基于时间感知逆倾向评分的序列推荐去偏

Sirui Huang
sirui.huang@connect.polyu.hk 香港理工大学 中国香港特别行政区 悉尼科技大学 澳大利亚悉尼

Jing Long
jing.long@uts.edu.au
悉尼科技大学
澳大利亚悉尼

Qian Li 科廷大学 珀斯，澳大利亚

Guandong Xu 香港教育大学 中国香港特别行政区 悉尼科技大学 悉尼，澳大利亚

Qing Li†
qing-prof.li@polyu.edu.hk 香港理工大学 中国香港特别行政区

arXiv:2603.04986v1 [cs.IR] 5 Mar 2026

摘要
序列推荐（SR）通过建模历史行为的时间顺序，预测用户的下一次交互。现有方法，包括传统序列模型和生成式推荐系统，虽然表现出色，但主要依赖点击或购买等显式交互，却忽略了物品曝光。这种忽略会引入选择偏差：已曝光但未点击的物品被误解为用户不感兴趣；同时也会引入曝光偏差：未曝光的物品被视为无关物品。要有效解决这些偏差，必须区分“未曝光”的物品和“用户不感兴趣”的物品，而这无法仅凭历史数据中的相关性可靠地推断。反事实推理通过估计假设曝光情境下的用户偏好，提供了一种自然的解决方案；逆倾向评分（IPS）是进行此类估计的常用工具。然而，传统IPS方法是静态的，无法捕捉用户行为中的序列依赖关系和时间动态。为克服这些局限，我们提出了时间感知逆倾向评分（TIPS）。与传统的静态IPS不同，TIPS能够有效考虑序列依赖关系和时间动态，从而更准确地捕捉用户偏好。大量实验表明，作为插件，TIPS能够持续提升各种序列推荐器的推荐性能。我们的代码将在论文录用后公开。

关键词
序列推荐，去偏，时间感知逆倾向评分

1 引言
序列推荐（SR）旨在通过建模历史交互的时间顺序来预测用户的下一次交互。序列推荐模型并非将每次交互视为独立事件，而是借助序列感知模型捕捉不断演化的偏好，使系统能够同时推断短期与长期兴趣，从而比仅基于静态用户画像的方法产生更准确的推荐。

这些序列推荐器大致可分为两类。首先，传统序列模型，例如循环神经网络（RNNs） [6] 和类 Transformer 架构 [16, 24, 25]，被用于根据历史交互序列中的显式反馈计算候选物品的概率，从而对推荐列表中的物品进行排序 [6, 10, 24, 39]。例如，Yuan等人使用基于 Transformer 的架构来捕捉多种类型的用户正向反馈，包括购买、点击和加入购物车，以丰富用户序列模式建模 [24]。其次，生成式推荐系统受自然语言处理进展的启发，旨在学习历史交互的分布并直接生成下一个交互物品 [11, 14, 18, 29, 31]。例如，Li 等人利用扩散模型中的分布来动态表示用户的多样化兴趣以及物品的不同特征 [14]。在这两类方法中，当前大多数序列推荐器仅依赖显式交互。尽管一些方法试图纳入上下文信息以校正偏差，但大多数现有方法主要依赖用户交互历史中的显式反馈信号（例如购买或点击）来提高预测准确率。然而，现有序列推荐模型因忽略物品曝光而存在放大偏差的风险。如图 1 所示，仅依赖交互（例如，点击）而忽略物品曝光的序列推荐模型会学到有偏差的用户偏好视图，因为它只建模用户对有限的预先交互物品集合的用户反馈（即，现有工作仅建模$\mathbf{C} \rightarrow \mathbf{U}$，而忽略$\mathbf{E} \rightarrow \mathbf{C} \rightarrow \mathbf{U}$和$\mathbf{E} \rightarrow \mathbf{U}$）。

物品曝光缺失主要会导致序列推荐中的两种偏差：**i)曝光偏差**，即将从未曝光的物品默认为用户不感兴趣；**ii) 选择偏差**，即将未发生交互的物品视为用户不感兴趣，尽管缺少交互可能是由展示策略或上下文所致。要有效处理这两种偏差，必须区分“未曝光”的物品和“用户不感兴趣”的物品，因为训练数据中缺失的交互可能由这两种情况中的任一种造成。然而，在缺少曝光日志的情况下，无法根据用户历史数据中的相关性可靠地推断这一区别。为此，反事实推理以物品已向用户曝光这一假设情形为条件[42]。具体而言，

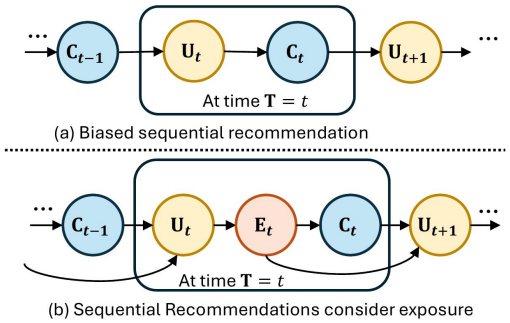


图1：（a）仅从交互中学习用户偏好的有偏序列推荐模型示例。（b）考虑曝光的系统示例。这里，$\mathbf{C}_t$ 表示时间 t的用户交互（例如，点击），$\mathbf{U}_t$ 表示用户在时间 t的偏好，而$\mathbf{E}_t$ 表示包含时间 t向用户曝光的所有物品的推荐列表。

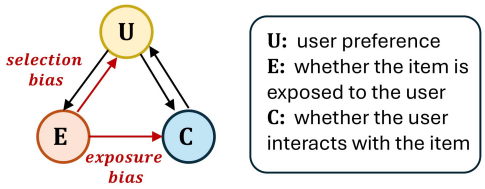


图2：SR 的结构因果模型（SCM）。

反事实推理通过提出"如果这个物品曾向用户曝光，会怎样？"，在处理**曝光偏差** 时区分未曝光与不感兴趣；同时，它通过提出"如果用户看到了这个物品，会表现出兴趣吗？"，来应对**选择偏差**，从而区分已曝光但不感兴趣的情况与潜在的兴趣。

现有的反事实推理通常依赖逆倾向评分（逆倾向评分），通过按曝光概率的倒数对已观测交互重新加权[17, 36]。然而，逆倾向评分本质上是静态的：它仅利用上下文信号（例如，用户属性或物品特征）独立地为每次交互分配权重，而不对时间依赖或交互间的动态关系建模。这种静态特性限制了逆倾向评分全面纠正序列推荐偏差的能力，因为这些偏差会随时间累积。i）第一个挑战是，静态逆倾向评分忽略了交互序列中物品之间的序列依赖关系。在序列推荐中，用户的下一次交互不仅取决于物品特征，也取决于过往交互，因为用户兴趣会随时间变化，先前的推荐也可能影响后续行为。简单地对交互样本重新加权可能会破坏用户交互序列中的因果链（例如，与 iPhone 17 交互的用户随后更可能与手机壳交互）。ii）其次，虽然逆倾向评分的重新加权模拟了所有物品的公平曝光，但它无法反映用户的真实兴趣，而用户兴趣往往会遵循时间趋势（例如，与 iPhone 17 交互的用户可能会对新发布的 iWatch 更感兴趣）。由于物品的实际曝光概率会因新品发布、促销活动或流行趋势而随时间变化，当前的逆倾向评分方法可能会过度增加新推出物品的权重，同时降低当前流行物品的权重。为了准确地对用户交互重新加权，逆倾向评分应考虑这些序列和时间动态，使估计的曝光与真实用户兴趣相一致。

为克服传统逆倾向评分在序列推荐中处理选择偏差和曝光偏差方面的局限，我们提出了一种新颖的框架——时间感知逆倾向评分（*HyperG*），以与模型无关的方式融入时间感知信息。该框架通过更准确地建模用户的隐式反馈，并捕捉交互中序列性且随时间变化的因果依赖，提升了传统序列模型和生成模型的效果。具体而言，我们不直接计算每个用户-物品交互的逆倾向评分，而是首先为训练数据中观察到的每个事实交互生成两个高质量的反事实交互。其中一个反事实交互改变该交互的时间表示，另一个则扰动物品本身的表示。结合历史交互，这两个反事实交互有助于准确建模符合用户行为序列性和时变特征的时间感知逆倾向性。*HyperG* 的主要贡献总结如下：

- 在缺少曝光日志的情况下，我们通过构建具有时间感知的反事实示例来估计物品曝光分布。
- 在缺少曝光日志的情况下，我们利用具有时间感知的倾向性对推荐进行重新加权，从而减轻曝光相关偏差（包括曝光偏差和选择偏差），并更准确地建模用户偏好。
- 我们在三个骨干模型上进行了大量实验，并将 *HyperG* 与两类推荐模型进行比较，以验证其有效性。

2 预备知识

本节首先提出结构因果模型（SCM）（图2），以阐明 SR 中的因果关系。随后，我们介绍传统的逆倾向评分（IPS）方法，并给出本文所关注问题的定义。

假设用户 $u \in \mathbf{U}$在时间$t \in \mathbf{T}$与物品 $v \in \mathbf{V}$发生交互，其中U、V、T分别表示用户集合、物品集合以及所有交互时间段。在因果模型（图2）中，变量$\mathbf{U}$ 表示用户偏好，二元变量$\mathbf{E} \in \{0,1\}$和$\mathbf{C} \in \{0,1\}$分别表示某物品是否被曝光给用户或被用户点击。本文中，为清晰起见，所有公式均针对单个用户 u进行表述。此处，$\mathbf{C}_v$和$\mathbf{E}_v$是二元变量，分别表示物品 v是否被点击以及是否被曝光。例如，$\mathbf{C}_v = 1$ 和$\mathbf{E}_v = 1$ 分别表示物品 v 被曝光以及被用户 u发生交互。

2.1 序列推荐的结构因果模型

图2展示了序列推荐中的因果关系。与图1不同，该结构因果模型（SCM）展示了固定时间下与曝光相关的因果关系，从而能够更清晰地加以解释。图2中的有向箭头表示信息从原因变量流向结果变量。图2中的每条边说明如下。

- $\mathbf{U} \rightarrow \mathbf{E}$ 表示用户的偏好（U）会影响推荐列表（E中向他们展示哪些物品。
- $\mathbf{E} \rightarrow \mathbf{U}$ 表示，除了交互数据外，推荐列表中物品的曝光情况（E）也反映了用户的真实偏好（U）（选择偏差）。

• $\mathbf{U} \rightarrow \mathbf{C}$ 表示用户的偏好（U）决定了用户会与哪些物品发生交互（例如 点击）（C）。
• $\mathbf{C} \rightarrow \mathbf{U}$ 表示用户交互（C）提供了有关用户偏好（U）的信息。
• $\mathbf{E} \rightarrow \mathbf{C}$ 表示交互（C）仅发生在已曝光物品（E）中（曝光偏差）。

曝光偏差 产生的原因是，推荐系统只能观察到向用户曝光的部分物品（$\mathbf{E} = 1$），而用户对未曝光物品（$\mathbf{E} = 0$）的偏好仍不可观测。如图2所示，曝光E是交互C的前提。例如，当用户只接触到十部电影时，模型无法推断用户是否会与推荐列表中从未出现过的其他未曝光电影发生交互。因此，模型只能估计以曝光为条件的交互概率，即*i.e.,*$P(\mathbf{C}|\mathbf{E} = 1)$，而不能对已曝光和未曝光物品的完整分布进行建模（即*i.e.,* $P(\mathbf{C}|\mathbf{E} \in \{0, 1\})$）。形式化地，曝光偏差可表述如下：

$$P(\mathbf{C}|\mathbf{E} = 1) \neq P(\mathbf{C}|\mathbf{E} \in \{1, 0\}), P(\mathbf{E} = 1) < 1 \tag{1}$$

其中，不等式左侧表示仅根据曝光物品学习到的有偏交互分布，而右侧则对应于以物品曝光概率为条件的理想交互分布。直观而言，这两个分布并不相等，因为物品的曝光概率通常小于1，且交互 C 本质上取决于物品曝光 E。

如图2所示，交互变量 C 由曝光 E 和用户偏好 U共同决定。然而，在缺少曝光日志的情况下，大多数推荐模型仅依赖具有 $\mathbf{C} = 1$ 的物品，即 *i.e.,* 交互物品。这会引入 **选择偏差**，因为模型会偏离与所选物品相关的已观测交互。例如，用户可能会看到十部电影，但只点击其中一部。如果模型仅从这部被点击的电影中学习，就会错误地认为用户对其余九部电影没有偏好。

$$P(\mathbf{U}|\mathbf{C} = 1, \mathbf{E} = 1) \neq P(\mathbf{U}|\mathbf{E} = 1) \tag{2}$$

其中，左侧表示在已观测交互条件下用户偏好的条件分布，而右侧表示所有曝光物品上的理想偏好分布。由于存在已曝光但未发生交互的物品，这两个分布存在差异。

在缺乏曝光日志的情况下，现有序列推荐模型主要利用显式用户反馈，考虑路径 $\mathbf{U} \rightarrow \mathbf{C}$、$\mathbf{C} \rightarrow \mathbf{U}$ 和 $\mathbf{U} \rightarrow \mathbf{E}$，而忽略了应对曝光偏差（路径 $\mathbf{E} \rightarrow \mathbf{C}$）和选择偏差（路径 $\mathbf{E} \rightarrow \mathbf{U}$）的必要性。

2.2 逆倾向评分

逆倾向评分是一种被广泛采用的统计方法，通过对用户-物品交互进行重加权来应对曝光偏差和选择偏差 [15, 17, 28, 35, 36, 40]。

2.2.1 倾向得分。基于逆倾向评分的推荐器使用倾向得分对交互进行重新加权，以纠正未观测曝光导致的偏差。具体而言，倾向得分对以下内容进行建模：

物品向用户曝光的概率，并用于对已观测交互重新加权，使其趋近于均匀曝光下的理想分布。这种重新加权可避免低估用户对曝光较少物品的偏好。形式上，物品 v 与用户 u 之间交互的倾向得分可定义为：

$$\pi(u, v) = P(\mathbf{E}_v = 1|u) \tag{3}$$

其中 $\pi(\cdot)$ 表示曝光分布估计函数。因此，系统会向曝光概率更高的物品分配更低的倾向得分，即 $P(\mathbf{E} = 1)$ 更高的物品。

2.2.2 使用逆倾向评分应对这两种偏差。 推荐中的**曝光偏差** 之所以产生，是因为推荐系统将未曝光物品视为用户不感兴趣。假设我们有一个未曝光物品 v。为使用逆倾向评分校正曝光偏差，考虑第1节中的“假如”问题：“如果该物品曾被曝光给该用户，会怎样？”我们的目标是在理想分布上估计交互概率，在该分布中所有物品具有同等的曝光机会。大多数推荐系统忽略了曝光是与物品发生交互的前提这一重要事实（即 路径 $\mathbf{E} \rightarrow \mathbf{C}$）。为应对这一忽略，基于逆倾向评分的方法针对每个已观测的用户-物品交互，按其物品曝光概率的倒数对推荐得分 $g(u, v)$ 进行重加权。形式化地，

$$\sum_{(u,v)\in\mathcal{D}_{\text{train}}} \frac{g(u, v)}{\pi(u, v)} P(\mathbf{U} = u) \simeq \mathbb{E}_{\mathbf{U},\mathbf{C}}[g(u, v)] \tag{4}$$

$$\mathbb{E}_{\mathbf{U},\mathbf{C}}[g(u, v)] = \sum_{u\in\mathcal{U}} P(\mathbf{U} = u) \sum_{v\in\mathcal{V}} P(\mathbf{C}_v = 1 \mid u) g(u, v) \tag{5}$$

公式（4）左侧表示对每个已观测交互的推荐分数采用逆倾向评分重新加权后的期望，其中 $\pi(u, v)$ 表示用户 u 与物品 v 之间特定交互的倾向得分。公式（4）右侧表示推荐分数的理想期望，该期望表示在公式（5）中定义的真实用户偏好分布 $P(\mathbf{U} = u)$ 下，用户与物品之间的交互概率。

选择偏差是指推荐模型仅从已交互物品子集中的已观测交互进行学习。我们有一个未观测的（即，未交互的）物品 v。为使用逆倾向评分解决选择偏差，考虑前述的“假设情境”问题：“如果用户 u 在物品曝光的情况下表现出兴趣 v呢？”我们的目标是在所有物品都有相同曝光机会的理想分布上推断用户的真实偏好 U。由于缺少曝光日志，模型在用户建模时未考虑曝光概率（即，图2中的路径 $\mathbf{E} \rightarrow \mathbf{U}$）。传统逆倾向评分通过将倾向得分 $\pi(\cdot)$ 的倒数应用于推荐模型 g 计算出的分数来纠正这一问题。

$$\sum_{u\in\mathcal{U},\mathbf{C}_v=1,\mathbf{E}_v=1} \frac{g(u, v)}{\pi(u, v)} \simeq \sum_{u\in\mathcal{U}} P(\mathbf{U} = u)[g(u, v)] \tag{6}$$

式（6）的左侧表示经逆倾向评分重新加权后的推荐分数的期望。右侧表示理想推荐分数的期望，其中 $P(\mathbf{U} = u)$ 是真实用户偏好分布中用户 u 的偏好。

2.3 问题定义

在序列推荐中，由式（3）所表述的倾向分数 $\pi(\cdot)$ 所描述的物品曝光概率会随时间变化，因为用户兴趣和曝光策略（例如，新品发布、促销活动）会发生变化。因此，本文使用逆倾向评分来处理曝光偏差和选择偏差，同时考虑序列推荐随时间变化的特性。具体而言，我们的目标是利用插件模型 f_φ，从观测到的事实交互中学习物品曝光分布 $P(\mathbf{E}=1|u)$，以适用于任意推荐模型 g_θ。我们随时间估计物品曝光，并用时间感知曝光概率的倒数对交互进行动态重加权，从而近似一种反事实情境，即所有物品具有相同的被曝光机会。这使得在考虑物品曝光的序列性与时间动态的同时，能够推断交互概率。对于每个 $u \in$ U、$v \in$ V 以及交互时间 $t \in$ T，都存在一个无偏推荐分数 $\hat{y}(\cdot)$：

$$\hat{y}(u,v,t) \approx (f_\varphi \circ g_\theta)(u,v,t) \tag{7}$$

其中，$\circ$ 表示用于曝光估计的插件模型f_φ与推荐模型 g_θ的函数复合，二者分别由 φ和 θ参数化。

3 方法论

本文重点研究如何使用逆倾向评分来处理序列推荐中的序列因果关系和时间因果关系，并解决曝光偏差与选择偏差这两种由缺少曝光数据而产生的主要偏差。我们提出了时间感知逆倾向评分（TIPS），将时间信息纳入逆倾向评分，以准确捕捉序列推荐的动态变化。具体而言，我们提出了一种新颖的框架（图 3），将插件曝光估计模型 f_φ 与 TIPS 相结合，以优化推荐系统的训练目标。模型 f_φ 通过对反事实物品—时间对进行推理来估计物品曝光分布，进而利用这些分布计算时间感知倾向得分，以进行偏差校正。

3.1 双编码策略

在进行反事实推理之前，我们对物品和时间戳进行编码，以将静态物品语义与用户偏好及物品曝光的动态变化解耦，从而使逆倾向评分能够向无偏的曝光分布进行调整。这种双编码包括维护两个独立的物品嵌入查找表，并提取归一化时间特征。

3.1.1 物品编码。 基于图2所示的结构因果模型，有必要分别对物品的交互分布和曝光分布进行建模。为此，我们维护两个隐藏嵌入矩阵$\mathbf{H}^{(C)}$ 和$\mathbf{H}^{(E)}$。每个物品 $v_i \in$ V都关联两个可训练嵌入，定义如下。

$$\mathbf{c}_i = \text{Lookup}(\mathbf{H}^{(C)}, i),\ \mathbf{e}_i = \text{Lookup}(\mathbf{H}^{(E)}, i) \tag{8}$$

其中，嵌入$\mathbf{H}^{(C)} \in \mathbb{R}^{|\mathcal{V}|\times d}$ 捕捉从点击等历史用户—物品显式反馈中学习到的交互信息，并编码反映用户偏好的物品语义和协同信息。曝光嵌入$\mathbf{H}^{(E)} \in \mathbb{R}^{|\mathcal{V}|\times d}$ 表示物品曝光的分布，其中涉及流行度或促销活动等因素，这些因素会影响物品是否展示给用户。

维护两个独立查找嵌入的必要性源于两点关键考量：i）交互与曝光反映不同的用户反馈。 交互物品代表用户的正向偏好，而曝光物品，尤其是那些未被交互的物品，可能表明缺乏兴趣。这两类信号都有价值，但本质上截然不同。ii）共享嵌入会放大偏差，并在后续逆倾向评分中进一步放大；逆倾向评分要求所估计的倾向性（即曝光概率，即分母）独立于推荐预测（分子）。共享嵌入会造成依赖，从而在后续调整过程中增加方差和偏差。

3.1.2 时间嵌入。 数据集中的交互时间 $t \in$ T 按时间顺序排列，其中每个 t_i 值都表示用户 u 与物品 v_i交互的时间。两次交互之间的间隔很重要，因为用户偏好会发生变化。一个常见的现象是，近期的交互更能代表用户当前的偏好。例如，如果用户昨天观看了一部喜剧片，而六个月前看了一部纪录片，那么推荐系统在推荐下一部影片时应优先考虑近期观看的喜剧片。我们计算位置 i处的时间戳 t_i与位置 $i-1$处的时间戳 t_{i-1}之间的差值，然后进行归一化。随后，通过多层感知机 MLP(·) 将归一化后的时间间隔映射到嵌入空间。形式上，$\mathbf{t}_i = \text{MLP}(\text{Norm}(t_i - t_{i-1}))$，其中 Norm(·) 表示归一化过程，$\mathbf{t}_i \in \mathbb{R}^{|\mathcal{T}|\times d}$ 是时间的嵌入表。

3.2 反事实样本构建

为了在没有曝光数据的情况下估计物品曝光，我们基于三个反映推荐场景常见模式的假设构建反事实样本。对于每个交互 $(u, v_i, t_i) \in \mathrm{D}_{\text{train}}$，物品 v_i在时间 t_i被正向曝光并点击，即，对于用户 u，存在$\mathbf{E}=1, \mathbf{C}=1$。对于交互数据$\mathrm{D}_{\text{train}}$中的每个事实物品-时间对 (v_i, t_i)，我们假设另外三个正向曝光的物品-时间对，其中$\mathbf{E}=1$, $\mathbf{C}=0$：

i) 相似物品 (v_i^{sim}) 更有可能在同一时间被曝光，因为推荐算法往往会同时推广相关的替代项 [2, 3, 43]。基于这一假设，我们提出“如果向用户展示一个相似物品会怎样？”为回答这一反事实问题，我们基于曝光嵌入表$\mathbf{H}^{(E)}$检索相似物品，该表旨在学习物品的曝光分布。反事实相似物品 v_i^{sim}的曝光嵌入可形式化表示为：

$$\mathbf{e}_i^{sim} = \arg\max_{v_j \in \mathcal{V}\setminus v_i} \text{sim}(\mathbf{e}_j, \mathbf{e}_i),\ \text{sim}(v_i, v_j) = \frac{\mathbf{e}_i^\top \mathbf{e}_j}{\|\mathbf{e}_i\|\,\|\mathbf{e}_j\|} \tag{9}$$

其中，$\mathbf{e}_i$、$\mathbf{e}_j \in \mathbf{H}^{(E)}$，余弦相似度函数 sim(·) 可替换为点积或其他相似度计算。

ii) 热门项目 (v_i^{pop}) 出现得更频繁，且通常与其他内容一同展示，因此增加了同时曝光的可能性 [1, 41]。因此，我们提出“如果向用户展示一个非常热门的物品，会怎样？”为了有效地对热门趋势下的用户偏好建模，我们还构造了正向曝光的物品-时间

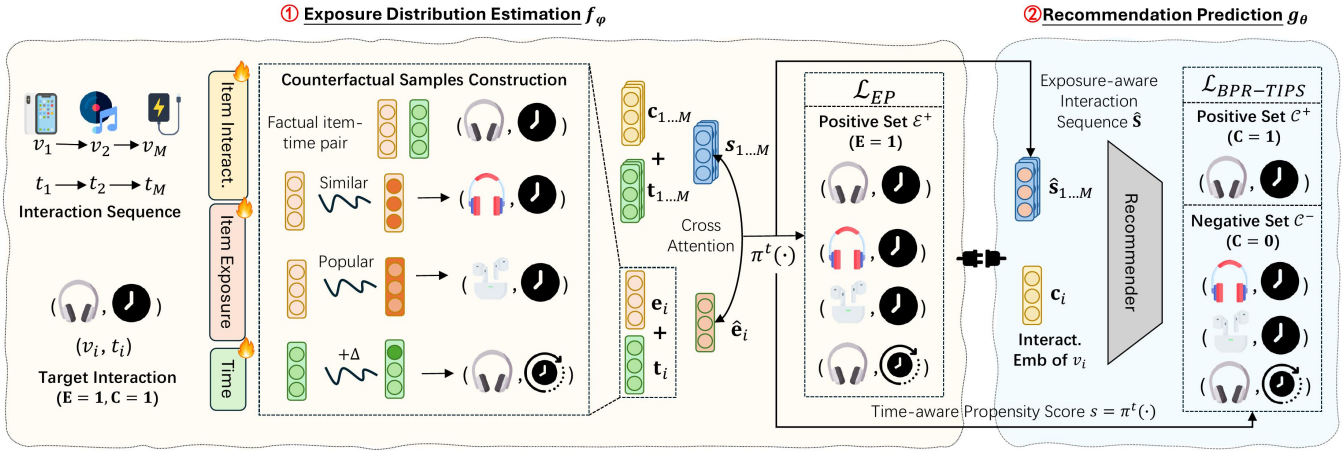


图 3：时间感知逆倾向评分（TIPS）框架概览，该框架旨在作为适用于任何序列推荐模型的插件。

对 $(v_i^{pop},\ t_i)$，其中包含一个反事实热门物品。具体而言，我们基于所有用户 $u \in \mathrm{U}$ 的历史交互数据，计算除 v_i 之外物品的流行度分数。由于我们专注于对物品的曝光特征建模，同时也为了与v_i^{sim}保持一致，我们还从曝光嵌入矩阵 $\mathbf{H}^{(E)}$ 中提取了 v_i^{pop} 的嵌入。

$$\mathbf{e}_i^{pop} = \mathrm{Lookup}(\mathbf{H}^{(E)}, \mathrm{index}(v^{pop})) \tag{10}$$

其中，$v^{pop} \in \mathrm{V}$ 是选取的热门物品，$\mathrm{index}(\cdot)$ 表示返回 v^{pop} 的 ID 的索引函数。具体而言，我们首先统计 $\mathrm{D}_{\mathrm{train}}$ 中每个物品 $v \in \mathrm{V}$ 的出现次数，然后从最热门的 K 个物品中，为每个 v_i 随机抽取一个物品 v^{pop}。

$$\begin{aligned} v^{pop} &\sim \mathrm{Unif}\Big(\mathrm{TopK}\big(\{\mathrm{count}(v) \mid v \in \mathcal{V}\}, K\big)\Big) \\ \mathrm{count}(v) &= \sum_{(v_j,t_j)\in\mathcal{D}_{\mathrm{train}}} \mathbf{1}\big(v_j = v \ \wedge\ t_j \in [t-\tau, t]\big) \end{aligned} \tag{11}$$

其中，$\mathrm{Unif}(\cdot)$ 表示从其余物品集合 $\mathrm{V}\backslash v_i$ 中均匀抽取物品，K 是控制所考虑的热门物品数量的超参数，$\mathrm{D}_{\mathrm{train}}$ 表示交互数据的训练集，τ 控制流行度计数函数 $\mathrm{count}(\cdot)$ 所采用的时间窗口。

iii) 同一物品不同时间 (t_i^*) 由于短期重排序等机制 [22]，同一物品可能会在短时间窗口内持续曝光给用户。基于这一发现，我们提出："如果向用户展示的是同一物品，但时间稍有不同，会怎样？"。为了推理这一问题中的反事实曝光时刻，我们对时间嵌入 $\mathbf{t}_i$ 进行轻微干预Δ，通过 $\mathbf{t}_i^* = \mathbf{t}_i + \Delta$构造物品-时间对（$v_i$，$t_i^*$）。$\Delta \in \mathbb{R}^d$是一个幅度较小的随机向量。此处，我们设定 $\Delta \in [-1\times10^{-4}, 1\times10^{-4}]$。

至此，我们已推导出三个反事实物品–时间对的嵌入（$\mathbf{e}_i^{sim}$，$\mathbf{t}_i$）、（$\mathbf{e}_i^{pop}$，$\mathbf{t}_i$）和（$\mathbf{e}_i$，$\mathbf{t}_i^*$）。在本文中，这三个反事实物品–时间对在 f_φ 中被视为用于曝光分布估计的正样本，但在 g_θ 中则被视为负样本，用于增强基于交互的用户偏好推断。以下各节将详细说明如何在 *HyperG* 中进一步利用这三个反事实对以及事实交互（v_i，t_i）。

3.3 曝光影响交互（E → C）

曝光偏差 的产生是因为用户只能与自己接触过的物品进行交互，导致观测到的交互反映的是曝光机制，而非纯粹的用户偏好。回顾图2中的结构因果模型可知，物品曝光是用户-物品交互的必要前提。基于这一因果关系，我们将曝光估计与包含时间信息的用户历史交互相结合，并进一步使用时间感知逆倾向评分框架调整预测，以缓解曝光偏差。

给定从交互嵌入矩阵 $\mathbf{H}^{(C)}$ 中获取的交互物品嵌入，我们按照其交互时间戳 t_i 对它们进行排序，得到序列 $\{\mathbf{c}_0, \mathbf{c}_1, ..., \mathbf{c}_M\}$。然后，我们将该物品序列与对应的时间嵌入 $\{\mathbf{t}_0, \mathbf{t}_1, ...\mathbf{t}_M\}$ 结合。

$$\mathbf{S} = \{\mathbf{c}_0, \mathbf{c}_1, ..., \mathbf{c}_M\} \oplus \{\mathbf{t}_0, \mathbf{t}_1, ...\mathbf{t}_M\} \tag{12}$$

其中，$\oplus$ 表示逐元素相加，$\mathbf{S} = \{\mathbf{s}_m\} \in \mathbb{R}^{M\times d}$ 表示仅根据历史交互得到、忽略曝光信息的用户偏好的基础嵌入，M 是用户历史交互序列的最大长度，m 表示序列中元素的位置索引。

在整合之前，为了动态估计候选物品的曝光分布，我们对每个事实或反事实物品-时间对的物品和嵌入进行聚合，即 (v_i, t_i)、(v_i^{sim}, t_i)、(v_i^{pop}, t_i) 和 (v_i, t_i^*)。例如，对于物品-时间对 (v_i, t_i)，其隐层嵌入 $\hat{}\mathbf{e}_i \in \mathbb{R}^d$ 表示如下：

$$\hat{\mathbf{e}}_i = \mathbf{e}_i + \mathbf{t}_i \tag{13}$$

其中$\mathbf{e}_i$和$\mathbf{t}_i$分别由第3.1.1节和第3.1.2节得到。符号"+"表示加法运算。随后，我们通过交叉注意力机制将曝光信息融入交互序列。具体而言，曝光嵌入$\hat{}\mathbf{e}_i$作为查询$\mathbf{q}$，而序列$\mathbf{S}$中每个交互的嵌入则同时作为键$\mathbf{k}$和值$\mathbf{v}$。形式化地，$\mathbf{q}^{(n)} = \hat{}\mathbf{e}_i\mathbf{W}_Q^{(n)} + \mathbf{b}_Q^{(n)}$，$\mathbf{k}^{(n)} = \mathbf{s}_m\mathbf{W}_K^{(n)} + \mathbf{b}_K^{(n)}$，以及$\mathbf{v}^{(n)} = \mathbf{s}_m\mathbf{W}_V^{(n)} + \mathbf{b}_V^{(n)}$，其中$\mathbf{W} \in \mathbb{R}^{\frac{d^2}{N}}$和$\mathbf{b} \in \mathbb{R}^{\frac{d}{N}}$分别表示

加权矩阵和偏差。(n)表示交叉注意力的第 n个注意力头， N 表示注意力头总数，而$\mathbf{s}_m$ 表示同一用户的历史交互子序列。第 n个注意力头的交叉注意力计算公式为：

$$\text{Attention}(\mathbf{q}^{(n)}, \mathbf{k}^{(n)}, \mathbf{v}^{(n)}) = \text{softmax}\left(\frac{\mathbf{q}^{(n)}(\mathbf{k}^{(n)})^\top}{\sqrt{d}}\right)\mathbf{v}^{(n)} \quad (14)$$

交叉注意力机制的输出嵌入通过平均池化操作进行聚合，以生成一个标量分数。该分数作为曝光倾向，表示物品 v_i 在时间 t_i 被曝光的概率。

$$s_{v_i,t_i} = \pi^t(\mathbf{S}, \hat{\mathbf{e}}_i) = \text{mean}(\hat{\mathbf{S}}) \quad (15)$$

$$\hat{\mathbf{S}} = \sum_n \text{Concat}_m(\text{Attention}(\mathbf{q}^{(n)}, \mathbf{k}^{(n)}, \mathbf{v}^{(n)})) \quad (16)$$

其中， $\pi^t(\cdot)$ 表示用于估计时间感知曝光分布的函数，其输入为交互序列 $\hat{}\mathbf{S}$ 和特定的曝光样本 $\hat{}\mathbf{e}_i$； Concat$(\cdot)$ 的下标 m 表示将历史交互所有子序列的结果拼接起来。 $\pi^t(\cdot)$ 的时间感知倾向得分 s 用于解决曝光偏差（见第 3.5 节）。

为训练插件式曝光估计模型 f_φ，使其能够准确捕捉曝光分布，并将曝光信息与用户–物品交互无缝整合，我们构建了正样本集 $\mathbf{E}^+$ 和负样本集 $\mathbf{E}^-$。正样本集 $\mathbf{E}^+$ 包含被视为正向曝光的物品及其对应的曝光时间，包括训练交互中观测到的事实物品-时间对（v_i， t_i），以及为每个物品-时间对人工构造的三个反事实曝光样本：（v_i^{sim}， t_i）、（v_i^{pop}， t_i）和（v_i， t_i^*）。负样本集 $\mathbf{E}^-$ 由从剩余数据 $\mathbf{D}\backslash\mathbf{E}^+$中随机采样的物品–时间对（$v'$， t'）组成。

$$\mathcal{L}_{\text{EP}} = \frac{-1}{|\mathcal{E}|}\left[\sum_{(v,t)\in\mathcal{E}^+}\log\sigma(s_{v,t}) + \sum_{(v',t')\in\mathcal{E}^-}\log\left(1-\sigma(s_{v',t'})\right)\right], \quad (17)$$

其中， $|\mathbf{E}| = |\mathbf{E}^+ \cup \mathbf{E}^-|$ 表示负样本集$\mathbf{E}^- = \{v', t'\}$和正样本集$\mathbf{E}^+ = \{(v_i, t_i), (v_i^{sim}, t_i), (v_i^{pop}, t_i), (v_i, t_i^*)\}$中的样本数量。$s_{v,t}$和$s_{v',t'}$分别表示正样本$(v, t)$和负样本 v', t' 被曝光的概率，使用公式(15)计算得到。 $\sigma(\cdot)$表示Sigmoid函数。

3.4 曝光对用户偏好的影响（$\mathbf{E} \rightarrow \mathbf{U}$）

由于推荐系统仅使用$\mathbf{D}_{\text{train}}$中的已观测交互进行训练，因此会产生**选择偏差**。$\mathbf{D}_{\text{train}}$仅包含用户交互过的物品，因此可能导致用户偏好的建模出现偏差。这种偏差往往会导致对热门或频繁曝光物品的偏好估计过高，同时低估未观测或曝光较少物品的偏好。本文利用反事实物品曝光数据，为学习用户偏好提供额外信息，从而减轻选择偏差的影响并提升模型的泛化能力，如图2中的路径$\mathbf{E} \rightarrow \mathbf{U}$ 所示。

我们将*HyperG*视为一种插件模型，旨在为任意序列推荐模型（记作 g_θ）增强曝光信息。为展示*HyperG*的灵活性，我们说明如何在两类推荐系统中对曝光效应进行建模：基于注意力的序列模型（以简单的注意力层 [25], 为例）和生成模型（以扩散 [14]为例）。

3.4.1 传统序列模型。按照 SR 中的常见做法，我们对曝光感知用户偏好 $\hat{}\mathbf{S}_u$ （如式 (16) 所示）应用自注意力机制，以捕捉交互序列中物品之间的信息。为了根据用户的历史交互序列对用户进行建模，式 (18) 中使用的自注意力机制可以替换为任意序列模型，例如循环神经网络或门控循环单元模块。

$$\mathbf{q}_u^{(n)}, \mathbf{k}_u^{(n)}, \mathbf{v}_u^{(n)} = \hat{\mathbf{s}}_m\mathbf{W}^{(n)} + \mathbf{b}^{(n)}, \ \mathbf{u} = \sum_n \text{Attention}(\mathbf{q}_u^{(n)}, \mathbf{k}_u^{(n)}, \mathbf{v}_u^{(n)}) \quad (18)$$

其中，$\mathbf{q}_u^{(n)}$、$\mathbf{k}_u^{(n)}$、$\mathbf{v}_u^{(n)}$分别表示用于计算用户嵌入 $\mathbf{u}$ 的查询、键和值，$\mathbf{W} \in \mathbb{R}^{\frac{d^2}{N}}$ 和 $\mathbf{b} \in \mathbb{R}^{\frac{d}{N}}$。

给定用户偏好嵌入$\mathbf{u}$ 以及候选物品 v_i 的交互嵌入 $\mathbf{c}_i$，可以通过计算二者的内积来预测推荐分数 y_i，即， $y_i = \text{InnerProduct}(\mathbf{u}, \mathbf{c}_i)$。

3.4.2 生成模型。对于以扩散表示的生成模型，我们以曝光感知用户交互序列 $\hat{}\mathbf{S}_u$ 的嵌入为条件，生成理想物品嵌入 $\hat{}\mathbf{c}_i$，该嵌入代表用户偏好。

$$\hat{\mathbf{c}}_i = Z_\Theta([\mathbf{z}_0, \mathbf{z}_1, ..., \mathbf{z}_M]) \quad (19)$$

$$\mathbf{z}_i = \hat{\mathbf{s}}_m + \lambda_i \odot (\dot{\mathbf{c}}_i + \mathbf{d}) \quad (20)$$

其中 Z_Θ 表示从历史序列中捕捉用户偏好的模型， $\odot$ 表示逐元素乘积， $\mathbf{d}$ 表示在 [25]中位置编码之后的扩散步嵌入， λ_i 从高斯分布中采样，$\hat{}\mathbf{s}_m$ 表示序列$\hat{}\mathbf{S}$中位置 i 处已交互物品的感知曝光嵌入，而 $\dot{}\mathbf{c}_i$ 则分别表示正向扩散或反向过程中候选物品v_i对应的加噪或重构后的交互嵌入。

基于扩散的模型根据理想物品$\hat{}\mathbf{v}$ 与实际物品集合$\mathbf{V}$中物品的相似度计算推荐分数。对于实际物品 v_i， 有 $y_i = \text{Sim}(\hat{}\mathbf{c}_i, \mathbf{c}_i)$。

3.5 推荐

基于对交互和用户偏好中曝光影响的建模，我们提出了一种新颖的时间感知逆倾向评分（TIPS）方法，将时间信息纳入最终的推荐目标。

3.5.1 时间感知逆倾向评分。对于训练交互数据中的事实物品-时间对（v_i, t_i），存在时间感知逆倾向评分s_{v_i,t_i}^{-1}。因此，序列推荐模型的训练目标可表述如下。

$$\mathcal{L}_{\text{BPR-TIPS}} = \frac{1}{N}\sum_{(v_i,t_i)\in C^+,(v_j,t_j)\in C^-}\frac{\mathbf{w}_{v_i,t_i} ln\sigma(y_i, y_j)}{\sum_{(v_i,t_i)\in C^+} s_{v_i,t_i}} \quad (21)$$

对于基于序列模型或基于扩散的方法， y 分别通过内积或相似度度量计算。$s_{v_i,\ t_i}$是（v_i， t_i）的曝光倾向 $\in \mathbf{C}^+$，如公式 (15) 所定义。请注意，事实物品-时间对（v_i， t_i） $\in \mathbf{C}^+$ 被

视为正交互样本，而三个反事实对应项（v_i^{sim}, t_i），（v_i^{pop}, t_i），（v_i, t_i^*）则被视为集合 C^- 中相应的负样本。

在式（21）中，$\mathbf{w}_{v_i,\ t_i}$ 将时间衰减纳入曝光倾向得分 $s_{v_i,\ t_i}$ 用于时间 t_i 的物品 v_i，而非直接进行调整。具体而言，$\mathbf{w}_{v_i,\ t_i}$ 会提高近期交互的权重，并进一步加强对曝光频率较低的物品的校正。权重的计算公式如下。

$$\mathbf{w}_{v_i,t_i} = \frac{\exp(-\mu(t_i - t_{i-1}))}{\max(s_{v_i,t_i}, \epsilon)} \tag{22}$$

其中，$\mu > 0$ 是一个衰减超参数，用于控制指数项随时间递减的速度；μ 越小，衰减越慢，因此过去的交互会在更长时间内保持影响力。ϵ 是一个较小的基线值，可确保稳定性并避免曝光概率过小而导致除数过小。

3.5.2 模型优化。所提出的框架 $f_\varphi \circ g_\theta$ 通过结合由式（17）中的 L_{EP} 引导的曝光分布学习，以及采用时间感知逆倾向评分调整的推荐（式（21）中的 $L_{BPR\text{-}TIPS}$）进行优化。具体而言，最终训练目标定义如下。

$$\mathcal{L} = \mathcal{L}_{BPR-TIPS} + \gamma \mathcal{L}_{EP} \tag{23}$$

其中 γ 是一个超参数，用于控制*HyperG* 框架对物品曝光的关注程度。

4 实验

为了验证*HyperG* 在序列推荐中的有效性，我们进行了大量实验，以回答以下研究问题。

- **研究问题1**：与最先进方法相比，*HyperG* 在不同骨干推荐模型上的表现如何?
- **研究问题2**：*HyperG* 的各个组件如何提升序列推荐性能?
- **研究问题3**：超参数的不同值如何影响推荐性能?
- **RQ4**：所提出的 *HyperG* 在哪些方面优于传统逆倾向评分?

5 实验设置

5.0.1 数据集。 我们在四个规模各异、广泛使用的推荐数据集上评估了*HyperG* （见表 1）。

表 1：数据集统计信息。

数据集	#用户	#物品	#交互	时间范围（月/日/年）
ML-1M	5950	3532	574619	2000年4月25日-2003年2月28日（34.2 个月）
ML-10M	66028	10254	4980475	1995年1月9日-2009年1月5日（168个月）
Music4All	14124	99596	5109577	2013年12月30日-2019年3月26日（62.8个月）
GoodReads	17660	29972	409600	2001年2月13日-2017年11月2日（200.3个月）

MovieLens (ML-1M/ML-10M) [4] 包含丰富的物品元数据和用户评分随时间变化的信息，因此对于评估和开发适用于不同推荐场景的系统极具价值。Music4All [21] 包含丰富的用户-音乐交互数据，并提供精确的时间戳和曲目发行年份，支持对短期动态和长期时间演化的研究。GoodReads [26, 27] 收集了2017年末来自 goodreads.com 上公开用户书架的用户-图书交互数据。

5.0.2 评估协议。为了评估推荐性能，我们采用两种常用指标：HR@K 用于衡量至少有一个真实交互物品是否出现在排名前-K 的推荐物品中。该分数反映了推荐系统成功检索用户感兴趣物品的能力。NDCG@K 通过为排名位置靠前且排序正确的物品赋予更高权重，衡量推荐列表中排名前-K 个物品的排序质量。

5.0.3 基线。我们将提出的*HyperG* 与传统序列模型 [6, 10, 12, 25] 和生成模型 [5, 11, 14, 18, 29], 进行比较，同时也与因果插件去偏方法 [33, 37]进行比较。

传统序列模型：**SASRec** [10] 使用自注意力对交互中的长程依赖进行建模。**TiSASRec** [12]将交互之间的时间间隔纳入考量，以更好地捕捉对时间敏感的用户偏好。**门控循环单元** [6]是一种基于会话的推荐模型，利用门控循环单元（GRU）捕捉用户的序列行为。**USR** [33] 基于门控循环单元，使用逆倾向评分对选择偏差和曝光偏差进行去偏。**Attention** [25] 将自注意力机制应用于历史交互序列，并计算均值作为最终的用户偏好。**LDPE** [37] 采用由大语言模型增强的注意力架构来消除曝光偏差。

生成模型：**CVAE** [5] 采用条件变分自编码器，根据用户的历史交互来预测下一个交互物品。**T-DiffRec** [29] 根据时间戳对用户交互重新加权，以更好地捕捉时间动态。**PDRec** [18] 通过时间间隔的扩散过程对用户的动态偏好进行建模。**DiffuRec** [14] 利用去噪扩散概率模型，从复杂的用户-物品交互中学习。**DDRM** [11]利用基于扩散模型的多步去噪过程，增强嵌入的鲁棒性。

5.0.4 实现细节。我们在 { 5 5, 1 5, 5 6 8, e- e- e- } 范围内探索 16, 32 μ λ 0 05, 0 1, 0 2, 0 学习率，并搜索批量大小 { }。超参数分别从范围 { . . . .5, 1 0, 2 0 005, 0 1, 0 3, 05, 0 7, 1 0 . . } 和 { . . . . . } 中选取。所有实验均在两块 NVIDIA RTX-A5000-24GB GPU 上进行。

5.1 推荐性能（RQ1）

表2将*HyperG* 与7种基线模型进行了比较，其中包括传统序列模型 [10, 12, 25] [5, 和生成 11, 14, 18, 29] 模型。我们将*HyperG* 应用于三个骨干模型：一种基于注意力机制 [25], 的传统序列推荐器，以及两种生成式推荐器 CVAE [5] 和 Dif-fuRec [14]。

我们提出的*HyperG*框架提升了传统序列模型和生成模型在不同数据集上的推荐性能。 如表2所示，*HyperG* 在三种骨干模型上，于规模各异、场景不同的四个数据集上均取得了有竞争力的表现。对于基于注意力机制的传统序列模型， [25], 融入 TIPS 后（Attention+TIPS），HR@10 平均提升约 6%, NDCG@10 平均提升约 5%。CVAE [5] 和 DiffuRec [14]等生成模型

表2：*HyperG* 与两类序列模型之间的推荐性能比较。

模型	ML-1M				Music4All				GoodReads				ML-10M			
	HR@5	HR@10	NDCG@5	NDCG@10	HR@5	HR@10	NDCG@5	NDCG@10	HR@5	HR@10	NDCG@5	NDCG@10	HR@5	HR@10	NDCG@5	NDCG@10
传统序列模型																
SASRec	0.3351	0.4501	0.2249	0.2836	0.5581	0.6469	0.4119	0.4473	0.4814	0.5571	0.3622	0.3877	0.4755	0.6319	0.3164	0.3704
TiSASRec	0.3525	0.4819	0.2387	0.2968	0.5808	0.6924	0.4429	0.4772	0.4981	0.6017	0.3812	0.4113	0.4987	0.6721	0.3375	0.3924
门控循环单元（基线）	0.3467	0.4731	0.2482	0.2895	0.5460	0.6253	0.3983	0.4336	0.4607	0.5372	0.3501	0.3732	0.4593	0.6027	0.3061	0.3538
+USR	0.3728	0.5123	0.2658	0.3119	0.5762	0.6501	0.4173	0.4581	0.4763	0.5567	0.3627	0.3862	0.4941	0.6502	0.3260	0.3807
+*HyperG*（我们的方法）		0.5317	0.2808	0.3269	0.5932	0.6716	0.4322	0.4728	0.4970	0.5842	0.3940	0.4129	0.5377	0.7171	0.3605	0.4167
提升	↑4.35%	↑5.86%	↑3.26%	↑3.74%	↑4.72%	↑4.63%	↑3.39%	↑3.92%	↑3.63%	↑4.70%	↑4.39%	↑3.97%	↑7.84%	↑11.44%	↑5.44%	↑6.29%
注意力（基线）	0.3666	0.4951	0.2662	0.3075	0.5404	0.6592	0.4069	0.4453	0.4733	0.5738	0.3657	0.3982	0.6123	0.7503	0.4492	0.4921
+LDPE	0.3689	0.5099	0.2882	0.3236	0.5547	0.6903	0.4458	0.4691	0.4772	0.5983	0.3782	0.4230	0.6271	0.7652	0.4609	0.5117
+*HyperG*（我们的方法			**0.2944**	**0.3418**	**0.6155**		**0.4956**	**0.5325**	**0.5077**		**0.4057**	**0.4273**	**0.6491**		**0.4806**	**0.5355**
提升	↑4.20%	↑6.07%	↑2.82%	↑3.43%	↑7.51%	↑7.03%	↑8.87%	↑8.72%	↑3.44%	↑3.16%	↑4.00%	↑2.91%	↑3.68%	↑2.47%	↑3.14%	↑4.34%
生成模型																
DDRM	0.3999	0.5501	0.2545	0.3218	0.5984	0.7256	0.4639	0.5009	0.5027	0.6326	0.3652	0.4027	0.5324	0.7119	0.3709	0.4246
T-DiffRec	0.3336	0.4624	0.2206	0.2717	0.5339	0.6427	0.4179	0.4531	0.4873	0.5787	0.3628	0.4024	0.4444	0.6077	0.3091	0.3619
PDRec	0.3374	0.486	0.2259	0.2805	0.5453	0.6517	0.4207	0.4598	0.4879	0.5877	0.3669	0.4127	0.4712	0.6278	0.3127	0.3875
CVAE	0.3391	0.4503	0.2497	0.2898	0.4957	0.5967	0.3652	0.3999	0.4468	0.5435	0.3354	0.3735	0.5941	0.7242	0.4422	0.4685
+*HyperG*（我们的方法）		0.4954	0.2893	0.3366	0.5629	0.6623	0.4221	0.4457	0.4844	0.5799	0.3789	0.4123	0.6584	0.7786	0.4995	0.5088
提升	↑3.94%	↑4.51%	↑3.96%	↑4.68%	↑6.72%	↑6.56%	↑5.69%	↑4.58%	↑3.76%	↑3.64%	↑4.35%	↑3.88%	↑6.43%	↑5.44%	↑5.73%	↑4.03%
DiffuRec	0.4801	0.6311	0.3209	0.3665	0.6304	0.7341	0.4856	0.5294	0.5302	0.6673	0.3822	0.4267	0.7162	0.8327	0.5189	0.5927
+*HyperG*（我们的方法			**0.3505**	**0.4072**	**0.6844**		**0.5391**	**0.5817**	**0.5703**		**0.4169**	**0.4653**	**0.7591**		**0.5778**	**0.6171**
提升	↑3.72%	↑4.01%	↑2.96%	↑4.07%	↑5.40%	↑6.23%	↑5.35%	↑5.23%	↑4.01%	↑2.12%	↑3.47%	↑3.86%	↑4.29%	↑4.67%	↑5.89%	↑2.44%

也能从 TIPS 中获益，HR@10 相对提升约 5%。

TIPS 对序列推荐模型的性能提升在大规模数据集上比在小规模数据集上更为显著值得注意的是，*HyperG* 在提升 Music4All 和 ML-10M 上的性能方面表现尤为出色（见表2），表明*HyperG* 在多样化的大规模推荐场景中具有更显著的提升效果。例如，*HyperG* 的 HR@10 提升最高可达 8.87%，NDCG@10 也获得了同样显著的 8.72% 提升。我们认为，这些提升得益于 MusicAll 数据集中活跃用户较多且交互丰富。Music4All 上的用户参与度较高，平均每位用户每月有 5.76 次交互，显著高于其他三个数据集的平均值 1.14。通过利用更多用户交互，逆倾向评分（IPS）能够更准确地校正选择偏差，进而有助于缓解曝光偏差。这一过程有助于更全面地学习真实的用户偏好，从而显著提升推荐效果。

事实证明，融入时间信息对序列推荐有效。表2显示，融入时间信息的模型始终优于未对时间信息建模的对应模型。例如，TiSASRec 的 HR@10 和 NDCG@10 平均分别提升了 5.84% 和 6.39%。这与我们的动机一致：充分利用时间信息，以增强逆倾向评分的调整效果。同样，我们的 *HyperG* 也受益于对时间动态的建模，*HyperG* 取得了更大的提升。具体而言，在复杂的推荐场景（即 Music4All、GoodReads 和 ML-10M）中，采用 *HyperG* 的注意力机制表现与生成模型相当（HR@10 均超过 40%），且显著优于采用时间信息的传统序列模型（即 Ti-SASRec）。

表3：*HyperG* 在 ML-1M 上的消融研究。

骨干变体		ML-1M	
		HR@10	NDCG@10
Attention	*HyperG*（本文）	0.5558	0.3418
	HyperG ¬时间	0.5550 ↓0.08%	0.3472 ↑0.54%
	HyperG ¬逆倾向评分	0.5356 ↓2.02%	0.3238 ↓1.80%
	HyperG ¬EP&time	0.5311 ↓2.47%	0.2874 ↓5.44%
CVAE	*HyperG*（本文方法）	0.4954	0.3366
	HyperG ¬时间	0。4911 ↓0.43%	0。3311 ↓0.55%
	HyperG ¬逆倾向评分	0.4855 ↓0.99%	0.3219 ↓1.47%
	HyperG ¬EP&时间	0.4415 ↓5.39%	0.2757 ↓6.09%
扩散	*HyperG*（我们的方法）	0.6712	0.4072
	HyperG ¬时间	0.6581 ↓1.31%	0.3875 ↓1.97%
	HyperG ¬逆倾向评分	0.6537 ↓1.75%	0.3889 ↓1.83%
	HyperG ¬EP&time	0.6217 ↓4.95%	0.3578 ↓4.94%

5.2 消融研究（研究问题2）

我们针对*Hy-perG*进行了以下变体的消融研究：1）***HyperG*¬time** 从曝光估计和逆倾向调整中同时移除时间信息，同时保留整体框架，并采用端到端训练策略。2）***HyperG*¬IPS** 舍弃逆倾向调整，仅依靠 BPR 损失进行推荐。曝光估计模块使用二元交叉熵损失，结合时间信息进行预训练，整个框架则以端到端方式优化。3）***HyperG*¬EP&time** 同时移除曝光估计模块和时间信息，使用预训练的倾向得分，本质上是一种标准的基于逆倾向评分的方法。

表3展示了这些变体在 ML-10M 上的性能。可以看出，三个变体的表现均不如原始的 *HyperG*。这凸显了 *HyperG* 设计的必要性和全面性。变体 *HyperG*¬EP&time 移除了曝光估计模块和时间信息，仅依赖 IPS 调整，因此性能下降幅度最大

。这表明，无论数据集规模如何，曝光建模和时间信息对于推荐性能都至关重要，尤其有助于实现准确排序。相比之下，变体 *HyperG*¬time 和 *HyperG*¬IPS 对整体性能下降的影响较小。这种差异凸显了将时间信息与物品反事实曝光估计相结合的重要性和必要性，进一步验证了我们提出 *HyperG* 的动机。然而，与传统序列模型相比，生成式推荐器更能抵御移除时间信息所带来的影响。具体而言，变体 *HyperG*¬time 在扩散模型 [14] 和 CVAE [5], 中的 HR@10 平均仅下降 0.59% 和 1.03%，而注意力模型 [25]的降幅则更大，为 2.68%。这可能是因为生成模型能够捕捉类似时间信息的信号，为推断提供额外线索。

5.3 超参数研究（RQ3）

为进一步评估*HyperG*的有效性，我们在注意力骨干网络上计算*HyperG*在超参数 μ 0 05, 0 10, 0 20, 0 50, 1 00, 2 00 γ 0 05, 0 10, 0 20, 分别取 0 50, 1 00, 2 00 自{ . . . . . }和{ . . . . . }时相对于HR@10和NDCG@10的推荐性能。我们在以注意力为骨干网络的ML-1M上进行实验，并在图4(a)(b)中展示性能。

超参数 μ 是一个衰减比率，用于控制历史用户交互随其过时而失去影响力的速度。较小的 μ 会产生更慢的衰减，使长期行为能够持续为用户建模做出贡献，而较大的 μ 则仅强调最近的行为。如图4(a)所示，当 μ 从0.05增加到约0.20–0.50时，性能上升，表明适度强调新近性能够提升推荐质量，但进一步增大（当 $\mu \geq 1.0$时）会降低HR@10。这表明过于激进的衰减会丢失有用的长期信号。我们在实验中设定 $\mu = 0.5$ 。**超参数** γ 控制曝光估计损失L_{EP}对整体训练目标的贡献。当 γ较小时，优化器主要侧重于拟合已观测的历史交互，对曝光偏差的校正很少。图4(b)表明，当 γ增加到约0.10时，HR@10有所提升，说明适度纳入曝光信息能够增强推荐预测。本文将 γ 设为0.3。

5.4 对比分析（RQ4）

本节针对 *HyperG* 在 ML-1M 数据集上与传统逆倾向评分方法相比的性能进行了详细分析。我们在三个骨干推荐模型上评估了 *HyperG* 和逆倾向评分：Attention [25] 用于传统序列模型，diffusion [14] 用于生成模型。具体而言，我们随机抽取 100 名用户，并针对每名用户的 100 个物品记录由我们的 *HyperG*（通过公式（16）中的 $\pi^t(\cdot)$）以及传统逆倾向评分方法计算得到的倾向得分，其中包括一个来自测试集的正样本物品，以及从剩余物品池中随机抽取的 99 个负样本物品。图 4(c)(d) 展示了正样本物品的倾向得分与负样本物品平均倾向得分之差的分布。

倾向得分本质上估计了某个物品被交互的概率，同时还将曝光信息纳入考量，例如，该信息编码于 $\hat{\mathbf{e}}_i$ 公式（16）中。随后，三个骨干模型利用这些倾向得分来校正偏差。**如图 4(c)(d) 所示，*HyperG* 相较于传统逆倾向评分方法，能够持续为正样本物品和负样本物品提供区分度更高的倾向得分**。正样本物品具有更高区分度的倾向得分，有助于缓解曝光偏差。正样本物品倾向得分的更高区分度有助于缓解曝光偏差：差异越大，表明模型越能准确识别观测到的正样本，从而对曝光较少的物品进行适当的重新加权。这同样有利于缓解选择偏差，因为这体现了模型识别已曝光但未发生交互的物品，并通过逆倾向加权在最终目标函数中为其分配适当得分的能力。

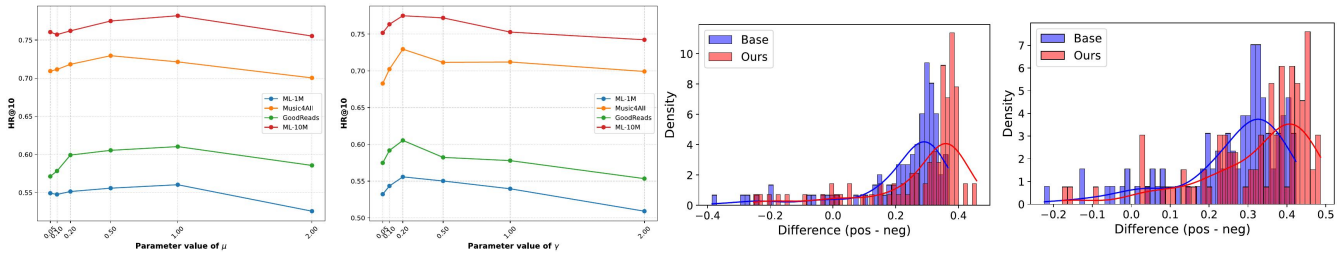
6 相关工作

6.1 序列推荐

序列推荐（SRs）旨在根据用户与物品的历史交互，预测用户的下一次交互。现有的序列推荐器大致可分为传统序列模型和生成模型。传统序列模型侧重于通过建模物品之间的顺序和时间依赖关系，从用户交互历史中学习模式。近期技术采用循环神经网络 [6, 10],、卷积网络 [9], 和基于注意力的 Transformer [24]等深度神经架构，从交互序列中捕捉用户偏好的序列信息。与之不同，生成模型将序列推荐视为生成问题，即模型根据用户历史直接生成下一个物品，并利用变分自编码器 [32, 34], 、扩散模型 [11, 29],和大语言模型 [7]等框架。例如，Li 等人 [14] 利用噪声和逆向过程，更好地捕捉用户偏好中的不确定性。生成模型比传统序列模型更能灵活适应用户偏好的变化，但二者都受到缺乏曝光日志的影响，导致对真实偏好的估计存在偏差。

6.2 序列推荐中的因果关系

基于因果关系的方法 [13, 38]表明，显式建模观测交互背后的因果结构是有效的，而非仅依靠模型参数捕捉相关性。为减轻因果结构中潜在混杂因素导致的偏差，许多此类方法采用逆倾向评分（IPS） [8, 19, 20, 23, 30] 对训练过程中的交互进行重新加权。Wang 等 [33] 利用逆倾向评分对训练数据中的交互重新加权，以消除序列推荐中的潜在混杂因素（如选择偏差）所带来的影响。Yu 等 [37] 将物品侧和用户侧的时间感知去偏倾向得分与大语言模型的语义理解能力相结合，以改进序列推荐。与我们的工作类似，CaseRec [42]也聚焦于系统曝光数据不足的问题，并利用反事实增强来探索潜在用户兴趣。尽管一些现有的基于因果关系的方法会纳入时间信息来处理偏差，但我们的做法有所不同：我们通过时间感知的反事实样本，为曝光估计提供明确指导。



(a) μ-HR@10 (b) γ-HR@10 (c) Attention (d) Diffusion

图 4：（a）（b）超参数 μ 和 γ 取不同值时的结果；（c）（d）使用*HyperG*（我们的方法）和传统逆倾向评分（基线）时，正样本物品与负样本物品的平均倾向得分之差的分布。

7 结论

本文提出时间感知逆倾向评分（*HyperG*），这是一种用于序列推荐的因果驱动框架，可缓解缺失曝光数据带来的偏差，尤其是曝光偏差和选择偏差。*HyperG*构建反事实物品–时间对，以估计具有时间动态的曝光分布，并通过时间敏感倾向得分对用户交互重新加权。我们在四个公开数据集上评估*HyperG*，并与两类最先进模型进行比较：（i）传统序列推荐器（基于循环神经网络/Attention）和（ii）生成模型（变分自编码器、扩散模型）。在三种骨干模型上，*HyperG*均能稳定提升性能。作为一种通用插件，它使现有序列推荐器能够在研究和实际应用场景中实现去偏，即使这些场景无法获得曝光数据。

参考文献

[1] Miaomiao Cai、Lei Chen、Yifan Wang、Haoyue Bai、Peijie Sun、Le Wu、Min Zhang 和 Meng Wang。2024。用于缓解流行度偏差的流行度感知对齐与对比。载于第30届 ACM SIGKDD知识发现与数据挖掘会议论文集（西班牙巴塞罗那）（KDD '24）。计算机协会，纽约州纽约市，美国，187–198。doi:10.1145/3637528.3671824[2] Wenqi Fan、Xiangyu Zhao、Xiao Chen、Jingran Su、Jingtong Gao、Lin Wang、Qidong Liu、Yiqi Wang、Han Xu、Lei Chen 等。2022。可信推荐系统综述。arXiv预印本 arXiv:2209.10117（2022）。[3] Chen Gao、Yu Zheng、Nian Li、Yinfeng Li、Yingrong Qin、Jinghua Piao、Yuhan Quan、Jianxin Chang、Depeng Jin、Xiangnan He 和 Yong Li。2023。面向推荐系统的图神经网络综述：挑战、方法与方向。ACM Trans. Recomm. Syst. 1，1，文章3（2023年3月），51页。doi:10.1145/3568022[4] F. Maxwell Harper 和 Joseph A. Konstan。2015。MovieLens数据集：历史与背景。ACM Trans. Interact. Intell. Syst. 5，4，文章19（2015年12月），19页。doi:10.1145/2827872[5] William Harvey、Saeid Naderiparizi 和 Frank Wood。2021。通过条件变分自编码器进行条件图像生成。arXiv预印本 arXiv:2102.12037（2021）。[6] Balázs Hidasi、Alexandros Karatzoglou、Linas Baltrunas 和 Domonkos Tikk。2015。利用循环神经网络进行基于会话的推荐。arXiv预印本 arXiv:1511.06939（2015）。[7] Jun Hu、Wenwen Xia、Xiaolu Zhang、Chilin Fu、Weichang Wu、Zhaoxin Huan、Ang Li、Zuoli Tang 和 Jun Zhou。2024。通过基于大语言模型的语义嵌入学习增强序列推荐。载于 ACM Web 会议2024伴随论文集（新加坡，新加坡）（WWW '24）。计算机协会，纽约州纽约市，美国，103–111。doi:10.1145/3589335. 3648307[8] Jin Huang、Harrie Oosterhuis 和 Maarten de Rijke。2022。物品越旧，情况越不同：选择偏差与用户偏好动态变化时的推荐去偏。载于第十五届 ACM 国际网络搜索与数据挖掘会议论文集（线上会议，美国亚利桑那州）（WSDM '22）。计算机协会，纽约州纽约市，美国，381–389。doi:10. 1145/3488560.3498375[9] Juyong Jiang、Peiyan Zhang、Yingtao Luo、Chaozhuo Li、Jae Boum Kim、Kai Zhang、Senzhang Wang、Xing Xie 和 Sunghun Kim。2023。AdaMCT：用于序列推荐的自适应 CNN-Transformer 混合模型。载于第32届 ACM 国际信息与知识管理会议论文集。976–986。[10] Wang-Cheng Kang 和 Julian McAuley。2018。自注意力序列推荐。载于2018年 IEEE 国际数据挖掘会议（ICDM）。IEEE，197–206。[11] Bahjat Kawar、Michael Elad、Stefano Ermon 和 Jiaming Song。2022。去噪扩散复原模型。神经信息处理系统进展35（2022），23593–23606。[12] Jiacheng Li、Yujie Wang 和 Julian McAuley。2020。时间间隔感知的序列推荐自注意力机制。载于第13届国际网络搜索与数据挖掘会议论文集（美国德克萨斯州休斯敦）（WSDM '20）。计算机协会，纽约州纽约市，美国，322–330。doi:10. 1145/3336191.3371786[13] Qian Li、Xiangmeng Wang、Zhichao Wang 和 Guandong Xu。2023。因果化：消除推荐中的社交网络混杂偏差。ACM Trans. Knowl. Discov. Data 17, 1, 14 2023 23 10 1145 3533725 文章（2024年2月），页码。doi: 。 / Zihao Li、Aixin Sun 和 Chenliang Li。 。 DiffuRec：用于序列推荐的扩散模型。ACM Trans. Inf. Syst. 42，3，文章66（2023年12月），28 10 1145 3631116 页。doi: 。 / Zihan Liao、Xiaodong Wu、Shuo Shang、Jun Wang 和 Wei Zhang。2024。通过因果干预对序列推荐中的动态物品倾向偏差进行建模。IEEE知识与数据工程汇刊（2024）。[16] Chengkai Liu、Jianghao Lin、Jianling Wang、Hanzhou Liu 和 James Caverlee。2024。Mamba4Rec：通过选择性状态空间模型实现高效序列推荐。CoRR（2024）。[17] Xiaoyu Liu、Jiaxin Yuan、Yuhang Zhou、Jingling Li、Furong Huang 和 Wei Ai。2025。CSRec：从因果视角重新思考序列推荐。。载于第48届国际 ACM SIGIR 信息检索研究与发展会议论文集。1562–1571。[18] Haokai Ma、Ruobing Xie、Lei Meng、Xin Chen、Xu Zhang、Leyu Lin 和 Zhan- hui Kang。2024。用于序列推荐的插件式扩散模型。载于 AAAI人工智能会议论文集，第38卷。8886–8894。[19] Teng Ma 和 Su Yu。2023。基于倾向得分估计的去选择偏差推荐算法。Applied Sciences 13，14（2023），8038。[20] Zohreh Ovaisi、Ragib Ahsan、Yifan Zhang、Kathryn Vasilaky 和 Elena Zheleva。2020。纠正学习排序系统中的选择偏差。载于 The Web Conference 2020论文集（中国台湾台北）（WWW '20）。计算机协会

[14] [15] [16] [17] [18] [19] [20] [21] Igor André Pegoraro Santana、Fabio Pinhelli、Juliano Donini、Leonardo Catharin、Rafael Biazus Mangolin、Yandre Maldonado e Gomes da Costa、Valéria Delisan- dra Feltrim 和 Marcos Aurélio Domingues。2020。Music4All：一个新的音乐数据库及其应用。载于*2020*年国际系统、信号与图像处理会议（*IWSSIP*）。399–404。doi:10.1109/IWSSIP48289.2020.9145170[22] Tao Qi、Fangzhao Wu、Chuhan Wu、Peijie Sun、Le Wu、Xiting Wang、Yongfeng Huang 和 Xing Xie。2022。ProFairRec：关注服务提供方公平性的新闻推荐。载于第*45*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（马德里，西班牙）（*SIGIR '22*）。计算机协会，美国纽约州纽约市，1164–1173。doi:10.1145/3477495.3532046[23] Lei Shi、Shuqing Li、Xiaowei Ding 和 Zhan Bu。2023。基于时间可见性，利用不感兴趣的项目缓解推荐系统中的选择偏差。*ExpertSyst.Appl.* 213, 2023 11 10 1016 2022 118932 PA（3月），[24] 页码。doi: 。/j.eswa.。Fei Sun、Jun Liu、Jian Wu、Changhua Pei、Xiao Lin、Wenwu Ou 和 Peng Jiang。2019。BERT4Rec：基于 Transformer 双向编码器表示的序列推荐。载于第*28*届 *ACM* 国际信息与知识管理会议论文集。1441–1450。[25] Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Łukasz Kaiser 和 Illia Polosukhin。2017。注意力就是一切。神经信息处理系统进展30（2017）。[26] Mengting Wan 和 Julian J. McAuley。2018。单调行为链上的物品推荐。载于第*12*届 *ACM* 推荐系统会议论文集（*RecSys 2018*），温哥华，不列颠哥伦比亚省，加拿大，*2018*年*10*月*2*日至*7*日，Sole Pera、Michael D. Ekstrand、Xavier Amatriain 和 John O'Donovan（编）。ACM，86–94。doi:10.1145/3240323.3240369[27] Mengting Wan、Rishabh Misra、Ndapa Nakashole 和 Julian J. McAuley。2019。从大规模评论语料库中进行细粒度剧透检测。载于第*57*届计算语言学协会年会论文集（*ACL 2019*），佛罗伦萨，意大利，*2019*年*7*月*28*日至*8*月*2*日，第*1*卷：长篇论文，Anna Korhonen、David R. Traum 和 Lluís Màrquez（编）。计算语言学协会，2605 2610 10 18653 1 19 1248 –。doi: 。/V/P - Lei Wang、[28] Chen Ma、Xian Wu、Zhaopeng Qiu、Yefeng Zheng 和 Xu Chen。2024。因果去偏时间感知推荐。载于*ACM Web* 会议 *2024* 论文集。3331–3342。[29] Wenjie Wang、Yiyan Xu、Fuli Feng、Xinyu Lin、Xiangnan He 和 Tat-Seng Chua。2023。扩散推荐模型。载于第*46*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集。832–841。[30] Xiaojie Wang、Rui Zhang、Yu Sun 和 Jianzhong Qi。2021。利用少量无偏评分应对推荐系统中的选择偏差。载于第*14*届 *ACM* 国际网络搜索与数据挖掘会议论文集（线上会议，以色列）（*WSDM '21*）。计算机协会，美国纽约州纽约市，427 435 10 1145 3437963 3441799 –。doi: 。/。Yu Wang、Zhiwei Liu、[31] Liangwei Yang 和 Philip S. Yu。2024。用于序列推荐的条件去噪扩散。载于知识发现与数据挖掘进展：第*28*届亚太知识发现与数据挖掘会议（*PAKDD 2024*），台北，台湾，*2024*年*5*月7日至*10*日，论文集，第五部分（台北，台湾）。施普林格出版社，柏林，海德堡，156–169。doi: 10 1007 978 981 97 2262 4_13 。/- - - - Yu Wang、[32] Hengrui Zhang、Zhiwei Liu、Liangwei Yang 和 Philip S. Yu。2022。ContrastVAE：用于序列推荐的对比变分自编码器。载于第*31*届 *ACM* 国际信息与知识管理会议论文集（美国佐治亚州亚特兰大）（*CIKM '22*）。计算机协会，美国纽约州纽约市，2056 2066 10 1145 3511808 3557268 –。[33] doi: 。/。Zhenlei Wang、Shiqi Shen、Zhipeng Wang、Bo Chen、Xu Chen 和 Ji-Rong Wen。2022。利用潜在混杂因素实现无偏序列推荐。载于*ACM Web* 会议 *2022* 论文集（线上会议，法国里昂）（*WWW '22*）。计算机协会，美国纽约州纽约市，2195–2204。doi:10.1145/3485447.3512092[34] Zhe Xie、Chengxuan Liu、Yichi Zhang、Hongtao Lu、Dong Wang 和 Yue Ding。2021。用于序列推荐的对抗与对比变分自编码器。载于*Web* 会议 *2021* 论文集（斯洛文尼亚，卢布尔雅那）（*WWW '21*）。计算机协会，美国纽约州纽约市，449–459。doi:10.1145/3442381.3449873[35] Wujiang Xu、Qitian Wu、Runzhong Wang、Mingming Ha、Qiongxu Ma、Linxun Chen、Bing Han 和 Junchi Yan。2024。在开放世界假设下重新思考跨域序列推荐。载于*ACM Web* 会议 *2024* 论文集。3173–3184。[36] Jiyuan Yang、Yue Ding、Yidan Wang、Pengjie Ren、Zhumin Chen、Fei Cai、Jun Ma、Rui Zhang、Zhaochun Ren 和 Xin Xin。2024。通过针对系统曝光进行分布鲁棒优化，为序列推荐系统去偏。载于第*17*届 *ACM* 国际网络搜索与数据挖掘会议论文集。882–890。[37] Dianer Yu、Qian Li、Sirui Huang、Jie Cao 和 Guandong Xu。2025。大语言模型与因果推断相遇：用于序列推荐的语义丰富的双重倾向评分。*IEEE*知识与数据工程汇刊（2025）

, 纽约 , NY, USA, 1863–1873. doi:10.1145/3366423.3380255

, 1– 12. d oi:10.1109/TKDE.2025.3606149

[38] Dianer Yu、Qian Li、Xiangmeng Wang和Guandong Xu。2023。基于因果干预的去混杂推荐。*Neurocomput.* 529, C（2023年4月），128–139。doi:10.1016/j.neucom.2023.01.089[39] Zhenrui Yue、Yueqi Wang、Zhankui He、Huimin Zeng、Julian Mcauley和Dong Wang。2024。面向序列推荐的线性循环单元。载于第*17*届*ACM*国际网络搜索与数据挖掘会议论文集（墨西哥梅里达）（*WSDM '24*）。计算机协会，美国纽约州纽约市，930–938。doi:10.1145/3616855.3635760[40] Honglei Zhang、Shuyi Wang、Haoxuan Li、Chunyuan Zheng、Xu Chen、Li Liu、Shanshan Luo和Peng Wu。2024。揭示去偏推荐中的倾向性识别问题。载于*2024 IEEE*第*40*届国际数据工程会议（*ICDE*）。IEEE，653–666。[41] Zihao Zhao、Jiawei Chen、Sheng Zhou、Xiangnan He、Xuezhi Cao、Fuzheng Zhang和Wei Wu。2023。流行度偏差并非总是有害：面向推荐解耦良性与有害偏差。*IEEE*知识与数据工程汇刊 35, 10（2023年10月），9920–9931。doi:10.1109/TKDE.2022.3218994[42] Ziqi Zhao、Zhaochun Ren、Jiyuan Yang、Zuming Yan、Zihan Wang、Liu Yang、Pengjie Ren、Zhumin Chen、Maarten de Rijke 和 Xin Xin。2025。通过对系统曝光进行反事实增强改进序列推荐器。载于第 *48* 届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（意大利帕多瓦）（*SIGIR '25*）。计算机协会，美国纽约州纽约市，1508–1518。doi:10.1145/3726302.3730005[43] Ziwei Zhu、Jianling Wang 和 James Caverlee。2020。在个性化排序系统中测量并缓解物品推荐不足偏差。载于第 *43* 届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（中国线上会议）（*SIGIR '20*）。计算机协会，美国纽约州纽约市，449–458。doi:10.1145/3397271.3401177

收稿日期：2007年2月20日；修订日期：2009年3月12日；录用日期：2009年6月5日