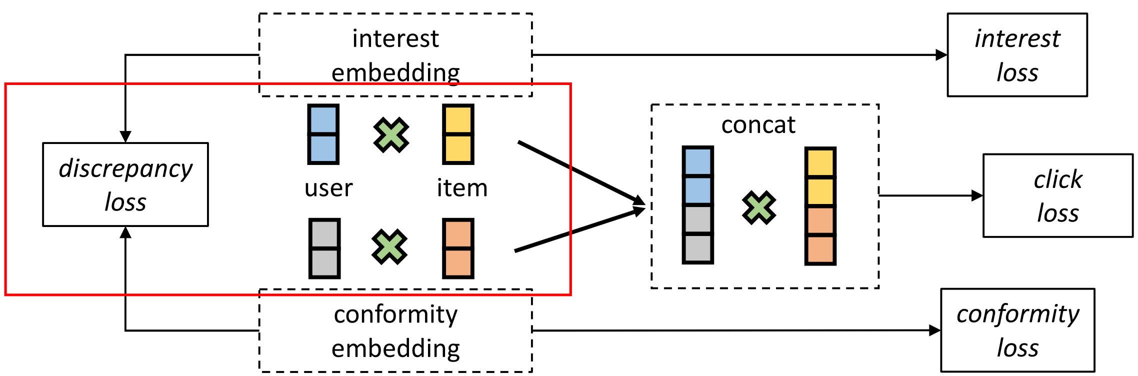
$$L^{O_1+O_2}_{\text{conformity}} = L^{O_1}_{\text{conformity}} + L^{O_2}_{\text{conformity}}.$$



Methodology: Our DICE Model

- Discrepancy task
  - direct supervision on disentanglement
- L1-inv: $-L1(\mathrm{E}^{(int)}, \mathrm{E}^{(con)})$
- L2-inv: $-L2(\mathrm{E}^{(int)}, \mathrm{E}^{(con)})$
- distance correlation:

$$dCor\left(\mathrm{E}^{(int)}, \mathrm{E}^{(con)}\right) = \frac{dCov(\mathrm{E}^{(int)}, \mathrm{E}^{(con)})}{\sqrt{dVar(\mathrm{E}^{(int)}) \cdot dVar(\mathrm{E}^{(con)})}}$$



Methodology: Our DICE Model

- Multi-task learning

$$L = L^{O_1+O_2}_{\text{click}} + \alpha(L^{O_2}_{\text{interest}} + L^{O_1+O_2}_{\text{conformity}}) + \beta L_{\text{discrepancy}}.$$

- Popularity based Negative Sampling with Margin (PNSM)
  - Popularity of the positive item: p
  - Sample negative items with popularity:
    - Larger than $p + m$
    - Lower than $p - m$
  - Large m: high confidence on inequalities, **easy**
  - Small m: low confidence on inequalities, **hard**

- Curriculum learning: an easy-to-hard strategy
  - **decay** m, α and β by a factor of **0.9** after each epoch

Experiments

Dataset	User	Item	Interaction	Ent. Train	Ent. Test
Movielens-10M	37962	4819	1371473	6.22	7.97
Netflix	32450	8432	2212690	6.85	8.54

- Datasets:
  - Movielens-10M
  - Netflix
- Evaluation: non-IID protocol (same with CausE[1]):
  - Train: 60% normal+ 10% intervened
  - Validation: 10% intervened
  - Test: 20% intervened
- Metrics:
  - Recall, Hit Ratio, NDCG
- Recommendation models
  - MF[2]
  - LightGCN[3]

[1] Bonner, S., & Vasile, F. (2018, September). Causal embeddings for recommendation. In Proceedings of the 12th ACM conference on recommender systems (pp. 104-112).
[2] Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2012). BPR: Bayesian personalized ranking from implicit feedback. arXiv preprint arXiv:1205.2618.
[3] He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020, July). Lightgcn: Simplifying and powering graph convolution network for recommendation. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 639-648).

Experiments

- **RQ1:** How does our proposed DICE framework perform compared with state-of-the-art causal recommendation methods under **non-IID circumstances**?

- **RQ2:** Can the proposed DICE framework guarantee **interpretability**?

- **RQ3:** Can the proposed DICE framework guarantee **robustness**?

Experiments

- Overall Comparison

Table 2: Overall performance on Movielens-10M dataset and Netflix dataset.

Dataset		Movielens-10M						Netflix					
		TopK = 20			TopK = 50			TopK = 20			TopK = 50		
Model	Method	Recall	HR	NDCG	Recall	HR	NDCG	Recall	HR	NDCG	Recall	HR	NDCG
MF	None	0.1286	0.4429	0.0846	0.2346	0.6295	0.1170	<u>0.1122</u>	<u>0.5194</u>	0.0943	0.1928	<u>0.6749</u>	0.1185
	IPS	0.1335	0.4434	0.0852	0.2376	0.6288	0.1174	0.1058	0.4882	0.0864	0.1855	0.6562	0.1112
	IPS-C	0.1367	0.4564	0.0875	0.2429	0.6383	0.1203	0.1119	0.5046	0.0919	<u>0.1938</u>	0.6700	0.1174
	IPS-CN	<u>0.1412</u>	<u>0.4700</u>	<u>0.0925</u>	<u>0.2509</u>	<u>0.6477</u>	<u>0.1264</u>	0.1080	0.5042	0.0935	0.1912	0.6621	0.1185
	IPS-CNSR	0.1365	0.4588	0.0895	0.2419	0.6366	0.1219	0.1110	0.5159	<u>0.0948</u>	0.1937	0.6713	<u>0.1192</u>
	CausE	0.1157	0.4066	0.0744	0.2121	0.5924	0.1037	0.0935	0.4641	0.0782	0.1651	0.6272	0.0994
	DICE	**0.1634**	**0.5197**	**0.1084**	**0.2872**	**0.6975**	**0.1468**	**0.1258**	**0.5545**	**0.1070**	**0.2164**	**0.7090**	**0.1345**
GCN	None	0.1378	0.4625	0.0898	0.2513	0.6505	0.1247	0.1026	0.4908	0.0870	0.1842	0.6609	0.1112
	IPS	0.1394	0.4645	0.0919	0.2538	0.6473	0.1275	0.1101	0.5091	0.0950	0.1941	0.6657	0.1203
	IPS-C	<u>0.1478</u>	<u>0.4829</u>	<u>0.0971</u>	<u>0.2654</u>	<u>0.6632</u>	<u>0.1339</u>	<u>0.1157</u>	<u>0.5219</u>	<u>0.1004</u>	<u>0.2037</u>	<u>0.6816</u>	<u>0.1270</u>
	IPS-CN	0.1119	0.3997	0.0701	0.2281	0.6112	0.1057	0.0726	0.3991	0.0643	0.1472	0.5841	0.0866
	IPS-CNSR	0.1300	0.4427	0.0852	0.2336	0.6282	0.1171	0.0826	0.4337	0.0715	0.1589	0.6124	0.0940
	CausE	0.1027	0.3729	0.0632	0.2044	0.5811	0.0941	0.0838	0.4289	0.0677	0.1569	0.6119	0.0902
	DICE	**0.1812**	**0.5563**	**0.1228**	**0.3100**	**0.7216**	**0.1629**	**0.1420**	**0.5910**	**0.1217**	**0.2367**	**0.7340**	**0.1499**

Experiments

- Observations

Table 2: Overall performance on Movielens-10M dataset and Netflix dataset.

Dataset		Movielens-10M						Netflix					
		TopK = 20			TopK = 50			TopK = 20			TopK = 50		
Model	Method	Recall	HR	NDCG	Recall	HR	NDCG	Recall	HR	NDCG	Recall	HR	NDCG
MF	None	0.1286	0.4429	0.0846	0.2346	0.6295	0.1170	<u>0.1122</u>	<u>0.5194</u>	0.0943	0.1928	<u>0.6749</u>	0.1185
	IPS	0.1335	0.4434	0.0852	0.2376	0.6288	0.1174	0.1058	0.4882	0.0864	0.1855	0.6562	0.1112
	IPS-C	0.1367	0.4564	0.0875	0.2429	0.6383	0.1203	0.1119	0.5046	0.0919	<u>0.1938</u>	0.6700	0.1174
	IPS-CN	<u>0.1412</u>	<u>0.4700</u>	<u>0.0925</u>	<u>0.2509</u>	<u>0.6477</u>	<u>0.1264</u>	0.1080	0.5042	0.0935	0.1912	0.6621	0.1185
	IPS-CNSR	0.1365	0.4588	0.0895	0.2419	0.6366	0.1219	0.1110	0.5159	<u>0.0948</u>	0.1937	0.6713	<u>0.1192</u>
	CausE	0.1157	0.4066	0.0744	0.2121	0.5924	0.1037	0.0935	0.4641	0.0782	0.1651	0.6272	0.0994
	DICE	**0.1634**	**0.5197**	**0.1084**	**0.2872**	**0.6975**	**0.1468**	**0.1258**	**0.5545**	**0.1070**	**0.2164**	**0.7090**	**0.1345**
GCN	None	0.1378	0.4625	0.0898	0.2513	0.6505	0.1247	0.1026	0.4908	0.0870	0.1842	0.6609	0.1112
	IPS	0.1394	0.4645	0.0919	0.2538	0.6473	0.1275	0.1101	0.5091	0.0950	0.1941	0.6657	0.1203
	IPS-C	<u>0.1478</u>	<u>0.4829</u>	<u>0.0971</u>	<u>0.2654</u>	<u>0.6632</u>	<u>0.1339</u>	<u>0.1157</u>	<u>0.5219</u>	<u>0.1004</u>	<u>0.2037</u>	<u>0.6816</u>	<u>0.1270</u>
	IPS-CN	0.1119	0.3997	0.0701	0.2281	0.6112	0.1057	0.0726	0.3991	0.0643	0.1472	0.5841	0.0866
	IPS-CNSR	0.1300	0.4427	0.0852	0.2336	0.6282	0.1171	0.0826	0.4337	0.0715	0.1589	0.6124	0.0940
	CausE	0.1027	0.3729	0.0632	0.2044	0.5811	0.0941	0.0838	0.4289	0.0677	0.1569	0.6119	0.0902
	DICE	**0.1812**	**0.5563**	**0.1228**	**0.3100**	**0.7216**	**0.1629**	**0.1420**	**0.5910**	**0.1217**	**0.2367**	**0.7340**	**0.1499**

- Our proposed DICE framework outperforms baselines with significant improvements with respect to all metrics on both datasets.

Experiments

- Observations

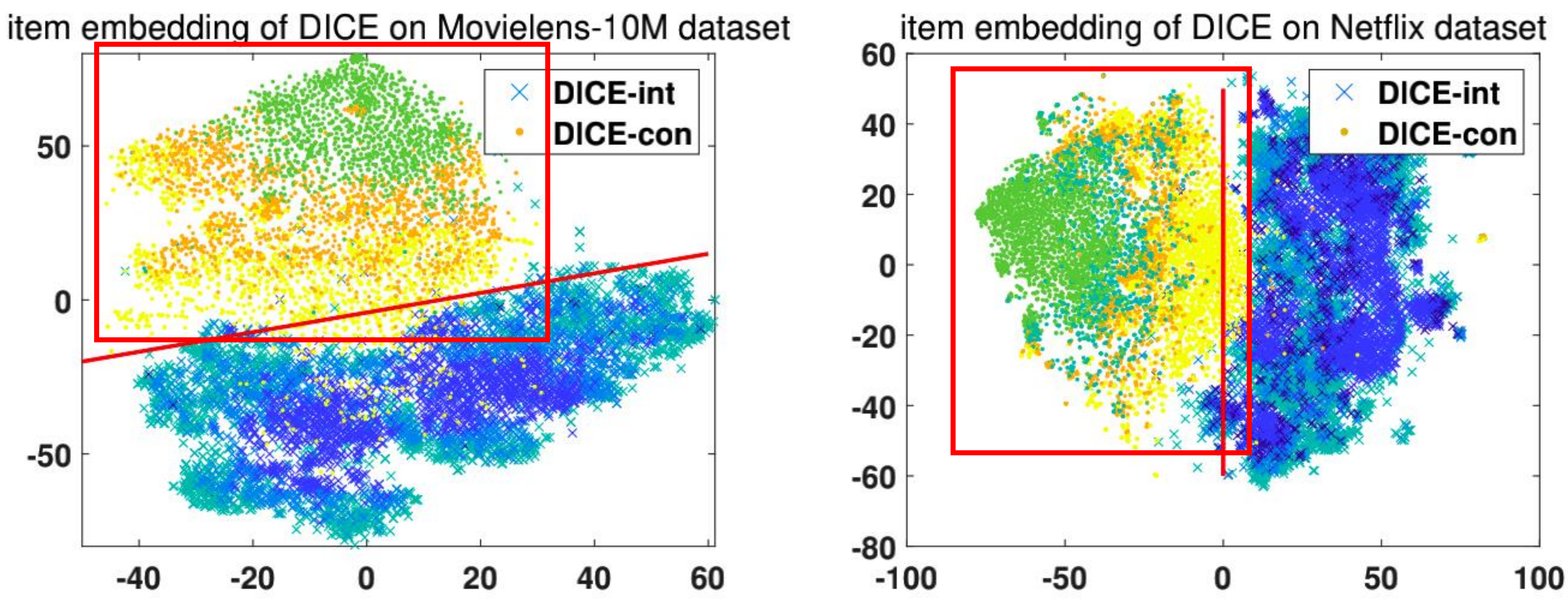
Table 2: Overall performance on Movielens-10M dataset and Netflix dataset.

Dataset		Movielens-10M						Netflix					
		TopK = 20			TopK = 50			TopK = 20			TopK = 50		
Model	Method	Recall	HR	NDCG	Recall	HR	NDCG	Recall	HR	NDCG	Recall	HR	NDCG
MF	None	0.1286	0.4429	0.0846	0.2346	0.6295	0.1170	0.1122	0.5194	0.0943	0.1928	0.6749	0.1185
	IPS	0.1335	0.4434	0.0852	0.2376	0.6288	0.1174	0.1058	0.4882	0.0864	0.1855	0.6562	0.1112
	IPS-C	0.1367	0.4564	0.0875	0.2429	0.6383	0.1203	0.1119	0.5046	0.0919	0.1938	0.6700	0.1174
	IPS-CN	0.1412	0.4700	0.0925	0.2509	0.6477	0.1264	0.1080	0.5042	0.0935	0.1912	0.6621	0.1185
	IPS-CNSR	0.1365	0.4588	0.0895	0.2419	0.6366	0.1219	0.1110	0.5159	0.0948	0.1937	0.6713	0.1192
	CausE	0.1157	0.4066	0.0744	0.2121	0.5924	0.1037	0.0935	0.4641	0.0782	0.1651	0.6272	0.0994
	DICE	**0.1634**	**0.5197**	**0.1084**	**0.2872**	**0.6975**	**0.1468**	**0.1258**	**0.5545**	**0.1070**	**0.2164**	**0.7090**	**0.1345**
GCN	None	0.1378	0.4625	0.0898	0.2513	0.6505	0.1247	0.1026	0.4908	0.0870	0.1842	0.6609	0.1112
	IPS	0.1394	0.4645	0.0919	0.2538	0.6473	0.1275	0.1101	0.5091	0.0950	0.1941	0.6657	0.1203
	IPS-C	0.1478	0.4829	0.0971	0.2654	0.6632	0.1339	0.1157	0.5219	0.1004	0.2037	0.6816	0.1270
	IPS-CN	0.1119	0.3997	0.0701	0.2281	0.6112	0.1057	0.0726	0.3991	0.0643	0.1472	0.5841	0.0866
	IPS-CNSR	0.1300	0.4427	0.0852	0.2336	0.6282	0.1171	0.0826	0.4337	0.0715	0.1589	0.6124	0.0940
	CausE	0.1027	0.3729	0.0632	0.2044	0.5811	0.0941	0.0838	0.4289	0.0677	0.1569	0.6119	0.0902
	DICE	**0.1812**	**0.5563**	**0.1228**	**0.3100**	**0.7216**	**0.1629**	**0.1420**	**0.5910**	**0.1217**	**0.2367**	**0.7340**	**0.1499**

- DICE is a highly general framework which can be combined with various recommendation models.

Experiments

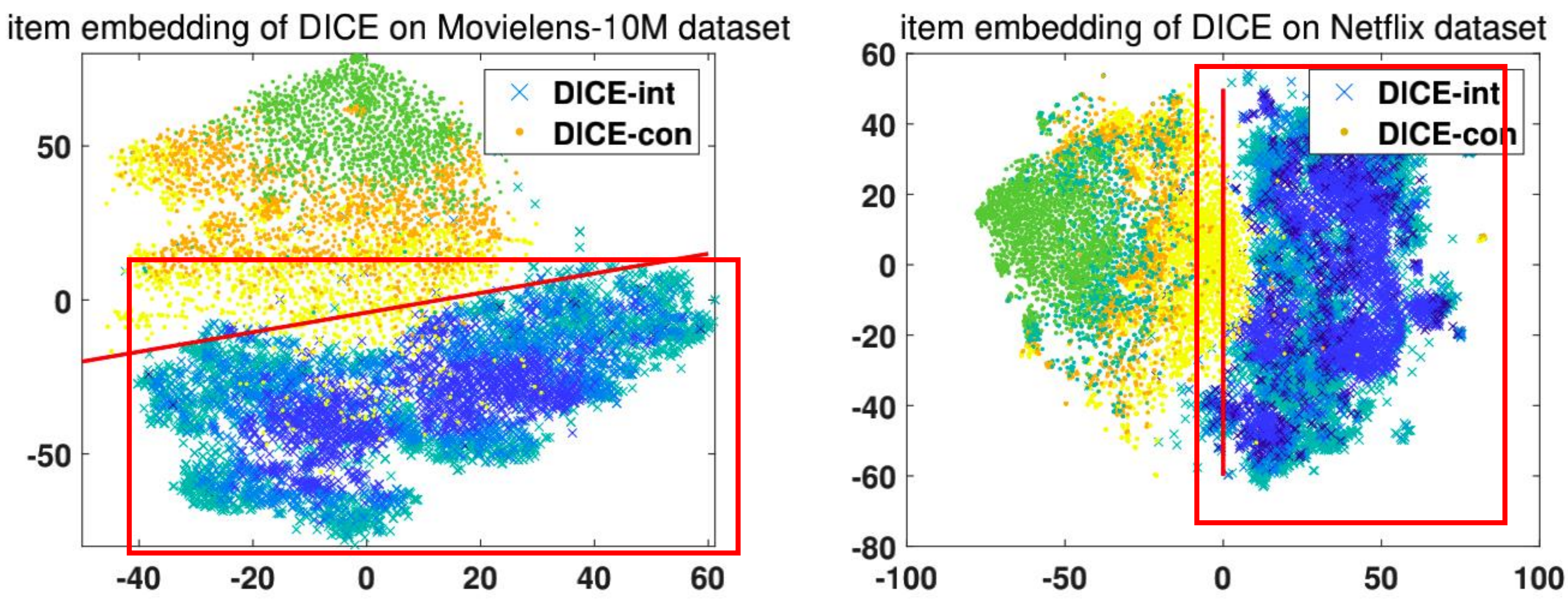
- Interpretability



- Conformity embeddings of items with different popularity form **layers.**

Experiments

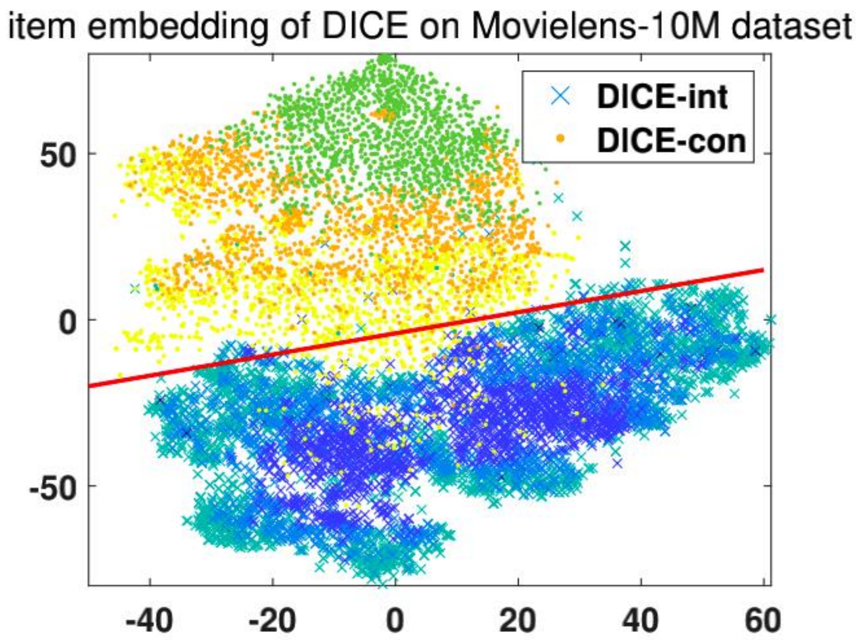
- Interpretability

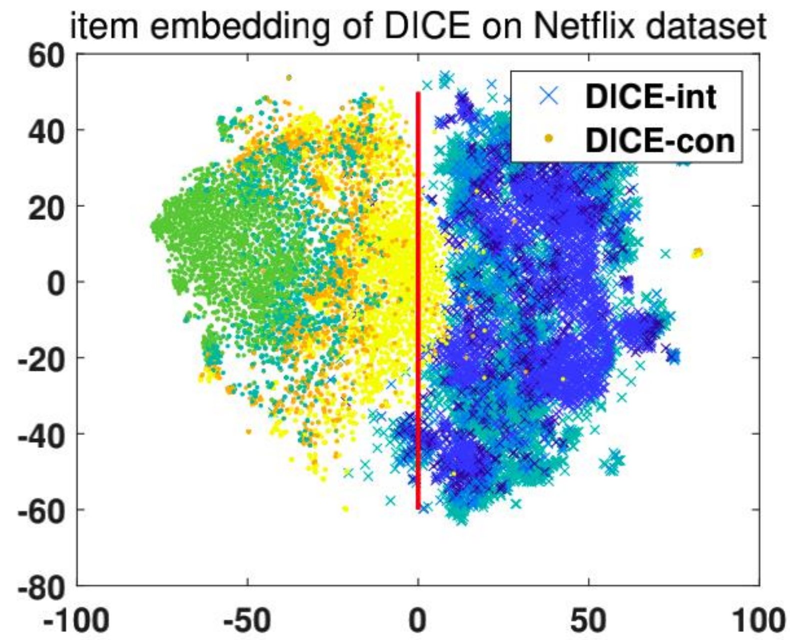


- Interest embeddings of items with different popularity are **uniformly** distributed in the space.

Experiments

- Interpretability

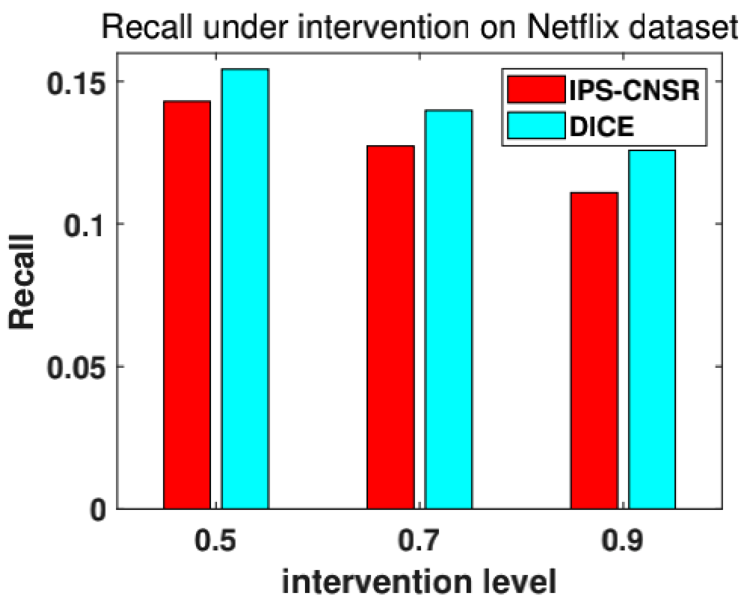


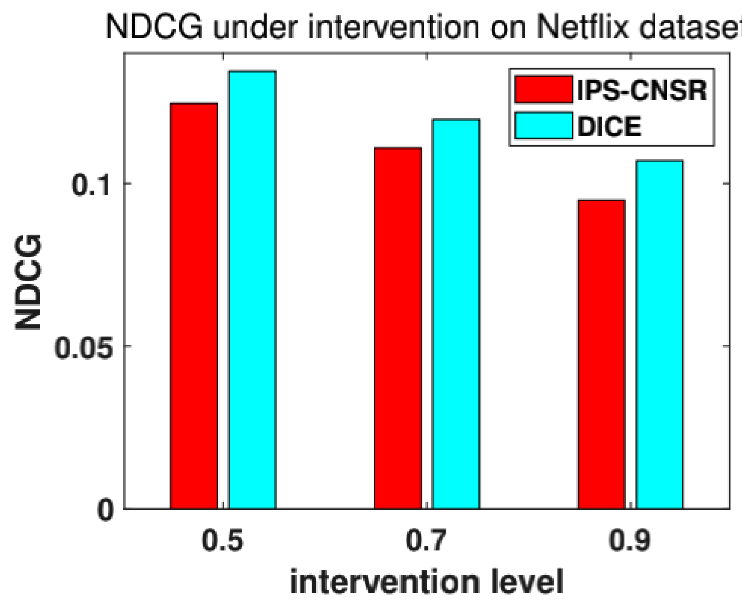


  - Conformity embeddings largely captures conformity, and interest embeddings squeeze out conformity

Experiments

- Robustness





- Test data with different strength of intervention
- DICE is more robust than IPS based method under different levels of intervention

Conclusion and Future Work

- We propose to learn disentangled representations of user interest and conformity for recommendation with tools of causal inference. A general framework DICE is developed which shows great robustness and interpretability under non-IID situations.

- **Future work**
- Extend DICE to **incorporate more features**.
- Learn disentangled representations for **finer-grained user interest**, e.g. price preference, brand preference…
- Codes can be found at: https://github.com/tsinghua-fib-lab/DICE

Thanks for listening!

Contact: liyong07@tsinghua.edu.cn
Lab Info: http://fi.ee.tsinghua.edu.cn