Causal Recommendation: Progresses and Future Directions

Lecture Tutorial for SIGIR 2023

Organizers: Wenjie Wang, Yang Zhang, Haoxuan Li, Peng Wu, Fuli Feng, Xiangnan He

23 July 2023
Webpage: https://causalrec.github.io/
Part 1 (90 min, 9:00—10:30)
  - Introduction (Wenjie Wang, 15 min)
  - Structural causal models for recommendation (Yang Zhang and Wenjie Wang, 60~70 min)
  - Q&A (5 min)
  - Coffee break (30 min)

Part 2 (90 min, 11:00-12:30)
  - Potential outcome framework for recommendation (Haoxuan Li and Peng Wu, 60~70 min)
  - Comparison (Fuli Feng, 2 min)
  - Conclusion, open problems, and future directions (Fuli Feng, 20 min)
  - Q&A (5 min)
Information Seeking

• **Information explosion era**
  - E-commerce: 12 million items in Amazon.
  - Social networks: 2.8 billion users in Facebook.
  - Content sharing platforms: 720,000 hours videos uploaded to Youtube per day.

• **Recommender system**

  Recommendation

Information seeking via implicit feedback

Recommender system is a powerful tool to address information overload.
Ecosystem of RecSys

- **Workflow of RecSys**
  - **Training**: RecSys is trained on observed user-item interactions.
  - **Serving**: RecSys infers user preference over items and recommend Top-K items.
  - **Collecting**: collect user interactions on the recommended items for further training.

- **Forming a feedback loop**
Shortcomings of Data-driven RecSys

• Bias in data (collecting):
  • Data is observational rather than experimental (missing-not-at-random)
  • Affected by many hidden factors:
    • Public opinions, etc.

• Bias shifting along time:
  • User/item feature changes
    • Income, marriage status
  • Preference shifting
Fitting Historical Data

- Minimizing the difference between historical feedback and model prediction

\[ f \]

- \[ f \text{ (Predicted Score)} = \begin{bmatrix} 3 & 2 & 1 & 4 \\ 2 & 2 & 3 & 5 \\ 2 & 4 & 5 & 5 \end{bmatrix} \]

- History feedback

\[ \begin{bmatrix} 3 & 1 \\ 2 & 3 & 5 \\ 2 & 5 \end{bmatrix} \]

- Collaborative filtering: Similar users perform similarly in future

Shallow representation learning
- Matrix factorization & factorization machines

Neural representation learning
- Neural collaborative filtering
- Graph neural networks
- Sequential model
- Textual & Visual encoders

Learning correlations between input features and interactions.
Shortcomings of Data-driven RecSys

• Correlation != preference: Correlations may not reflect the true causes of interactions.

• Three basic types of correlations:
  • Causation
    • Stable and explainable
  • Confounding
    • Ignoring X
    • Spurious correlation
  • Collision
    • Condition on S
    • Spurious correlation
Shortcomings of Data-driven RecSys

• Data-driven methods will learn skewed user preference:

True preference distribution on testing data

Biases (Confounding, Collision)

Skewed preference distribution exhibited on training data (With spurious correlations)

• Data-driven methods may infer spurious correlations, which deviates from users’ true preference.

Correlation ≠ preference
Why Causal Inference?

- Aim: Understanding the inherent **causal mechanism** behind user behaviors
  - Capturing user true preference

- Making reliable & explainable recommendations
  - Correlation + Causality > Correlation
Classification of Causal Recommendation

- Structural Causal Model (SCM)
  - Judea Pearl
  - Donald B. Rubin

- Potential Outcome Framework
  - Judea Pearl
  - Donald B. Rubin

Evaluation
Debiasing
Explanation
Recommendation
Fairness
Robustness & OOD generalization
Outline

• Part 1 (90 min, 9:00—10:30)
  • Introduction (Wenjie Wang, 15 min)
  • Structural causal models for recommendation (Yang Zhang and Wenjie Wang, 60~70 min)
  • Q&A (5 min)
  • Coffee break (30 min)

• Part 2 (90 min, 11:00-12:30)
  • Potential outcome framework for recommendation (Haoxuan Li and Peng Wu, 60~70 min)
  • Comparison (Fuli Feng, 2 min)
  • Conclusion, open problems, and future directions (Fuli Feng, 20 min)
  • Q&A (5 min)
• How can common understandings, such as the fact that symptoms do not cause diseases, be expressed mathematically?

\[
X = U_X \\
Y = \beta X + U_Y
\]

\(U_X\) and \(U_Y\): exogenous

Causal diagrams encodes causal assumption via missing arrows, representing claims of zero influences

• General form:

\[
Z = f_Z(U_Z) \\
X = f_X(Z, U_X) \\
Y = f_Y(X, U_Y)
\]

Structural Causal Model

- Basic causal structures in causal graph

**Chain**
- $Z$: mediator
- $X$ and $Y$ are associated.
- Condition on $Z$, $X$ and $Y$ are independent.

**Confounding**
- $Z$: confounder
- $X$ does not affect $Y$, but $X$ and $Y$ are correlated. (Spurious correlations).
- Condition on $Z$, $X$ and $Y$ are independent, blocking the spurious correlations.

**Colliding**
- $Z$: collider
- $X$ and $Y$ are independent.
- Condition on $Z$, $X$ and $Y$ are correlated, bringing spurious correlations.
• Correlation is not causation

Confounders and controlling colliders would bring spurious correlations between treatment and outcome.

It is impossible to answer causal question with correlation-level tools

• do-calculus

It provides various principles to identify target causal effect.
For example, utilize the backdoor adjustment when confounders exist

If any node in Z isn’t a descendant of X, and Z blocks every path between X and Y that contains an arrow into X (backdoor path), then the average causal effect of X on Y is:

\[ P(Y|do(X)) = \sum_Z P(Y|X,Z)P(Z)\]
SCM provides both a mathematical foundation and a friendly calculus for the analysis of causal effects and counterfactuals. It can deal with the estimation of three types of causal queries:

- Queries about the effect of potential interventions. To compute causal effect, e.g., \( P(Y|do(X)) \)
- Queries about counterfactuals. e.g., whether event A would occur if event B had been different?
- Queries about the direct / indirect effects. (based on counterfactuals)

\[ X \rightarrow Y \]

the direct effects of \( X \) on \( Y \):

\[ X \rightarrow Z \rightarrow Y \]

the indirect effects of \( X \) on \( Y \):

© Copyright National University of Singapore. All Rights Reserved.
SCM for Recommendation

Q1: Queries about causal effect.

Q2: Queries about counterfactuals.

Q3: Queries about the direct/indirect effects.

Deal with confounding/colliding

De-biasing via deconfounding
- Observed confounding bias
- Unobserved confounding bias

Utilize colliding structure
- Disentangle
- Model retraining

Counterfactual inference:
- (in)direct effect for debiasing
- data augmentation
- fairness
- explanation

Causal queries

Recommendation

answer counterfactual questions
SCM for Recommendation

• Dealing with confounding structures in recommendation (Yang Zhang)
  • Confounding in recommendation.
  • Deal with observed confounders.
  • Deal with unobserved confounders.

• Considering colliding structures in recommendation (Yang Zhang)
  • Colliders in recommendation
  • Modeling the colliding effect

• Counterfactual recommendation (Wenjie Wang)
  • Counterfactual inference for debiasing
  • Counterfactual inference against filter bubbles
  • Counterfactual data synthesizing
  • Counterfactual fairness
  • Counterfactual explanation
  • Causal modeling for OOD generalization
Confounders in Recommendation

- Are there confounders in recommendation?
  - some examples

- What’s more, some confounder are observable/measurable, some confounder are unobservable/unmeasurable. e.g., company is measurable, quality is unmeasurable.
Confounders in Recommendation

- Is it necessary to deal with confounding effects?
  - The goal of recommendation: estimate user preference. But user preference is implicit.
  - We estimate it as $P(Y|U,I)$, i.e., taking the correlations between $(U,I)$ pair and click $Y$ as the preference.

- However, when there are confounders between $U/I$ and $Y$ (red line), the confounding effect will also bring correlations, while it cannot reflect user preference.

Thus, it is essential to deal with the confounding problem in recommendation!

But HOW?
The backdoor adjustment is an obvious solution in this line of research.

Existing Work Regarding Observed Confounders

2021 SIGIR&KDD&CIKM
- Zhang et.al. PDA
- Wang et.al. DecRS
- Yang et.al. DCM
- Gupta et.al. CauSeR

2022 TKDE&KDD&CIKM&SIGIR
- Wang et.al. CaDSI
- Zhan et al. D2Q
- He et al. CISD
- Rajanala et al. DeSCoVeR

2023 TOIS
- He et.al. DCR
- Zhang et.al. DML

The above work considers different problems caused by confounders, and uses different strategies to implement the backdoor adjustment.
PDA: Confounding View of Popularity Bias

- Popularity bias
  - **Favor a few popular items** while not giving deserved attention to the majority of others
  - The popular items are recommended even more frequently than their popularity would warrant, **amplifying** long-tail effects.

- Previous methods ignore the underline causal mechanism and blindly remove bias to purchase an even distribution.

- But, **not all popularity biases data are bad**.
  - Some items have higher popularity because of better quality.
  - Some platforms have the need of **introducing desired bias** (promoting the items that have the potential to be popular in the future).
PDA: Confounding View of Popularity Bias

- What is the bad effect of popularity bias?
  - Traditional causal assumption
    - \((U, I) \rightarrow C\): user-item matching affects click.
    - Item popularity also has influence on the recommendation process, but is not considered.
  - Cofounding view
    - \(Z \rightarrow I\): Popularity affects item exposure.
    - \(Z \rightarrow C\): Popularity affects click probability.
    - \(Z\) is a confounder, bringing spurious (bad effect) correlation between \(I\) and \(C\).
    - Take the causation \(P(C|do(U, I))\), instead of the correlation \(P(C|U, I)\), as user preference.

Causation (backdoor adjustment):
\[ P(C|do(U, I)) = \sum_Z P(C|U, I, Z)P(Z) \]

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

\[ \propto \sum_Z P(C|U, I, Z)P(I|Z)P(Z) \]

Bad effect

\(U\): user; \(I\): exposed item; \(C\): interaction label

Zhang et al. SIGIR 2021. Causal Intervention for Leveraging Popularity Bias in Recommendation
PDA: Confounding View of Popularity Bias

- **Training & Inference**: Popularity De-confounding (PD, remove bad effect)

- **To estimate** \( P(C|do(U, I)) = \sum_z P(C|U, I, z)P(z) \):
  - **Step 1.** Estimate \( P(C|U, I, Z) \)
    - \( P_{\theta}(c = 1|u, i, m_i^t) = f_{\theta}(u, i) \times m_i^t \)
    - \( m_i^t \) the popularity of item \( i \) in timestamp \( t \)
    - Learn with traditional loss
  - **Step 2.** Compute \( P(C|do(U, I)) \)
    - \( \sum_Z P(C|U, I, Z)P(Z) \propto f_{\theta}(u, i) \)
    - Derivation sees the paper

- **Another Inference**: Popularity Adjusting (inject desired popularity bias)
  - Inject the desired pop bias \( \tilde{Z} \) by causal intervention

\[
P(C|do(U, I), do(Z = \tilde{Z})) \quad \iff \quad f_{\theta}(u, i) \times \tilde{m}_i
\]
DCR: Deconfounding for Solving Unreliable Label Issue

- **Unreliable label issue:**
  - No ground-truth label for the prediction objective — user preference
  - Only have indirect label: user behaviors

- **Causal Modeling:**
  - Traditional assumption: U-I matching affect label
  - Some item feature directly affect the label

DCR: Deconfounding for Solving Unreliable Label Issue

Causal analyses

- **direct path** $A \rightarrow Y$: make $P(Y|X, A)$ biased towards short videos
- **Backdoor path** $X \leftarrow Z \rightarrow A \rightarrow Y$: make $P(Y|X)$ learn spurious correlation

Should beyond correlation-level

Causal effect as interest

true user preference: the **causal effects** path through $M$ to $Y$

$$P(Y|U, do(X)) = \sum_{a \in \mathcal{A}} P(Y|U, X, A = a)P(A = a),$$
DCR: Deconfounding for Solving Unreliable Label Issue

How to estimate the causal effect?

\[ P(Y|U, do(X)) = \sum_{a \in \mathcal{A}} P(Y|U, X, A = a)P(A = a), \]

• DCR: model-based estimation

\[ k^{th} \text{ expert: } P(y = 1|u, x, A = a_K) \]

◆ Training --- fitting \( P(Y|U, X, A) \)

◆ Inference --- backdoor adjustment

• DCR involves changing the model architecture, DML [2] proposes to achieve the adjustment directly at the label level/

DecRS: Alleviating Bias Amplification

- Bias amplification:
  - What is it?
  - Why?

- Over-recommend items in the majority group

- An item with low rating receives a higher prediction score because it belongs to the majority group.
- Intuitively, we can know that the user representation shows stronger preference to majority group.

Wang et al. SIGKDD 2021. Deconfounded recommendation for alleviating bias amplification.
DecRS: Alleviating Bias Amplification

- **Causal view of bias amplification**

  \[ D: \text{user historical distribution over item group.} \quad d_u = [p_u(g_1), ..., p_u(g_N)], \quad \text{e.g.,} \quad d_u = [0.8, 0.2]. \]

  \[ M: \text{describe how much the user likes different item groups, decided by} \ D \ \text{and} \ U. \]

  \[ (U, M) \rightarrow Y: \quad \text{an item} \ i \ \text{can have a high} \ Y \ \text{because:} \]

  - 1) the user’s pure preference over the item \ (U \rightarrow Y), \text{or}
  - 2) the user shows interest in the item group \ (U \rightarrow M \rightarrow Y). \]

  ✓ \( D \) is a confounder between \( U \) and \( Y \), bringing spurious correlations: given the item \( i \) in a group \( g \), the more superior \( g \) is in \( u \)’s history, the higher the prediction score \( Y \) becomes.

- **Backdoor adjustment**

  \[
  P(Y|U = u, I = i) = \frac{\sum_{d \in D} \sum_{m \in M} P(d)P(u|d)P(m|d, u)P(i)P(Y|u, i, m)}{P(u)P(i)} \quad (1a)
  
  = \sum_{d \in D} \sum_{m \in M} P(d|u)P(m|d, u)P(Y|u, i, m) \quad (1b)
  
  = \sum_{d \in D} P(d|u)P(Y|u, i, M(d, u)) \quad (1c)
  
  = P(d_u|u)P(Y|u, i, M(d_u, u)) \quad (1d)
  
  P(Y|do(U = u), I = i) = \sum_{d \in D} P(d)P(Y|do(U = u), i, M(d, do(U = u))) \quad (2a)
  
  = \sum_{d \in D} P(d)P(Y|do(U = u), i, M(d, do(U = u))) \quad (2b)
  
  = \sum_{d \in D} \boxed{P(d)P(Y|U = u, i, M(d, u))} \quad (2c)
  
© Copyright National University of Singapore. All Rights Reserved.

Wang et al. SIGKDD 2021. Deconfounded recommendation for alleviating bias amplification.
• Deconfounded Recommender System (DecRS)
  • To implement:

\[
P(Y|do(U = u), I = i) = \sum_{d \in D} P(d)P(Y|u, i, M(d, u))
\]  

**Challenge**: the sample space of \( D \) is infinite.

• Backdoor adjustment approximation:
  (1) Sampling distributions to represent \( D \);
  Use function \( f(\cdot) \) (FM) to calculate \( P(Y|u, i, M(d, u)) \).

\[
P(Y|do(U = u), I = i) \approx \sum_{d \in D} P(d)P(Y|u, i, M(d, u)) \approx \sum_{d \in D} P(d)f(u, i, M(d, u))
\]

(2) Approximation of \( E_d[f(\cdot)] \).
  • Expectation of function \( f(\cdot) \) of \( d \) in Eq. 4 is hard to compute because we need to calculate the results of \( f(\cdot) \) for each \( d \).
  • **Jensen’s inequality**: take the sum into the function \( f(\cdot) \).

\[
P(Y|do(U = u), I = i) \approx f(u, i, M(\sum_{d \in D} P(d)u)).
\]

Different to PDA, this term directly represents the target casual effect.

Wang et al. SIGKDD 2021. Deconfounded recommendation for alleviating bias amplification.
The methods based on backdoor adjustment need the confounders to be observable and controllable.
However, unobserved/unmeasurable/uncontrollable confounders exist in recommendation. How to deal with them?

There are two lines of work:

- **Front-door adjustment**
  - 2023 TKDE & TORS
    - Zhu et al. CausalD
    - Xu et al. DCCF
  - 2022 ArXiv
    - Zhu et al. HCR

- **Learning substitutes**
  - 2023 KDD
    - Zhang et al. iDCF
  - 2022 ArXiv
    - Zhu et al. Deep-Deconf
  - 2020 NeurIPS
    - Wang et al. DCF
  - 2020 RecSys
    - Zhou et al. VSR
Source of confounding bias is the **confounder** that affects item attributes and user feedback simultaneously.

Some confounders are hard to measure:
- Technical difficulties, privacy restrictions, *etc.*
- E.g., product quality.

Removing hidden confounders is hard:
- Inverse Propensity Weighting
  - Based on strict assumption of no hidden confounder.
- Backdoor Adjustment
  - Require the confounder's distribution.

---

**Reverse Correlation**

- Positive Ratings
  - High Price
  - Spurious correlations

---

HCR: Front-door Adjustment-based Solution

• Abstract user feedback generation process into causal graph.
  • $V$: hidden confounder; $L$: like feedback; $I$: item; $U$: user.
  • $M$: a set of variables that act as mediators between $\{U, I\}$ and $L$, e.g., user-item feature matching, and click.

• Key:
  • Block the backdoor path $I \leftarrow V \rightarrow L$
  • Estimate the causal effect of $I$ on $L$, i.e.,
    $$P(L|U, \text{do}(I)).$$

• Hidden Confounder Removal (HCR) framework.
  • Front-door adjustment
    • decompose causal effect of $I$ on $L$ into: 1) the effects of $I$ on $M$ and 2) the effect of $M$ on $L$.

\[
P(L|U, \text{do}(I)) = \sum_M P(M|U, \text{do}(I))P(L|U, \text{do}(M))
= \sum_M P(M|U, I) \sum_{I'} P(I')P(L|M, U, I')
\]
HCR: Front-door Adjustment-based Solution

- **Hidden Confounder Removal (HCR) framework**
  - \( P(L|do(I), U) = \sum_M P(M|U, I) \sum_{I'} P(I')P(L|U, I', M) \)
  - Multi-task learning
    - Learns \( P(M|U, I) : = f_m(U, I) \)
    - Learn
      \[
      P(L|M, U, I) : = h(U, I, M) \\
      = h^1(U, M)h^2(U, I')
      \]

- **Inference**
  - Infer \( P(M|U, I) \) and \( P(L|U, I, M) \)
  - Get rid of the sum over \( I \) and obtain
    \[
    P(L|U, do(I)) \\
    = \sum_M f_m(U, I) \sum_{I'} P(I') h^1(U, M) h^2(U, I') \\
    = \sum_M f_m(U, I) h^1(U, M) \sum_{I'} P(I') h^2(U, I') \\
    = S_u \sum_M f_m(U, I) h^1(U, M)
    \]
CausalD: Front-door Adjustment-based Solution

- Consider Hidden Confounder in Sequential Recommendation

Sequential recommendation: predict user next behavior using historical behaviors

X: historical interaction  Y: Next behavior  M: Representations  U: unobserved confounder, such as social relationships

\[
P(Y|do(X)) = \sum_m P(m|do(X))P(Y|do(m)) \\
= \sum_m P(m|X) \sum_{x'} P(X = x')P(Y|m, x')
\]

- Estimation method: similar to HCR but in a distillation manner

Learning Substitutes-based Solution

- Multiple causes assumption for recommendation:
  - multiple causes: each user's binary exposure to an item $a_{ui}$ is a cause(treatment), thus there are multiple causes.
  - There are **multiple-cause confounders** (confounders that affect ratings and many causes).
  - Single-cause confounders (confounders that affect ratings and only one cause) are negligible.

Learning Substitutes-based Solution

• Learning substitutes to deconfounding:

Key: if $Z_u$ renders the $a_{u,i}$’s conditionally independent then there cannot be another multi-cause confounder

Contradiction: assume $p(a_{u1}, ..., a_{um}|z_u) = \prod_i p(a_{ui}|z_u)$, if there is a multi-cause confounder, the conditional independence cannot hold.

• Step 1: learning substitutes

Finding a $Z_u$, such that:
$$p(a_{u1}, ..., a_{um}|z_u) = \prod_i p(a_{ui}|z_u)$$

Example:
find a generative model:
$$p_\theta(A_u|Z_u) = \prod_{i=1}^m \text{Bern}(a_{ui}|\theta(z_u)_i)$$
then:
find $q_\Phi(Z_u|A_u)$ with variation-inference

• Step 2: deconfounded recommender

Control the substitutes to fit recommender model

Example:
$$y_{ui}(a) = \theta_u^\top \beta_i \cdot a + \gamma_u \cdot z_{ui} + \epsilon_{ui}$$
where $\theta_u$ and $\beta_i$ refer user preference and item attributes, respectively.
Papers on Deconfounding Recommendation

- Gupta, Priyanka, et al. "CauSeR: Causal Session-based Recommendations for Handling Popularity Bias." In CIKM 2021. (Gupta et.al., CauSeR)
- Rajanala, Sailaja, et al. "Discover: Debiased semantic context prior for venue recommendation." In SIGIR 2022 (Rajanala et al. DeSCoVeR)
- Yang, Xun, et al. "Deconfounded video moment retrieval with causal intervention." In SIGIR 2021. (Yang et.al. DCM)
- He, Ming, et al. "Causal intervention for sentiment de-biasing in recommendation." In CIKM 2022. (He et al. CISD)
- Qing Zhang et.al. Debiasing Recommendation by Learning Identifiable Latent Confounders. KDD 2023. (Zhang et al. iDCF)
SCM for Recommendation

- Dealing with confounding structures in recommendation (Yang Zhang)
  - Confounding in recommendation.
  - Deal with observed confounders.
  - Deal with unobserved confounders.

- Considering colliding structures in recommendation (Yang Zhang)
  - Colliders in recommendation
  - Modeling the colliding effect

- Counterfactual recommendation (Wenjie Wang)
  - Counterfactual inference for debiasing
  - Counterfactual inference against filter bubbles
  - Counterfactual data synthesizing
  - Counterfactual fairness
  - Counterfactual explanation
  - Causal modeling for OOD generalization
Colliding Effects in Recommendation

• Are there colliders in recommendation?
  • There are variables affected by many factors. Such as, the happening of clicking is affected by user preference and the exposure position.
  • Existing work also tries to construct colliders manually.

• To utilize or eliminate colliding effects?
  • Assume that we have known $X_2$, try to estimate $X_1$.
  • Condition on $Z$, $X_1$ and $X_2$ could be correlated.
  • That means condition on $Z$, $X_2$ would provide us more information to estimate $X_1$.

In recommendation, we usually face with this case (know $X_2$ and $Z$ to predict $X_1$). Thus existing work based on SCM tries to utilize colliding effects to better learn some targets.
What are causes of a user-item interaction (click)?

Two main causes:
• Interest
• Conformity
User tend to follow the mainstream

Disentangle Interest and Conformity to identify true interest.

But it is hard because of lacking ground-truth. (An interaction can come from either factor or both factors)

Colliding effect can come to help:

- Interest and Popularity (conformity) are independent
- But, they are correlated given clicks:
  A click on a less popular item \(\rightarrow\) High Interest
DICE: Partial pairwise data identifies true interest:

- $O_1$: \{<u, pos\_item, neg\_item>, wherein pos\_item is less popular than neg\_item\}
- Pairwise cause-specific data (interest-driven): we can ascertain that the interaction is more likely due to user interest

Key 1: split user/item representation into two embeddings

Key 2: learning interest embedding on interest-driven pairwise data ($O_1$).

The core idea of leveraging colliding effects has also been extended to Sequential Recommendation. (Sun et al. MiceRec. 2022.)
Colliding Effects for Incremental Training

• Incremental training for recommender system
  • Usually, using the incremental interaction data $I_t$ for efficient retraining.
  • Only updating the representations of active user/item corresponding to $I_t$.
  • Ignoring the representations of inactive user/item.

Causal graph of incremental training

- $R_{In,t-1}$: Representations of inactivate user/item at time $t-1$.
- $R_{In,t}$: Representations of inactivate user/item at time $t$.
- $R_{Ac,t-1}$: Representations of activate user/item at time $t-1$.
- $R_{Ac,t}$: Representations of activate user/item at time $t$.
- $I_t$: Incremental interaction data collected from time $t-1$ to $t$. 

Colliding Effects for Incremental Training

- Causal incremental training with colliding effects

Creating a collider $S_t$ between $R_{In,t}$ and $R_{Ac,t}$, $S_t$ is the similarity between representations of active and inactivate user/item.

Restraining $S_t = S_{t-1}$ to open the causal path $I_t \rightarrow R_{Ac,t} \rightarrow R_{In,t}$ with the help of colliding effect.

Using the incremental data $I_t$ simultaneously update both $R_{Ac,t}$ and $R_{In,t}$.
SCM for Recommendation

• Dealing with confounding structures in recommendation (Yang Zhang)
  • Confounding in recommendation.
  • Deal with observed confounders.
  • Deal with unobserved confounders.

• Considering colliding structures in recommendation (Yang Zhang)
  • Colliders in recommendation
  • Modeling the colliding effect

• Counterfactual recommendation (Wenjie Wang)
  • Counterfactual inference for debiasing
  • Counterfactual inference against filter bubbles
  • Counterfactual data synthesizing
  • Counterfactual fairness
  • Counterfactual explanation
  • Causal modeling for OOD generalization
Counterfactual Recommendation

• Counterfactual inference for debiasing
  o Focus on **removing path-specific effects** for debiasing
  o First estimate the causal effect by comparing a counterfactual world with the factual world, and then mitigate path-specific effects.

• Representative Work
Clickbait bias

- User interactions are biased to the items with attractive exposure features.
- Clickbait items: exposure features (e.g., title/cover image) attract users while content features (e.g., video) are disappointing.
- Recommender models learned from the biased interactions will frequently recommend these clickbait items, decreasing user experience.

Fig. Statistics of clicks and likes on Tiktok dataset. Partly show the wide existence of clickbait issue.
Counterfactual for Mitigating Clickbait Bias

• **Counterfactual Inference**

  ❖ **Causal Graph**
  • A causal graph to describe the causal relationships between the features and user-item prediction scores.
  • **Reason for clickbait issue**: $E \rightarrow Y$ a clickbait item has high prediction scores purely due to its attractive exposure features, *i.e.*, title/cover.

  ❖ **Causal learning** for training: learn structural functions $I(E,T)$ and $Y(U,I,E)$ from data.

  ❖ **Causal reasoning** for inference: counterfactual inference.
  • Reduce the direct effect of exposure features.
  • 1) **Estimate** the effect in the counterfactual world, which imagines *what the prediction score would be if the item had only the exposure features*.
  • 2) **Reduce** the direct effect of exposure features for inference.

---

Wang et al. *Clicks can be cheating: Counterfactual recommendation for mitigating clickbait issue*. In SIGIR 2021.
Counterfactual for Mitigating Clickbait Bias

• Overall Performance

Table 2: Top-K recommendation performance of compared methods on Tiktok and Adressa. %Improve. denotes the relative performance improvement of CR over NT. The best results are highlighted in bold. Stars and underlines denote the best results of the baselines with and without using additional post-click feedback during training, respectively.

<table>
<thead>
<tr>
<th>Dataset Metric</th>
<th>Tiktok</th>
<th>Adressa</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P@10</td>
<td>R@10</td>
</tr>
<tr>
<td>NT [50]</td>
<td>0.0256</td>
<td>0.0357</td>
</tr>
<tr>
<td>CFT [50]</td>
<td>0.0253</td>
<td>0.0356</td>
</tr>
<tr>
<td>IPW [27]</td>
<td>0.0230</td>
<td>0.0334</td>
</tr>
<tr>
<td>CT [50]</td>
<td>0.0217</td>
<td>0.0295</td>
</tr>
<tr>
<td>NR [51]</td>
<td>0.0239</td>
<td>0.0346</td>
</tr>
<tr>
<td>RR</td>
<td>0.0264*</td>
<td>0.0383*</td>
</tr>
<tr>
<td>CR</td>
<td>0.0269</td>
<td>0.0393</td>
</tr>
<tr>
<td>%Improve.</td>
<td>5.08%</td>
<td>10.08%</td>
</tr>
</tbody>
</table>

%Improve. CR significantly recommends more satisfying items by mitigating clickbait bias.

Wang et al. Clicks can be cheating: Counterfactual recommendation for mitigating clickbait issue. In SIGIR 2021.
Popularity Bias in RecSys

• Popularity bias $\neq$ Uneven popularity distribution
  • The popular items are gradually over-recommended, amplifying long-tail effects.
  • Favor a few popular items while not giving deserved attention to the majority of others.

• From data perspective:

  ![Graph](image)

  Long-tail distribution

  - 67% of items account for 20% of click count in Kwai.
  - 86% of items account for 20% of click count in Douban.
**Counterfactual for Mitigating Popularity Bias**

**Causal View of Popularity Bias**

- **Edge** $I \rightarrow R$ captures popularity bias.
- **Edge** $U \rightarrow R$ captures the user sensitive to popularity.

**Solution:**
- Train a recommender based on the causal graph via a multi-task learning.
- Perform **counterfactual inference** to eliminate popularity bias (*Question to answer: what would the prediction be if there were only popularity bias?*)

---

Counterfactual Inference to Remove Bias

**Question:** what the prediction would be if there were no bias?

\[
\text{TIE} = TE - NDE = Y(U = u, I = i, K = K_{u,i}) - Y(U = u, I = i, K = K^*_{u,i})
\]

Inference with
\[
\text{TIE} = \hat{y}_k \times \sigma(\hat{y}_i) \times \sigma(\hat{y}_u) - c \times \sigma(\hat{y}_i) \times \sigma(\hat{y}_u)
\]
Counterfactual for Mitigating Popularity Bias

- Evaluate MACR framework on two base models: MF and LightGCN.
- Testing data is intervened to be uniform.

<table>
<thead>
<tr>
<th>Method</th>
<th>Adressa</th>
<th>Yelp2018</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Recall</td>
<td>NDCG</td>
</tr>
<tr>
<td>MF</td>
<td>0.0853</td>
<td>0.0341</td>
</tr>
<tr>
<td>ExpoMF</td>
<td>0.0896</td>
<td>0.0365</td>
</tr>
<tr>
<td>MF_causE</td>
<td>0.0835</td>
<td>0.0365</td>
</tr>
<tr>
<td>MF_BS</td>
<td>0.0900</td>
<td>0.0377</td>
</tr>
<tr>
<td>MF_reg</td>
<td>0.0659</td>
<td>0.0332</td>
</tr>
<tr>
<td>MF_IPS</td>
<td>0.0964</td>
<td>0.0392</td>
</tr>
<tr>
<td>MACR</td>
<td>0.1090</td>
<td>0.0495</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Adressa</th>
<th>Yelp2018</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Recall</td>
<td>NDCG</td>
</tr>
<tr>
<td>Lgcn</td>
<td>0.0977</td>
<td>0.0395</td>
</tr>
<tr>
<td>Lgcn_causE</td>
<td>0.0823</td>
<td>0.0374</td>
</tr>
<tr>
<td>Lgcn_BS</td>
<td>0.1085</td>
<td>0.0469</td>
</tr>
<tr>
<td>Lgcn_reg</td>
<td><strong>0.0979</strong></td>
<td><strong>0.0390</strong></td>
</tr>
<tr>
<td>Lgcn_IPS</td>
<td>0.1070</td>
<td>0.0468</td>
</tr>
<tr>
<td>MACR</td>
<td>0.1273</td>
<td>0.0525</td>
</tr>
</tbody>
</table>

MF as the backbone

LightGCN as the backbone

• Evaluate MACR framework on two base models: MF and LightGCN.
• Testing data is intervened to be uniform.
Counterfactual for Leveraging Popularity Bias

- **Conflicting Observation:**
  - The more popular an item is, the larger average rating value the item tends to have (positive correlation).
  - From the temporal view, for a large proportion of items, the rating value exhibits negative correlation with the item popularity at that time.

- Quality + Conformity $\rightarrow$ Popularity, thus disentangle benign and harmful Bias.
Counterfactual for Leveraging Popularity Bias

- **Time-aware DisEntangled framework (TIDE)**
  - Main challenge: Lack of explicit signal for disentanglement

- **Quality is static**: $I \rightarrow Q \rightarrow Y$
  - Quality has **stable** influence on users’ behavior

- **Conformity is dynamic**: $(I, t) \rightarrow C \rightarrow Y$
  - Conformity is **time-sensitive**

- **User interest**: $(U, I) \rightarrow M \rightarrow Y$
  - User and item’s matching score, can be implemented by various recommendation models, such as MF, LightGCN, etc.

(a) Causal graph of our TIDE.

- **U**: User
- **I**: Item
- **t**: time
- **C**: conformity
- **Q**: Quality
- **Y**: Prediction
- **M**: Matching score

© Copyright National University of Singapore. All Rights Reserved. Zhao et al. Popularity Bias is not Always Evil: Disentangling Benign and Harmful Bias for Recommendation. TKDE’ 22.
Counterfactual for Leveraging Popularity Bias

- **Training Stage:**
  - Popularity comes from Quality and Conformity
  - Prediction with Popularity and matching score
    \[
    \hat{y}_{ui} = \tanh(q_i + c_i^t) \times \text{Softplus}(m_{ui})
    \]

- **Inference Stage:**
  - Intervention: set \( c \) as reference vector \( c^* \) (e.g., zero) during inference to remove the improper effect from \( C \) to \( Y \).
    \[
    \hat{y}_{ui} = \tanh(q_i + c^*) \times \text{Softplus}(m_{ui})
    \]

- **Comparison with PD**
  - TIDE further conduct disentanglement of popularity bias

Zhao et al. Popularity Bias is not Always Evil: Disentangling Benign and Harmful Bias for Recommendation. TKDE’ 22.
SCM for Recommendation

• Dealing with confounding structures in recommendation (Yang Zhang)
  • Confounding in recommendation.
  • Deal with observed confounders.
  • Deal with unobserved confounders.

• Considering colliding structures in recommendation (Yang Zhang)
  • Colliders in recommendation
  • Modeling the colliding effect

• Counterfactual recommendation (Wenjie Wang)
  • Counterfactual inference for debiasing
  • Counterfactual inference against filter bubbles
  • Counterfactual data synthesizing
  • Counterfactual fairness
  • Counterfactual explanation
  • Causal modeling for OOD generalization
Counterfactual Recommendation

• Counterfactual for Alleviating Filter Bubbles
  o Filter bubbles in recommendation: RecSys emphasizes only a small set of items in the feedback loop.
  o Similar concepts: echo chamber, information cocoon.
  o Build causal models to interactive with users.

• Representative Work
Filter bubbles in recommendation: continually recommending many homogeneous items, isolating users from diverse contents.

Solution: let users control the filter bubbles by directly adjusting recommendations.

Two-level user controls regarding either a user or item feature.

- Fine-grained level: increase the items w.r.t. a specified user or item feature.
  - For example, “more items liked by young users”.
- Coarse-grained level: no need to specify the target user/item group.
  - For example, “no bubble w.r.t. my age”

A counterfactual imagination

- Real-time response to user controls.
- Need to reduce the effect of historical user representations.
- Counterfactual inference to mitigate the effect of out-of-date user interactions.
Counterfactual for handling filter bubbles

- Propose an unbiased causal user model $\phi_M$ in the model-based offline reinforcement learning (RL) framework to disentangle the intrinsic user interest from the overexposure effect of items.

Counterfactual IRS (CIRS) based on offline RL learning

- Utilize counterfactual inference to disentangle and reduce the over-exposure effect on some items

Sample data from historical policies

Buffer $\mathcal{D}$

Learn $\phi_M$: the causal user model

Plan $\pi_\theta$: the RL policy

Save interaction data of policy $\pi_\theta$: $\{(u, i, r, t)\}$

SCM for Recommendation

• Dealing with confounding structures in recommendation (Yang Zhang)
  • Confounding in recommendation.
  • Deal with observed confounders.
  • Deal with unobserved confounders.

• Considering colliding structures in recommendation (Yang Zhang)
  • Colliders in recommendation
  • Modeling the colliding effect

• Counterfactual recommendation (Wenjie Wang)
  • Counterfactual inference for debiasing
  • Counterfactual inference against filter bubbles
  • Counterfactual data synthesizing
  • Counterfactual fairness
  • Counterfactual explanation
  • Causal modeling for OOD generalization
Counterfactual Recommendation

• Counterfactual data synthesis for alleviating data sparsity
  o Generate counterfactual interaction sequences for sequential recommendation.
  o Simulate the recommendation process and generate counterfactual samples, including recommendations and user feedback.

• Representative work
Counterfactual Data Synthesis

• Counterfactual data synthesis
  - Generate counterfactual interaction sequences for sequential recommendation.


Counterfactual Data Synthesis

- Counterfactual data synthesis
  - Simulate the recommendation process and generate counterfactual samples, including recommendations and user feedback.
    1) Learn SCM from observed data to simulate the recommendation process.
    2) Conduct intervention on the recommendation list (R) to generate counterfactual samples.
    3) Use observed and generated data to train the ranking model.

---

SCM for Recommendation

• Dealing with confounding structures in recommendation (Yang Zhang)
  • Confounding in recommendation.
  • Deal with observed confounders.
  • Deal with unobserved confounders.

• Considering colliding structures in recommendation (Yang Zhang)
  • Colliders in recommendation
  • Modeling the colliding effect

• Counterfactual recommendation (Wenjie Wang)
  • Counterfactual inference for debiasing
  • Counterfactual inference against filter bubbles
  • Counterfactual data synthesizing
  • Counterfactual fairness
  • Counterfactual explanation
  • Causal modeling for OOD generalization
Counterfactual Fairness

- Pursue fair recommendation for the users with different **sensitive attributes** (e.g., age and gender).
- Counterfactual fair recommendation.
- Use **adversarial learning** to remove the sensitive information from user embedding ($r_u$).

**Definition 1 (Counterfactually fair recommendation).** A recommender model is counterfactually fair if for any possible user $u$ with features $X = x$ and $Z = z$:

$$P(L_Z | X = x, Z = z) = P(L_Z' | X = x, Z = z)$$

for all $L$ and for any value $z'$ attainable by $Z$, where $L$ denotes the Top-N recommendation list for user $u$.

- $X_u$ and $Z_u$ are insensitive and sensitive features of the user $u$.
- $H_u$ is the user interaction history.
- $r_u$ is the user embedding.
- $C_u$ is the candidate item set for $u$.
- $S_u$ are the predicted scores over the candidate items.

Counterfactual Fairness

- **Path-specific (PS) counterfactual fairness**
  - PS fair recommendation
    - eliminate the unfair influences of sensitive features (e.g., race)
    - preserve fair influences of sensitive features (e.g., chopsticks for East-Asian users).
  - Calculate and remove PS bias based on path-specific counterfactual inference.

\[
\begin{align*}
\text{PSBias}(x, s, s') &= \mathbb{E}[R_{S \leftarrow s}(U_{f \leftarrow s}, U_{b \leftarrow s}, s') | X = x, S = s] \\
&\quad - \mathbb{E}[R_{S \leftarrow s}(U_{f \leftarrow s}, U_{b \leftarrow s}, s') | X = x, S = s]
\end{align*}
\]

SCM for Recommendation

• Dealing with confounding structures in recommendation (Yang Zhang)
  • Confounding in recommendation.
  • Deal with observed confounders.
  • Deal with unobserved confounders.

• Considering colliding structures in recommendation (Yang Zhang)
  • Colliders in recommendation
  • Modeling the colliding effect

• Counterfactual recommendation (Wenjie Wang)
  • Counterfactual inference for debiasing
  • Counterfactual inference against filter bubbles
  • Counterfactual data synthesizing
  • Counterfactual fairness
  • Counterfactual explanation
  • Causal modeling for OOD generalization
Counterfactual Explanation

- Generate explanation by counterfactual thinking.
- Find the **minimal changes** that lead to a different recommendation.
- Identify the most critical features causing the recommendations.


SCM for Recommendation

• Dealing with confounding structures in recommendation (Yang Zhang)
  • Confounding in recommendation.
  • Deal with observed confounders.
  • Deal with unobserved confounders.

• Considering colliding structures in recommendation (Yang Zhang)
  • Colliders in recommendation
  • Modeling the colliding effect

• Counterfactual recommendation (Wenjie Wang)
  • Counterfactual inference for debiasing
  • Counterfactual inference against filter bubbles
  • Counterfactual data synthesizing
  • Counterfactual fairness
  • Counterfactual explanation
  • Causal modeling for OOD generalization
Counterfactual Recommendation

- Causal Modeling for OOD Recommendation
  - The interaction distribution is shifting over time in recommendation.
  - Leverage causal modeling to enhance the recommender generalization.

- Representative Work
Causal Modeling for OOD Recommendation

- User preference is shifting over time.

- **Reason** of the preference shifts: *change of user features.*
  - User features $\rightarrow$ preference $\rightarrow$ interactions.

- Explore OOD recommendation under two settings:
  - OOD recommendation with **observed user features**. (*e.g.*, increased consumption levels and changed location)
  - OOD recommendation with **unobserved user features**. (*e.g.*, friend recommendations, hot event, and context factors)

Causal Modeling for OOD Recommendation

- **OOD recommendation with observed user features.**
  1) Figure out the mechanism how feature shifts affect user preference.
     - User features $\rightarrow$ preference $\rightarrow$ interactions.
     - Leverage VAE framework to **model the causal relations** behind the interaction generation process.
  2) Mitigate the effect of out-of-date interactions.
     - **Counterfactual inference:** what the user preference would be if the out-of-date interactions were removed?

Causal Modeling for OOD Recommendation

- OOD recommendation with **unobserved** user features.
  - Unobserved factors cause preference shifts.
  - Example: friend recommendations, hot event, and other environmental factors.

Papers on Counterfactual Recommendation

Outline

• Part 1 (90 min, 9:00—10:30)
  • Introduction (Wenjie Wang, 15 min)
  • Structural causal models for recommendation (Yang Zhang and Wenjie Wang, 60~70 min)
  • Q&A (5 min)
  • Coffee break (30 min)

• Part 2 (90 min, 11:00-12:30)
  • Potential outcome framework for recommendation (Haoxuan Li and Peng Wu, 60~70 min)
  • Comparison (Fuli Feng, 2 min)
  • Conclusion, open problems, and future directions (Fuli Feng, 20 min)
  • Q&A (5 min)
Outline

• Part 1 (90 min, 9:00—10:30)
  • Introduction (Wenjie Wang, 15 min)
  • Structural causal models for recommendation (Yang Zhang and Wenjie Wang, 60~70 min)
  • Q&A (5 min)
  • Coffee break (30 min)

• Part 2 (90 min, 11:00-12:30)
  • Potential outcome framework for recommendation (Haoxuan Li and Peng Wu, 60~70 min)
  • Comparison (Fuli Feng, 2 min)
  • Conclusion, open problems, and future directions (Fuli Feng, 20 min)
  • Q&A (5 min)
Outline

• Part 1 (90 min, 9:00—10:30)
  • Introduction (Wenjie Wang, 15 min)
  • Structural causal models for recommendation (Yang Zhang and Wenjie Wang, 60~70 min)
  • Q&A (5 min)
  • Coffee break (30 min)

• Part 2 (90 min, 11:00-12:30)
  • Potential outcome framework for recommendation (Haoxuan Li and Peng Wu, 60~70 min)
  • Comparison (Fuli Feng, 2 min)
  • Conclusion, open problems, and future directions (Fuli Feng, 20 min)
  • Q&A (5 min)
Comparison between PO and SCM for Recommendation

• Connections
  • logically equivalent: most theorem and assumptions can be equally translated.

• SCM
  • Intuitive: use causal graph to explicitly describe causal relationships.
  • Need more knowledge and assumptions on the causal graph.

• PO
  • Easy to capture some assumptions that can not be naturally represented by DAGs, such as the identification of the Local Average Treatment Effect (LATE).

An intuitive example:
  • To estimate the causal effect of T on Y, SCM might first assume the relationships between $X_1, X_2, X_3, T,$ and $Y$, and then SCM can control $X_1$.
  • PO might directly control $X_1, X_2, and X_3$ without knowing the fine-grained causal relationships.
Outline

• Part 1 (90 min, 9:00—10:30)
  • Introduction (Wenjie Wang, 15 min)
  • Structural causal models for recommendation (Yang Zhang and Wenjie Wang, 60~70 min)
  • Q&A (5 min)
  • Coffee break (30 min)

• Part 2 (90 min, 11:00-12:30)
  • Potential outcome framework for recommendation (Haoxuan Li and Peng Wu, 60~70 min)
  • Comparison (Fuli Feng, 2 min)
  • Conclusion, open problems, and future directions (Fuli Feng, 20 min)
  • Q&A (5 min)
Summary of Causal Recommendation

• Causal frameworks → Better recommender systems
  - Debiasing
  - Fairness
  - OOD Generalization
  - … (Many other researches, we apologize for not covering all! Kindly let us know about your work and suggestions: wenjiewang96@gmail.com)

• Try a causal perspective to solve your recommendation problem

• Two frameworks: PO and SCM-based methods
  - Causal graph is the key of the SCM-based methods.
  - SCM based methods may need more causal assumptions.
  - Propensity scores are usually used in PO-based methods.

• How to choose between PO and SCM? Practical requirements
Open Problems and Future Directions

Data

Collecting
(clicks, rates ...)

User

Feedback Loop

Serving

(Top-N recommendations)

Training

Causal assumption

Modeling

Evaluation

© Copyright National University of Singapore. All Rights Reserved.
Open Problems and Future Directions

• PO & SCM requires causal assumptions
  • Existing PO-based methods need to choose covariates to satisfy the exchangeability assumption.
  • Existing SCM-based methods need to manually draw the casual graph.

\[ P(Y^a \perp A | L) \]

How to obtain proper causal assumptions?

• Recommender system is a complex environment.
• Prior knowledge are insufficient.
Open Problems and Future Directions

- Future direction: causal discovery in recommendation

Automatic discovery of cause graphs with causal discovery algorithms
Open Problems and Future Directions

• Future direction: causal discovery in recommendation
• Challenges for applying casual discovery algorithms in recommendation

• Normal causal discovery algorithm only deals with few variables
• Challenge 1:

High-dimensional and hidden variables.
Open Problems and Future Directions

• Future direction: causal discovery in recommendation
• Challenges for applying casual discovery algorithms in recommendation

• The output usually is a set of causal graphs instead of only one graph.
• Challenge 2:
  Unreliable graphs in the output.
Open Problems and Future Directions

Bias is amplified in the feedback loop.

How to model the causal effect in the feedback loop?
Open Problems and Future Directions

Future direction: Temporal causal modeling
Open Problems and Future Directions

- One thousand papers, one thousand evaluation protocols

Normal setting is hard to show the superiority of the causal recommendation. Lack the standard evaluation setting.

- Future direction: benchmark

New benchmark dataset for causal recommendation, standardize the evaluation setting.
Open Problems and Future Directions

• Future direction: causality-aware evaluation metrics

Example 1 -- the effect of recommending operation

A and B are both matched to user preference, but recommending B can bring uplift gains.


<table>
<thead>
<tr>
<th>Item</th>
<th>recommend</th>
<th>Not-recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>purchase</td>
<td>purchase</td>
</tr>
<tr>
<td>B</td>
<td>purchase</td>
<td>Not-purchase</td>
</tr>
</tbody>
</table>

Example 2 --- path-specific fairness

Z affects C via two paths: $Z \rightarrow A \rightarrow C$ and $Z \rightarrow C$

Only $Z \rightarrow C$ is unfair.

Open Problems and Future Directions

• How can ChatGPT support recommender systems?
  • ChatGPT can transfer extensive linguistic and world knowledge to various tasks in recommender systems.
  • Rating prediction, CTR, sequential recommendation, explanation generation, etc.

• Using users’ historical interaction behaviors.
  • Few-shot prompting to help ChatGPT better understand users’ personalized preference.


What about causality for recommendation with LLM?
Open Problems and Future Directions

- **Future direction:** Fairness of LLM4Rec

RQ: If sensitive attribute is not given, will the recommendation result be biased towards a certain sensitive attribute?

-> biased to certain sensitive attribute will lead to unfair!

Open Problems and Future Directions

- Future direction: Fairness of LLM4Rec

If you don't disclose your sensitive attributes, ChatGPT will treat you as a young white American

Open Problems and Future Directions

Conversational rec. and generative rec.:
- guide/nudge users
- new preference
- less misinformation
- less polarity

Wang et al. arxiv, Generative Recommendation: Towards Next-generation Recommender Paradigm. 2023
Open Problems and Future Directions

• Future direction: Physical Communication
Thanks!

Call for papers

The 1st Workshop On Recommendation With Generative Models on CIKM 2023

https://rgm-cikm23.github.io/

Slides: https://causalrec.github.io/