广西师范大学

研究生课程论文

课 程 名 称 《新技术讲座》

授 课 学 期 2022 学年至 2023 学年

第 一 学期

学 院 计算机科学与工程学院

专 业 计算机科学与技术

姓 名 陶仕儇

学 号 2022010585

任 课 教 师 李智

交 稿 日 期 2023.3.21

成 绩

阅读教师签名

阅 卷 日 期

推荐的因果推断:基础、方法与应用

陶仕傒

摘要：推荐系统是实现各种个性化服务的重要而强大的工具。传统上，这些系统使用数据挖掘和机器学习技术，根据数据中发现的相关性提出建议。然而，仅仅依赖相关性而不考虑潜在的因果机制可能会导致各种实际问题，如公平性、可解释性、稳健性、偏差、回音室和可控性问题。因此，相关领域的研究人员已经开始将因果关系纳入推荐系统来解决这些问题。在这项调查中，我们回顾了现有的文献中因果推理的推荐系统。我们讨论了推荐系统和因果推理的基本概念以及它们之间的关系，并回顾了现有的关于推荐系统中不同问题的因果方法的工作。最后，我们讨论了因果推断领域的开放问题和未来方向，并提出建议。

关键词：软件定义网络;边缘计算;网络架构

中图分类号：TP393　　**文献标识码：**A

Causal Inference for Recommendation: Foundations, Methods and Applications

TAO Shixuan

(Guangxi Normal University)

Abstract: Recommender systems are important and powerful tools for various personalized services. Traditionally, these systems use data mining and machine learning techniques to make recommendations based on correlations found in the data. However, relying solely on correlation without considering the underlying causal mechanism may lead to various practical issues such as fairness, explainability, robustness, bias, echo chamber and controllability problems. Therefore, researchers in related area have begun incorporating causality into recommendation systems to address these issues. In this survey, we review the existing literature on causal inference in recommender systems. We discuss the fundamental concepts of both recommender systems and causal inference as well as their relationship, and review the existing work on causal methods for different problems in recommender systems. Finally, we discuss open problems and future directions in the field of causal inference for recommendations.

Key words: Recommender System; Causal Inference

0 引言

推荐系统已被公认为缓解信息过载最有效的工具之一，并已广泛部署在许多现实世界的系统中，如电子商务平台(如 Amazon, eBay)、社交网络(如 Facebook, Twitter)、视频分享平台(如 Youtube, TikTok)和流媒体服务(如 Netflix, Hulu)。一般来说，这些系统使用先进的技术从历史数据以及收集的用户、项目和内容信息中学习用户的偏好。近年来，这些技术发展迅速。

一般来说，推荐算法可以分为三大类:协同过滤、基于内容的推荐和混合方法[1,2,3]。协同过滤(CF)模型基于一个关键思想:相似的用户可能有相似的兴趣，相似的物品可能被相似的用户所喜欢。早期基于记忆的 CF 模型，如基于用户的 CF[4,5]和基于物品的 CF[6,7]，以用户-物品评级矩阵的行向量或列向量作为用户和物品向量表示，基于预定义的相似度函数(如余弦相似度和皮尔逊相关系数)计算用户或物品之间的相似度进行推荐。为了从矩阵中提取潜在的语义含义，研究人员后来探索了习得的用户和项目向量表示。这始于潜在因子模型(LFM)，如矩阵分解[8]，概率矩阵分解[9]和分解机[10]，这些模型在实践中被广泛采用。在这些模型中，每个用户和物品都被学习为一个潜在的表示，以计算每个用户-物品对的匹配分数，通常基于内积。深度学习和神经网络的发展进一步扩展了 CF 模型。例如，[11,12,13,14]采用简单的用户和项目表示(例如，one-hot 向量)并学习了复杂的匹配函数。[15,16,17,18,19]学习复杂的用户和项目表示，并采用简单的匹配函数(例如，内积)。用户表示也可以直接从历史交互中计算出来，例如在顺序推荐中[20,21]。基于内容的推荐将利用用户和商品的丰富信息，甚至上下文信息来增强推荐。为了根据边信息学习物品之间的相似性，基于内容的推荐所采用的表示方法已经从简单的 TF-IDF[22]模型发展到基于深度学习的 DNN[23]、CNN[24]等模型。混合方法结合了协同过滤和基于内容的方法，利用了这两种方法的优点，避免了它们的某些局限性[1,2,25]。

传统推荐算法的基础是从数据中挖掘或学习相关模式。例如，许多协同过滤模型旨在学习用户-物品相关模式，一些基于内容的推荐模型旨在学习特征-特征相关模式。然而，现实世界的应用是由潜在的因果机制驱动的，单纯的相关学习而不考虑因果关系会导致一些实际问题。我们以经典的“啤酒和尿布”问题为例。纯相关学习会学习到啤酒和纸尿裤之间的强相关模式，从而为买了纸尿裤的顾客推荐啤酒，反之亦然。但潜在的机制是，年轻的父亲通常会同时购买啤酒和纸尿裤，在不考虑潜在机制的情况下推荐啤酒或纸尿裤会造成混淆，进一步损害用户的满意度。因此，从相关学习向因果学习过渡是非常重要的。

从形式上讲，因果推理研究的是因果关系，原因对结果负责。两个著名和流行的框架是潜在结果框架(也称为内曼-鲁宾潜在结果或鲁宾因果模型)[26]和结构因果模型(SCM)[27,28]。这两个因果框架都有助于因果推荐的发展。通过利用推荐系统中潜在的因果机制，因果推荐能够处理不同的实际问题，包括可解释性、公平性、鲁棒性、提升性和无偏性。

本次调查组织如下:第 1 部分介绍了推荐系统的初步情况。从第 2 节到第 6 节，我们将介绍因果推理的基本知识以及与推荐系统的联系。第 7 至 11 节分别介绍了现有的因果方法:可解释推荐、推荐公平性、基于提升的推荐、稳健推荐、无偏推荐。在第 12 节中，我们讨论了因果推理推荐中的一些悬而未决的问题和未来的发展方向。第 13 节总结了这一调查。

1 推荐系统的基本介绍

一般来说，推荐系统的目标是基于收集到的信息，包括用户简介、物品简介和用户-物品交互，对用户的偏好进行建模，并进一步预测用户未来的交互。用户档案代表用户的注册信息，可能包括用户 id、用户年龄、用户性别、用户收入等。推荐系统可能只使用部分信息进行推荐(例如，仅使用用户 id)。“物品”一词代表不同推荐系统中的不同对象(如电子商务中的产品、社交网络中的其他用户、在线视频平台中的视频等)。根据“项目”定义的不同，项目概况可能包含不同的项目特征。例如，电子商务中的产品可能会在项目简介中包含品牌、类别、价格、形象等;视频在在线视频平台项目简介中可能会采取视频长度、内容描述等方式进行视频推荐;社交网络中的其他用

户可以将相应的用户档案作为项目档案。同样，推荐系统也可以只提供部分项目概况信息进行推荐。交互是指用户根据定义的任务对物品可能的行为(例如，点击、购买、费率、加入购物车、电子商务推荐的评论、喜欢、不喜欢、视频推荐的分享等)。在一般的推荐系统中，交互通常表现为两种方式,一种是显式反馈,另一种是隐式反馈。显式反馈，如评分、评价，是用户偏好的显式表示(如评分为 5，表示用户喜欢该商品)，而隐式反馈,如点击,是用户与系统交互过程中收集的，隐式表示用户偏好(如用户的点击行为,表示用户很可能喜欢该商品)。

推荐模型根据收集到的信息学习用户的偏好，并根据习得的偏好进行推荐。具体来说，推荐系统将为特定用户提供个性化的推荐列表以及可能的解释。推荐系统首先会预测用户对一组候选商品的偏好。然后系统将对候选项目进行排序，提供个性化的推荐列表。值得一提的是，排名过程并不一定仅仅基于推荐算法提供的预测分数。根据不同的需求，如多样性、公平性、一些商业目的等，可以重新排列列表。在生成个性化的推荐列表后，一些推荐系统可能会在推荐的同时提供解释。解释可以与推荐同时生成，也可以在推荐之后生成，这取决于推荐模型是可解释的还是黑盒模型。

为了评估推荐系统的性能，定义一个好的推荐系统的特征并将其量化是很重要的。对于一个具有评分预测能力的推荐模型，一个优秀的模型应该能够预测准确的评分。因此，使用 RMSE 或 MSE 来评价推荐性能。通过考虑排名列表的准确性，以及列表中是否推荐了用户喜欢的物品，常用的指标有 Precision, Recall, F-Measure, NDCG, ROC Curve, AUC, MRR 等。除了上述用于评估推荐性能的指标外，还有一些指标用于从其他目的的角度评估推荐模型。例如，使用绝对差异(AD)[66]来评估推荐系统的公平性。

2 推荐中的因果符号

因果推断是源于统计学的一个重要研究课题[28,67,68]，几十年来被广泛应用于许多领域，如计算机科学、公共政策、经济等。在本节中，我们将介绍因果符号，并演示如何在推荐中应用它们。

2.1 什么是因果关系

因果关系是一个通常与相关性进行比较和讨论的术语。虽然相关性和因果关系都探讨了变量之间的关系，但众所周知,“相关性并不意味着因果关系”[68]。因果关系比相关性更进一步。直观上，因果关系明确适用于事件 A 导致事件 B 的情况。另一方面,相关性是一种非常简单的关系，即事件 A 与事件 B 相关，但一个事件不一定导致另一个事件的发生。例如，一项研究表明，每月冰淇淋销售的数据与美国每月鲨鱼袭击的数量高度相关。虽然这两个变量高度相关，但不可能得出食用冰淇淋导致鲨鱼袭击的结论(反之亦然)。更有可能的是，由于温暖的天气等其他因素，冰淇淋销量和鲨鱼袭击事件在夏季都有所增加，这导致两个变量相互关联。类似的例子可以在建议中找到。啤酒和尿布的故事是一个很好的例子，说明了推荐中因果关系和相关性之间的区别。有一种说法是，啤酒和尿布一起卖得很好。基于纯相关学习，啤酒和纸尿裤之间存在很强的相关模式,所以对于购买了纸尿裤的顾客应该推荐啤酒，反之亦然。然而，潜在的因果机制是，年轻的父亲们可能会在买啤酒的时候买一些尿布。因此，直接推荐项目而不考虑潜在的因果关系可能会导致混乱和推荐性能的下降。总的来说，理解因果关系有助于我们更好地理解世界是如何运作的，并可以提高推荐系统的性能。

要从理论上研究因果关系，就必须理解因果关系的数学表示。一般来说，因果推断有两种常用的框架，一种是潜在结果框架(也称为 Neyman-Rubin potential outcomes 或 Rubin causal Model)[26]，另一种是 Pearl 提出的结构因果模型框架[27,28]。现有的工作通常分别介绍两个框架，但我们认为这两个框架在逻辑上是等价的[28]，遵循相似的直觉。在接下来的章节中，我们将按照直观的因果关系来介绍这两个框架，包括两个框架的联系和区别。

2.2 因果关系中的关键数学符号

因果推断是指得出结论的过程，即特定的治疗是观察到的结果的“原因”[69]。

定义 1 潜在结果是个体在可能的处理下的结果

设 $X(X \in \{x_1, x_2, \cdots, x_n\})$ 表示处理,其中 n 为可能处理的总数。文献大多考虑二元处理，如吃药记为 $X = 1$，不吃药记为 $X = 0$。在二元处理

中，处理 $X=1$ 的个体组称为被处理组，处理 $X=0$ 的个体组称为对照组。一般情况下，x_i值治疗的潜在结果记为 $Y(X=x_i)$，可以简化为 $Y(x_i)$，x_i值治疗的平均潜在结果记为 $\mathbb{E}[Y(x_i)]$。对于任何个体，在保持其他变量不变的情况下，只能应用一种治疗方法，因此只能观察到一种潜在的结果。因此，潜在结果可以进一步分为两类， 观察到的潜在结果称为观察到的结果，其余未观察到的潜在结果称为反事实结果。

在推荐中，结果通常定义为用户行为(例如，点击，购买)或用户偏好(例如，评级)。无偏推荐模型将处理定义为暴露，其中观察到的反馈 Y(即观察到的结果)可以建模为两个未观察到的变量暴露 O 和相关性 R（即 $Y=O\cdot R$）的乘积[70,71,72,73,74]。更具体地说，在推荐系统中，$Y=0$ 既可以是负样本(即 $R=0$)，也可以是潜在的正样本(即 $R=1,O=0$)，这就导致了推荐的数据偏倚。为了实现个性化推荐，模型通常对某个用户-物品对 (u,v)（即 $Y_{u,v}=O_{u,v}\cdot R_{u,v}$）估计潜在结果 $Y_{u,v}$。通过正确估计潜在结果 $Y_{u,v}(O_{u,v}=1)$（即 $R_{u,v}=Y_{u,v}(O_{u,v}=1)\big)$，所设计的模型能够实现无偏推荐。基于提升的推荐模型将处理定义为推荐(即，1 为推荐，0 为不推荐)[75]。对于每个观察到的用户-项目对，只能观察到一种处理(即推荐或不推荐)。因此，如何估算反事实结果，计算推荐的提升值，是一个挑战。为了达到公平，也可以将处理定义为敏感属性[76](例如:，特权群体为 1，弱势群体为 0)。

除了处理变量和结果变量外，还可以观察到一些其他变量，这些变量可以进一步分为处理前变量和处理后变量[38]。

定义 2 处理前变量是不受处理影响的变量，也称为背景变量。

定义 3 处理后变量是受处理影响的变量。

不同的推荐场景可能包含不同的信息和因果机制，因此处理前变量和处理后变量的具体定义可能不同。

除了潜在的结果之外，Pearl 还从概率的角度使用 do-operation[27,68]来区分相关性和因果关系。假设 X 表示处理，Y 表示结果，相关性和因果关系追求不同的概率。具体来说，相关性从观测数据中估计条件概率 $P(Y\mid X)$，以确定 X 和 Y 之间的相关关系。相比之下，因果推断估计 $P\big(Y\mid do(X=x_i)\big)$表示可能的治疗 x_i 下的结果，其中 do-operation 直观地表示应用处理而不是观察处理。施用处理 x_i 的平均结果可用 $\mathbb{E}[Y\mid do(X=x_i)]$ 表示。特定概率 $P(Y=y\mid do(X=x))$ 可以简化为 $P\big(y\mid do(X=x)\big)$。正如我们之前提到的，现有的因果框架遵循相同的直觉，因此，在大多数情况下，操作和潜在结果的数学符号可以相互转换。例如，在无偏推荐模型中，处理通常定义为暴露。曝光下用户-物品对 (u,v) 的结果可以表示为 $P\big(Y\,\big|u,v,do(X=1)\big)$，其中 Y 是结果，X 是曝光变量。如果我们将变量 V 定义为暴露的项目，那么它也可以表示为 $P\big(Y\,\big|u,do(V=v)\big)$。同样，基于提升的推荐和推荐公平中的因果符号也可以用 do-operations 来表示。

通过定义行为操作，可以正式定义处理作为因果推理中的一个基本概念。如上所述，do-operation 表示应用处理，也可以定义为对处理变量的干预。我们将在第 7 节介绍更多细节。反事实是潜在结果框架和结构因果模型中的一个重要概念，它代表了与事实的差异。更具体地说，反事实表示处理变量与事实世界中的观察值相比具有不同的值。例如，考虑到治疗是服用药物，结果是康复，一个服用药物并康复的患者可能会想，如果他没有服用药物，他是否会康复。在这个例子中，在事实世界中，病人服药后康复，而在反事实世界中，病人没有服药，我们想知道他是否会康复。类似的例子可以在推荐系统中观察到，对于基于提升的推荐，处理定义为推荐，结果定义为用户行为，系统的目标是最大化推荐引起的用户行为增量。然而，在事实世界中，该项目不可能同时被推荐和不被推荐，因此，有必要将反事实应用到推荐中。反事实推荐系统已被广泛应用于解决实际问题，并取得了巨大的成功。我们将在这个调查中演示细节。

3 推荐中的因果假设

在本节中，我们将介绍因果推断中常用的假设。

定义 4 (Stable Unit Treatment Value Assumption(SUTVA))任何个体的潜在结果不随分配给其他个体的处理而变化，并且对于每个个体，每个处理水平没有不同的形式或版本，从而

导致不同的潜在结果。

这一假设强调了每个个体的独立性，这意味着个体之间没有相互联系。在推荐中，个人通常代表用户。传统的推荐隐含地假设了用户之间的独立性，这满足了 SUTVA 假设。然而，这种假设在实际的推荐系统中并不总是成立的。例如，在社交网络的推荐中，用户可以通过网络结构相互联系。一些推荐模型没有明确的用户，例如，基于会话的推荐。在这种情况下，个体可以被视为会话，它们暂时相互连接。

定义 5 (可忽略性)考虑到不受处理影响的变量 W，处理分配 X 与潜在结果无关，即 $Y(1),Y(0) \perp\!\!\!\perp X \mid W$ 。

可忽略性假设也被称为不可混淆性假设。这一假设确定了一定条件下的处理分配。具体来说，对于具有相同变量 W 的个体，处理分配是随机的。这一假设被许多推荐算法所接受，然而，在现实世界的推荐系统中，可能存在一些未观察到的变量影响着处理和结果，已有的著作对此进行了研究[77,78]。

定义 6 (积极性)对于不受处理影响的变量 W 的任意值，处理分配是不确定的：

$$P(X = x \mid W = w) > 0, \forall x \text{ and } w \qquad (1)$$

这一假设保证了评价处理效果的可行性和意义。如果对于 W 的某些值，处理分配是确定的，那么对于这些值，至少有一种处理的结果不能永远观察到。在这种情况下，估计处理效果是不切实际和毫无意义的。这个假设适用于推荐算法的设计。对于每个用户，每个项目都有机会向用户展示。不能公开的项目不在推荐系统的研究范围之内。

通过以上三个假设，可以建立观察到的结果与潜在结果之间的联系。

$$\begin{aligned}\mathbb{E}[Y(x) \mid W = w] &= \mathbb{E}[Y(x) \mid W = w, X = x] \\ &= \mathbb{E}[Y \mid W = w, X = x]\end{aligned} \qquad (2)$$

除了以上三个常用的假设外，还有另一种方法来表示假设的机制。

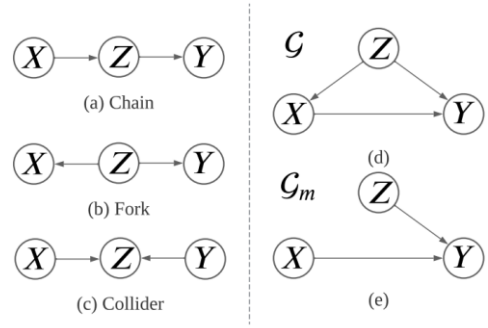

图 1 X,Y,Z 代表三个变量。(a)-(c)显示三个基本因果图。(d)表示因果图并举例，(e)表示对变量 X 进行干预时(d)的操纵图。

定义 7 结构因果模型(SCM)由一组内生 ($\boldsymbol{V}$) 变量和一组外生 ($\boldsymbol{U}$) 变量组成，由一组函数 ($\boldsymbol{F}$) 连接，这些函数根据 $\boldsymbol{U}$ 中的变量的值确定 $\boldsymbol{V}$ 中的变量的值。

SCM 是 Pearl 因果框架中的关键概念，它提供了关于场景背后机制的更强假设，它表明了除处理和结果之外的变量之间的关系。每个 SCM 都与一个图形模型 $\mathcal{G}$，相关联，表示为有向无环图(DAG)，其中每个节点是 U 或 V 中的一个变量，每条边是一个函数 f。每条边对应一个因果假设：如果变量 Y 是变量 X 的子变量，那么假设 X 是 Y 的直接原因；如果变量 Y 是变量 X 的后代，则假设 X 是 Y 的潜在原因。因果图是潜在结果框架和结构性因果模型框架的关键区别，潜在结果框架不考虑因果图来描述因果关系。然而，我们认为这两个框架都是建立在一些假设之上的，因果图只是一个更强的假设，它不能完全分离两个框架。我们在图 1 中介绍了三个基本的因果图。

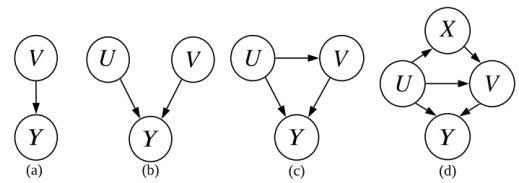

图 2 许多传统的推荐和因果推荐可以在不同的因果图下统一。图中 U 为用户，V 为项目，X 为用户交互历史，Y 为偏好评分。(a)非个性化模型的因果图。(b)基于相似性匹配的 CF 模型的因果图。(c)考虑用户与物品之间因果关系的因果图[82]。(d)[81]中使用的因果图。

因果图是表示推荐系统底层机制的一种直接方式，图 1 中三个典型的因果图经常出现在推荐系统的机制中。例如，图 1(a)的链式结构出现在

[77]中，商品决定商品的内在特征，商品的内在特征进一步决定用户的偏好;图 1(b)的分叉结构出现在[79]中，其中物品流行度被认为是物品曝光率和交互概率的共同原因;图 1(c)中的碰撞器结构出现在[80]中，其中用户点击是用户兴趣和一致性的共同结果。对于 SCM，已有的一项工作[81]表明，传统的推荐和因果推荐可以通过因果视图统一起来，其中推荐模型旨在估计$P(Y|U,do(X))$(即Y表示用户偏好，U表示用户，V表示商品)，但使用不同的因果图。更多的细节可以在图 2 中找到。

正如我们之前提到的，对处理变量的干预可以解释为对处理变量应用 do-operation。直观地说，do-operation 意味着直接干预，切断了其他变量对处理的影响。因此，考虑X和Y两个变量，可以直观地计算出期望的介入概率$P(y \mid do(x))$为$P_m(y \mid x)$，即操作图上观测到的概率。具体地说，经过处理的图形删除了处理变量的所有收入边。例如，考虑一个简单的因果图，如图 1 (d)，其中X是处理，Y是结果，Z 是混杂因素，原始因果图$\mathcal{G}$,上的$P(y|do(X))$与图 1(e)所示的操纵图$\mathcal{G}_m$上的$P_m(y \mid x)$相同。建议中的一个例子是对项目暴露进行干预,产生随机实验的数据(即数据生成过程遵循操纵因果图)。我们将在第 5 节介绍更多关于随机实验的细节。与对因果图的干预类似，对结构方程的干预以干预值作为输入，计算结构方程的输出。

引入的假设弥补了观测到的相关性和估计的因果关系之间的差距。我们将根据所介绍的假设，介绍一些常用的方法。

4 推荐中的因果效应

在介绍了因果表示的基本表示之后，可以使用基本表示定义许多不同类型的因果效果。一个基本的因果效应被称为处理效应(即,如果应用了另一种处理，结果会发生变化)。更具体地说，处理效果可以在人群、处理组、亚组和个体水平上衡量。这里我们定义了二元处理下的处理效果，以便明确，通过比较多种处理的潜在结果[38]，可以将其推广到多种处理。我们以潜在的结果为例，该操作可以以类似的方式应用。

在人群水平上的处理效果被称为**平均处理效果(ATE)**(一些参考文献也将其称为平均因果效应[68]或总效应[83])，其定义为:

$$ATE = \mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] \tag{3}$$

处理组水平的处理效果称为**处理组的平均处理效果(ATT)**(也有文献称其为处理组的处理效果(ETT)[27,68])，其定义为:

$$ATT = \mathbb{E}[Y(1) \mid X = 1] - \mathbb{E}[Y(0) \mid X = 1] \tag{4}$$

其中$Y(1)|X = 1$和$Y(0)|X = 1$表示处理组在两种处理下的潜在结果。

对于亚组水平，处理效果被命名为**条件平均处理效果(CATE)**，定义为:

$$CATE = \mathbb{E}[Y(1) \mid W = w] - \mathbb{E}[Y(0) \mid W = w] \tag{5}$$

其中W表示变量(即由多个变量分组)，定义不受处理影响的子组，$Y(1)|W = w$和$Y(0)|W = w$是$W = w$的子组中两种处理下的潜在结果。

在个体层面，处理效果被称为**个体处理效果(ITE)**，可以表示为:

$$ITE = Y_i(1) - Y_i(0) \tag{6}$$

其中$Y_i(1)$和$Y_i(0)$分别是个体i在$X = 1$和$X = 0$时的潜在结果。如果每个子组代表一个个体，则 ITE 被认为等同于 CATE[84,85]。

在推荐系统中，不同层次的处理效果被用作定量评价来处理许多问题。例如，ITE 用于估计推荐的隆起值[75,86,87,88];ATE 可用于评估解释[89]和估计无偏偏好[90];ATT 用于评估反事实公平性[91];等。

除了我们上面介绍的不同层次的处理效果外，还有一些因果效应用于中介分析。中介模型试图通过包含第三个变量(称为中介变量)来解释强调治疗和结果之间因果关系的过程。设X、Y、M分别表示处理、结果和中介。我们将引入二元处理下的三种效应进行中介分析。

首先，受控直接效应(Controlled Direct Effect, CDE)测量随着治疗变化Y的预期增长，而中介被设置为整个人群的特定值m，可以定义为:

$$\begin{aligned} CDE(m) = &\mathbb{E}[Y \mid do(X = 1, M = m)] \\ &- \mathbb{E}[Y \mid do(X = 0, M = m)] \end{aligned} \tag{7}$$

其次，自然直接效应(Natural Direct Effect, NDE)衡量的是随着治疗改变而预期的结果增加，而中介因素被设置为改变前的任何值，即 $X = 0$，可以定义为:

$$NDE = \mathbb{E}[Y \mid do(X = 1, M = M_0)] - \mathbb{E}[Y \mid do(X = 0, M = M_0)] \quad (8)$$

其中 M_0 表示被处理为 0 的中介的值。

最后，自然间接效应(NIE)测量当 $X = 0$ 时处理保持不变，而 M 改变为 $X = 1$ 时它将达到的任何值时，结果的预期增加，可以定义为:

$$NIE = \mathbb{E}[Y \mid do(X = 0, M = M_1)] - \mathbb{E}[Y \mid do(X = 0, M = M_0)] \quad (9)$$

其中 M_1 表示被处理的中介值为 1。NIE 捕获了仅通过中介就能解释的那部分效应。

上述直接和间接的影响在推荐中也起着重要的作用。直接效应和间接效应有助于模型定量评估路径特异性效应，以检测和消除不期望的效应。例如，它们可以用来识别直接和间接歧视，以实现或解释公平[92,93]，它们可以用来识别和消除一些偏见[94,95]等。

5 推荐中的因果估计方法

定义了因果关系后，下一个合乎逻辑的步骤是，我们如何估计这些影响。一种方法是进行随机实验。

5.1 随机试验

要衡量平均处理效果，理想的方法是对同一组个体采用不同的处理方法。然而，理想的解决方案在现实情况下是不切实际的。它只能通过随机实验得到近似结果。具体来说，一个随机实验将个体随机分配到治疗组和对照组。估计 ATE 可由两组平均结局的差值得到。为了理解为什么随机实验是估计平均处理效果的黄金标准，有必要理解相关性与因果关系的区别。

$$\begin{aligned} &\mathbb{E}[Y \mid X = 1] - \mathbb{E}[Y \mid X = 0] \\ &\overset{1}{=} \mathbb{E}[Y(1) \mid X = 1] - \mathbb{E}[Y(0) \mid X = 0] \\ &\overset{2}{=} \underbrace{\mathbb{E}[Y(1) \mid X = 1] - \mathbb{E}[Y(0) \mid X = 1]}_{\text{ATT}} \\ &+ \underbrace{\mathbb{E}[Y(0) \mid X = 1] - \mathbb{E}[Y(0) \mid X = 0]}_{\text{bias}} \end{aligned} \quad (10)$$

在这里，步骤 1 遵循的事实是，当条件作用于 $X = 1$ 时，$Y(1)$ 是观察到的结果，当条件作用于 $X = 0$ 时，$Y(0)$ 是观察到的结果;步骤 2 对 $E[Y(0)|X = 1]$ 进行加减法，得到 ATT 项和偏置项。(10)式中的偏差项造成了相关性和因果关系之间的差距。随机实验通过将个体随机分为处理组和对照组来消除偏倚项。更具体地说，随机分配使得潜在结果独立于处理 $Y(1), Y(0) \perp\!\!\!\perp X$（这并不意味着结果独立于处理），因此 $E[Y(0)|X = 1] = E[Y(0)|X = 0]$。给定 $Y(1)$，$Y(0) \perp\!\!\!\perp X$，式(10)可以改写为:

$$\mathbb{E}[Y \mid X = 1] - \mathbb{E}[Y \mid X = 0] = \mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] \quad (11)$$

因此，随机实验可以简单地将 ATE 估计为处理组与对照组的平均结局之差。在推荐中，通常使用随机实验来处理偏差[82,96,97,98,99,100,101,102]。具体而言，随机实验以项目暴露为处理，采用随机推荐策略，而不是部署策略，返回无偏数据(也称为均匀数据)进行推荐。

随机实验并不是万能的因果推理解决方案。在现实中，随机实验总是耗时且昂贵，因此研究通常涉及的个体数量较少，这可能不能代表总体。与此同时，伦理问题在很大程度上限制了随机实验的应用，如环境健康研究。此外，随机实验无法解释个体层面的因果关系。因此，鉴于观察性数据的广泛可用性，观察性研究是因果推断的捷径。

5.2 观测数据

虽然观察性研究可能是因果推断的捷径，但在设计因果模型时应仔细考虑观察性数据的一些问题。混杂因素的存在是观测数据中的一个关键问题。

定义 8 混杂因素是影响处理分配和结果的变量。

由于混杂因素的存在，可能会观察到一些伪效应(以冰淇淋消费与鲨鱼袭击之间的关系为例)。

混杂因素广泛存在于推荐系统中。混杂因素的存在往往会导致基于混杂因素定义的不同偏差。例如，将物品的受欢迎程度作为混杂因素，会导致受欢迎程度偏差[79]。除了一些可观察和可测量的混杂因素，如商品受欢迎程度，一些不可观察或不可测量的混杂因素(即违反第 4 节中的可忽略性假设)存在于现实世界的推荐中，并已被社区广泛研究[74,78,81]。

辛普森悖论是另一个可以在观测数据中观察到的现象。从表 1 中可以看出，无论男性组还是女性组，服用该药都有较好的康复率;但在总人口中，不服用药物的康复率更高。这种现象通常是由混杂因素引起的。辛普森悖论也可以在推荐系统中观察到[103]。

表 1
一项考虑性别因素的新药研究结果[68]

	使用药物	未使用药物
男性	81/87(93%)	234/270(87%)
女性	192/263(73%)	55/80(69%)
总数	273/350(78%)	289/350(83%)

Macdonald[103]观察到了线下推荐评价中的辛普森悖论，并提出了一种缓解线下评价悖论的方法。

与实验数据相比，观察数据只提供了发生了什么的信息，但为什么特定的处理是象征性的是未知的。由于处理分配机制未知，无法消除(10)式中的偏置项，也无法定量测量。因此，未知的处理分配带来的偏差也是模型设计中需要慎重处理的关键问题。

5.3 基于假设的方法

在一些复杂的情况下，基于先验知识假设因果机制是有风险的。在这种情况下，SUTVA、可忽略性和积极性假设支持一些估计潜在结果的方法。

一种常用的方法是基于重新加权的思想。如前所述，由于未知的处理分配机制，可能存在偏差问题。通过为观察数据中的每个样本分配适当的权重，可以创建一个伪总体，其中处理组和对照组的分布相似。有两种常用的权重调整方法:逆倾向评分和混杂因素平衡。

定义 9 倾向性得分被定义为给定背景变量的处理条件概率：

$$e(w) = P(X = 1 \mid W = w) \tag{12}$$

鉴于以上定义的倾向性分数，反倾向性评分方法[104, 105]根据倾向性分数给每个观察样本分配了一个权重。因此，基于观察样本的估计 ATE 可以改写为：

$$ATE_{IPS} = \frac{1}{n_1}\sum_{i,x_i=1}\frac{y_i}{e(w_i)} - \frac{1}{n_0}\sum_{j,x_j=0}\frac{y_j}{1-e(w_j)} \tag{13}$$

尽管使用倾向得分可以有效地减少偏差，但在实际应用 IPS 的过程中也有一些问题。首先，IPS 估计器的正确性高度依赖于倾向得分估计器的正确性。为了处理这个难题，人们提出了一些增强的 IPS 方法，如双重稳健估计器[107]。另一个缺点是 IPS 估计器有方差问题，即如果估计的倾向分数很小，估计器是不稳定的。为了克服这个缺点，一些方法提出了剪辑倾向分数[70]或修剪倾向分数小的样本[108]。

另一种重新加权方法是混杂因素平衡 [109、110、111]。动机是混杂因素可以通过矩来平衡，它唯一地决定了变量的分布。因此，可以学习样本权重以通过重新加权来估计因果效应。基于混杂因素平衡的方法用于稳定学习[112]和稳健推荐[113]。

除了重加权方法外，分层是另一种具有代表性的方法。分层的思想是将整个人群分成同质的子组，这使得每个子组中的处理组和对照组相似。理想情况下，在这种情况下，同一子组中的样本可以看作是从随机实验下的数据中抽样得到的。麦克唐纳 [103] 采用这种想法来减轻推荐离线评估中的辛普森悖论。

在某些应用中，因果机制是根据先验知识或专家知识安全地假设的。在这种情况下，因果机制可以表示为我们之前介绍的 SCM。尽管结构因果模型框架需要比潜在结果框架更强的假设，但它也可以通过图表进行推理。使用 SCM，因果关系和相关性之间的关键区别在于操作，这是估计因果效应的基本要素。正如我们提到的，do-operation 可以通过操纵图来估计。然而，操纵图中的数据是从随机实验中生成的。基于原始因果图生成的数据的方法在实践中很有用。应用后门调整是一种流行的方法。

定义 10 一组变量 Z 满足与因果图 $\mathcal{G}$ 中的一对有序变量 (X,Y) 相关的后门准则，如果 Z 满足两者 (1) Z 中没有节点是 X 和 (2) Z 阻塞 X 和 Y 之间包含指向 X 的箭头的所有路径。

通过识别一组满足后门准则的变量，可以使用后门调整公式来估计因果效应。

定义 11 如果一组变量 Z 满足与有序变量对 (X,Y) 相关的后门准则，并且如果 $P(x,z) > 0$，则因果效应 Y 上的 X 是可识别的，由下式给出

$$P\big(y \mid do(x)\big) = \sum_{z} P(y \mid x,z)P(z) \qquad (14)$$

给定观测数据的总体，如果我们根据 Z 值划分子组，则式（14）可以认为是通过每个子组的加权和来计算因果效应,这与分层方法非常相似。另外，式(14)可以改写为：

$$P\big(y \mid do(x)\big) = \sum_{z} \frac{P(y,x,z)}{P(x \mid z)} \qquad (15)$$

其中 $P(x|z)$ 被称为“倾向得分”，因此，后门调整也是 IPS 方法的另一种表示。后门调整广泛用于解决推荐中的问题，例如偏差问题[79、114]、回声室[115] 等。

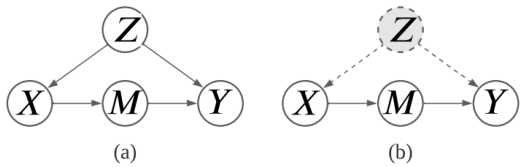

图 3 (a)在因果图上应用后门调整的示例。(b)具有未观察到混杂因素的因果图示例，其中可以通过前门调整来估计因果值。

当我们考虑 do 操作时，干预不限于强制一个变量或一组变量具有特定值的操作。一般来说，干预可能涉及动态政策，其中治疗变量 X 以特定方式响应其他变量的某些集合 Z，表示为 $x = g(z)$。在这种情况下，估计的因果效应 $P(Y = y|do(X = g(Z))$ 可以计算为：

$$\begin{aligned} &P(Y = y \mid do(X = g(Z))) \\ &= \sum_{z} P(Y = y \mid do(X = g(Z)), Z = z)P(Z = z \mid do(X = g(Z))) \\ &= \sum_{z} P(Y = y \mid do(X = g(Z)), Z = z)P(Z = z) \\ &= \sum_{z} P(Y = y \mid do(X = x), Z = z)\Big|_{x=g(z)} P(Z = z) \end{aligned} \qquad (16)$$

在推荐中，反馈数据是从部署的推荐算法中收集的，因此推荐策略存在于数据生成过程中。将动态策略视为推荐策略，条件干预也可以应用于设计因果推荐模型[81]。在推荐场景中，观察反馈数据中的交互并不意味着交互是注定要发生的，因此因果调整方法有时会与反事实推理一起应用[77,81,115]。

除了上述调整公式外，还有一些对介入概率有效的规则，称为 do-calculus 规则。在介绍具体规则之前，我们先介绍一些符号。设 X、Y、Z 和 W 是因果 DAG $\mathcal{G}$ 中任意不相交的节点集。$\mathcal{G}_{\bar{X}}$ 表示从 $\mathcal{G}$ 中删除所有指向 X 中节点的箭头得到的图。同样，$\mathcal{G}_{\underline{X}}$ 表示从 $\mathcal{G}$ 中删除得到的图从 X 中的节点出现的所有箭头。可以使用上述符号表示 do-calculus 的规则。

定义 12 以下三个规则适用于与因果图 G 兼容的每个介入分布

规则 1（观察的插入/删除）：

$$P(y \mid do(x),z,w) = P(y \mid do(x),w) \quad if\ (Y \perp\!\!\!\perp Z \mid X,W)_{\mathcal{G}_{\bar{X}}} \qquad (17)$$

规则 2（行动/观察交流）：

$$P(y \mid do(x),do(z),w) = P(y \mid do(x),z,w) \quad if\ (Y \perp\!\!\!\perp Z \mid X,W)_{\mathcal{G}_{\bar{X}\underline{Z}}} \qquad (18)$$

规则 3（动作的插入/删除）：

$$P(y \mid do(x),do(z),w) = P(y \mid do(x),w) \quad if\ \ (Y \perp\!\!\!\perp Z \mid X,W)_{\mathcal{G}_{\bar{X}\bar{Z(W)}}} \qquad (19)$$

其中 $Z(W)$ 是 $\mathcal{G}_{\bar{X}}$ 中不属于任何 W 节点的祖先的 Z 节点的集合。

在 do-calculus 规则和引入的调整公式的帮助下，可以通过观察数据来估计干预概率。

5.4 放宽假设的方法

尽管上述依靠引入假设的方法基本上满足了从观察数据中估计因果效应的要求，但在实践中，对于一些特定的应用，引入的假设可能并不总是成立。有一些方法试图用宽松的假设来估计因果效应。

SUTVA假设个人是独立的、相同的分布。然而，在一些现实世界的应用中，如社会网络，SUTVA 不能再成立了，因为个体之间通过网络结构有着内在的相互联系。为了处理实际应用中的这个问题，一个常用的方法是将捕捉到的相互联系的模型应用到因果推理模型中。例如，将图卷积网络应用于因果推理模型，以处理网络数据[116]。

无视性假设假定，在背景变量的情况下，治疗分配与潜在结果无关。然而，在现实世界中不可能识别和收集所有的背景变量，因此，可忽略性假设很难满足。换句话说，正如我们之前提到的，可能存在未观察到的混杂因素。仅仅使用观察数据来估计因果效应是困难的，另一种方法是将有限的实验数据和观察数据结合起来[117]。在推荐中，无偏数据是从随机实验中收集的，使用一小部分无偏数据和一大部分观察到的反馈是设计无偏推荐模型的一种流行方式。

另一个解决方案是基于假设的 SCM，它将未观察到的混杂因素建模到因果图中（图 3(c)中显示了一个例子）。与应用后门调整类似，我们首先确定一组满足前门标准的变量。

定义 13 （前门准则)给定因果图 $\mathcal{G}$ 中的一对有序变量（X，Y），如果 Z 满足以下条件，则一组变量 Z 相对于（X，Y）满足前门准则：

- **Z 拦截了从 X 到 Y 的所有定向路径。**
- **从 X 到 Z 没有未被阻止的后门路径。**
- **X 阻止了从 Z 到 Y 的所有后门路径。**

给定一组满足前门标准的变量，我们可以用未观察到的混杂因素来识别因果效应[68]。

定义 14（前门调整）如果一组变量 Z 满足与一对有序变量 (X, Y) 相关的前门标准，并且如果 $P(x, z) > 0$，那么 X 对 Y 的因果效应是可识别的，并由以下公式给出

$$P\big(y \mid do(x)\big) = \sum_{z} P(z \mid x) \sum_{x'} P(y \mid x', z) P(x') \quad (20)$$

业界普遍认识到未观察到的混杂因素的存在[74, 77, 78, 118]，有一些工作[77, 118]试图在推荐中应用前门调整。

使用工具变量是绕过可忽略性假设并进行因果推断的一种可能方式。工具性变量被定义为只通过处理变量影响结果的变量。典型的工具变量方法[119, 120]采用两阶段模型：第一阶段根据工具变量重建处理变量，第二阶段根据第一阶段的处理结果重建结果。在推荐系统中，Si 等人[121]采用工具变量来设计一个使用搜索数据的模型诊断推荐框架。

6 推荐中的因果发现

上述方法旨在学习因果效应，还有一个因果模型的分支，旨在学习因果关系，这也被称为因果发现。除了少数工作仅以识别治疗和结果为目的[122]，大多数工作的目的是发现因果图。按照[39, 123]，传统的方法可以分为三类：基于约束、基于 Socre 和基于功能的因果模型。

基于约束的算法学习一组满足数据中嵌入的条件独立性的因果图，并利用统计测试来验证候选图是否满足独立性。基于分数的算法通过最大化评分函数 $S(\mathbf{X}, \mathcal{G})$ 来学习因果图，该函数返回给定数据 $\mathbf{X}$ 的因果图 $\mathcal{G}$ 的分数。基于功能因果模型（FCM)的算法通常将变量定义为其有向原因和一些噪声项的函数（例如，由因果图的邻接矩阵线性加权[124]），并优化设计的目标以学习函数的参数。我们只简单介绍了因果发现方法，感兴趣的读者可以参考[39, 123]了解更多细节。

大多数现有的因果推荐工作都是基于预设的因果图，代表潜在的因果机制。预先定义的因果图通常是基于专家的知识来定义的，这可能是不准确的，而且相当简单（即只涉及几个变量）。在推荐中利用因果发现将处理这些问题。有少数工作[125, 126]在连续优化的基础上用因果发现技术设计推荐系统[127]。学习到的因果机制将增加推荐系统的可解释性，并指导其他方面的模型设计，如公平性、无偏见性。

7 推荐中的因果可解释性

随着机器学习的发展，准确性不再是唯一的追求。此外，透明度和可信度开始获得越来越多的关注。例如，医疗保健人工智能不仅需要提供准确的诊断，还需要提供支持性的解释来说服病人。有人类参与的推荐系统也需要透明度。可解释的推荐，是随着追求推荐系统的透明度和可信度而出现和发展的，在学术界和工业界都越来越受欢迎。它的目的是为推荐的项目提供解释，这将使社区在

许多方面受益。对于消费者来说，可解释的推荐能够帮助他们做出更好的决定。对于平台来说，它可以提高系统的透明度、说服力、可信度和用户的满意度。对于模型开发者来说，它是理解所设计的模型和加速设计周期的重要工具。在本节中，我们将首先介绍可解释建议的概况，然后总结现有的因果方法，以及一些与因果推理有关的开放性问题。

7.1 问题介绍

作为可解释人工智能的一个子领域，可解释推荐的研究是由[128]提出并定义的。随着深度神经网络的快速发展，先进的推荐系统广泛采用深度模型来提高推荐性能。然而，这些深度模型过于复杂，用户无法理解智能系统的决策，因此深度模型通常被认为是一个黑箱。推荐系统作为日常生活中必不可少的决策系统，需要提供准确的决策结果和基本原因。例如，一个股票投资者在做出最终决定之前，需要知道哪些特征导致了推荐。一个消费者希望在付款前了解为什么推荐的物品值得购买。

可解释的模型可以是模型内在的，也可以是模型无关的。前者是指与推荐结果同时生成解释，后者是指在提供推荐结果后生成解释。模型内在的（也被称为临时的）可解释模型通常将解释生成机制设计为决策过程的一部分，而模型不可知的解释（也被称为事后的）可解释模型通常设计单独的解释生成机制。

解释可以以许多不同的方式呈现，这通常取决于用于解释的信息源是什么样的。通常，解释可以以相关用户或项目[89, 129]、用户或项目的特征[128, 130]、生成的文本句子[131, 132]、视觉解释[133]、图表[134]等方式呈现。现有的工作在利用不同的信息源进行可解释的推荐方面取得了许多成功。例如，Zhang 等人[128]提出了显性因素模型（EFM），从用户评论中提取显性的项目特征和用户意见，以提供特征层面的解释。Peake 和 Wang[129]提取关联规则，以模型无关的方式提供购买的物品作为解释。Xian 等人[134]用知识图谱进行显式推理路径，以提供推荐和解释。此外，现有的工作已经将可解释性引入对话式推荐。Chen 等人[135]开发了一个可解释的对话式推荐（ECR）模型，通过多轮对话提供准确的推荐以及高质量的解释。融入因果推理思想和技术为可解释的推荐带来了新的机会。在下面的部分，我们将重点讨论与因果相关的方法。有兴趣的读者可以参考其他调查[29, 30]，了解更多可解释的推荐方法。

7.2 因果方法

近年来，反事实推理在可解释人工智能中引起了越来越多的关注。对于任何基于机器学习模型进行预测的人工智能系统，不管是白盒还是黑盒，反事实推理都要寻找哪些输入（例如，方面、特征）应该被改变，以及改变多少，以获得不同的预测。然后，改变后的输入将构成解释。例如，当为一个被拒绝的贷款申请产生解释时，它可能是这样的：如果你的年收入是 50,000，而不是 30,000，你的申请就不会被拒绝。一些现有的工作将反事实推理的思想引入到推荐场景中，用于生成解释，它寻找推荐系统中最小的变化（如项目特征、历史上的项目、用户的行为等）导致不同的预测，以确定最基本的部分（如项目特征、历史上的项目、用户的行为等）作为解释。Ghazimatin 等人[136]根据用户在历史上的行为为推荐系统生成解释。更具体地说，它在知识图谱上引入了一种搜索算法，以寻找用户历史记录的最小集合，从而使用户得到不同的推荐结果。Tan 等人[130]提出了一个用于生成特征级解释的反事实解释框架。它引入了两个新的概念，即解释复杂性和解释强度。这两个概念被用来制定一个反事实的优化问题，以及一个评价指标来评估生成的解释。后来在[137]中，一个类似的反事实解释框架也被用来解释哪些特征会导致推荐系统中的公平性问题。Tran 等人[138]利用一个影响函数来分析训练数据。然后，一组反事实的训练数据被用于生成解释。

基于因果发现的可解释推荐模型仍处于起步阶段。因果发现方法旨在从数据中提取各变量之间的因果关系。现有的基于因果发现的推荐方法提供模型内在的解释。更具体地说，通过提取的因果关系，基于因果发现的推荐模型能够同时提供相应的因果关系作为解释的建议。正如我们提到的，因果发现方法通常试图学习一个因果图。在推荐场景中，考虑到极大量的项目，学习的因果图通常是基于项目组层面的。例如，Wang 等人[125]提出学习集群级的因果图来指导顺序推荐。基于学习到的集群级因果图和每个项目的集群分配，该模型能够计算出项目之间的因果关系。交互历史中与被推荐项目具有最强因果关系的项目被确定为解释。Xu 等人[126]旨在学习产品类型（PT）层面的因果图，用于 PT 级推荐。特别是，该模型将收集的反馈数据作为两种完成机制的混合结果：

基于用户意图的因果机制和基于部署的推荐算法的干预机制。推荐和相应的解释是通过学习的 PT 级因果图产生的。

8 推荐中的因果公平性

推荐系统作为一种强大的商业工具，已经被广泛用于提高用户参与度,并进一步创造更高的利润。经典的推荐系统主要关心的是如何精确估计用户的偏好。然而，近年来，对推荐中公平性的关注引起了工业界和学术界的广泛关注[31, 32, 139, 140, 141]。随着推荐技术的发展，推荐系统已被广泛用于协助甚至取代若干领域的人类决策。一些研究表明，不公平可能导致负面的后果[142, 143, 144]，而这又可能产生重大的社会影响。例如，在电子商务中，物品曝光的不公平可能会损害平台和供应商的长期利益[145]；在教育推荐中[146]，由于性别不平衡[147]，一个不公平的系统可能会阻碍女性选择 STEM（即科学、技术、工程和数学）主题，这可能会影响社会几代人；一个不公平的广告推荐甚至可能导致种族歧视[148]。因此，为了增加推荐系统的应用并保持健康的社会影响，考虑推荐中的公平性并建立一个可靠的决策系统是至关重要的。

8.1 问题介绍

在实现推荐系统的公平性之前，首先应该了解不公平的原因。偏见和歧视是两个普遍接受的不公平的原因[31, 32, 33, 149]。推荐系统中的偏见主要包括数据中的偏见和算法中的偏见。数据中的偏见可能来自于数据的生成、收集、采样和存储。例如，在推荐系统中，训练数据是从已部署的系统中收集的，如果已部署的系统的基础算法做出有偏见的预测，那么生成的数据就可能涉及偏见。数据中的偏差可能会影响到算法，因为大多数机器学习算法都依赖数据进行训练，并在训练后做出预测。如果训练数据包含偏见，在这些数据上训练的算法将从这些偏见中学习到有偏见的知识，并进一步导致不公平。例如，如果训练数据显示多数派用户/项目组和少数派用户/项目组之间存在明显的不平衡，那么推荐算法很可能在多数派群体上学习得更好，并导致对少数派群体的歧视。除了数据中的偏见，推荐算法本身也可能加强现有的偏见，造成不公平，这就是所谓的算法中的偏见。例如，一些推荐算法可能会加强流行度的偏差，流行的项目会比质量相同或相似的不太流行的项目得到更多的推荐。歧视，作为一个多学科的问题[150, 151, 152]，也是不公平的一个原因，被定义为由于人类的偏见和刻板印象，基于任何物理或文化特征（如种族、性别等）的不合理的区别对待。值得一提的是，不公平不仅是由偏见和歧视造成的。例如，不同种类的公平之间可能存在冲突或权衡[31, 33, 153]，实现一种公平会损害另一种公平。

为了对抗不公平，定义公平性是很重要的。在一般的机器学习中，可以在目标层面上定义公平性（即在群体层面或个人层面上实现公平性）。具体来说，公平性可以分为群体公平性和个体公平性。

- **群体公平**：群体公平定义了群体层面的公平，它是基于不同群体应该被平等对待的想法。在这里，群体可以以多种方式划分，其中最常用的方式是根据一些明确的敏感属性来划分群体。
- **个人公平性**：个人公平定义了个人层面的公平，它是基于类似的个人应该得到类似的预测的想法。此外，个人公平在理论上可以被认为是一种非常特殊的群体公平，它将每个人分为不同的群体。

由于推荐系统中的公平性与来自多个利益相关者的利益相关[144、154、155、156、157]，公平性的要求可能来自不同方面。因此，推荐中公平性的定义也可以分为用户端公平性和物品端公平性。

- **用户侧公平**：用户侧公平旨在满足用户（消费者）的公平需求。用户侧的需求主要集中在推荐质量（即推荐性能）上。用户端的公平性可以在组级和个人级实现。用户组级别的公平性旨在减少不同用户组之间推荐质量的差异，其中用户组按敏感特征划分，例如种族或性别[142、158]，或按分配的特征（例如，冷用户与. 重度用户 [159]、活跃用户与不活跃用户[140、160]）。对于个人层面的用户端公平性，即使个人的敏感特征发生变化，推荐质量也应该保持不变。例如，李等人[139]结合反事实公平[91]的思想设计了一种推荐模型，即使用户的敏感特征在反事实世界中被翻转，推荐性能也不会改变。
- **物品侧公平性**：物品侧公平性旨在满足物品侧的公平性，主要是要求物品的曝光机会均等，以维护市场公平。这里的项目是指要排名或推

荐的“项目”。例如，在电子商务中，物品是指要销售的产品；在招聘系统中，物品是指求职者（物品提供者）。现有工作的一个分支侧重于根据项目属性实现公平。例如，一些工作[145、161、162、163、164、165]实现了流行和冷门项目之间的公平曝光，以防止冷门项目曝光不足。此外，研究工作的另一个分支主要侧重于根据项目提供者的敏感属性实现公平，例如性别［166、167、168]、地理出处［169、170、171］等。

值得注意的是，用户端公平性和项目端公平性可能并不排斥，双侧公平[172、173、174、175、176]方法被提出来满足双方的公平需求。除了上述分类法之外，还有一些分类法[31, 33]用于从其他角度对推荐的公平性进行分类。例如，静态公平与动态公平［143、143、177、178]；短期公平与长期公平［145, 179]；人口公平与个性化公平［139, 180, 181]；黑盒公平与可解释公平[137]，集中公平与分散公平［182、183]。

通常，所提出的实现推荐公平性的方法可以大致分为三类：预处理方法、处理中方法和后处理方法[31、33、149、184]。预处理方法通常旨在通过在模型训练之前最小化数据中的偏差来实现公平。与其他类型的方法相比，预处理方法的工作较少。一些有代表性的方法包括公平感知数据采样方法以覆盖所有组的项目，数据平衡方法[185] 以增加少数群体的覆盖范围和数据修复方法以确保标签正确性并消除不同的影响［186]。处理中方法建议将公平性要求作为目标函数的一部分，以在训练期间实现公平性。通常，公平性要求用作正则化器或约束［66、140、145、158、162、187、188、189、190、191]。为了在最小化原始损失函数（即推荐精度损失）的同时最小化不公平性，在推荐精度和公平性之间找到一个权衡点也很重要［145, 192]，这有时也被表述为多目标学习问题[192]。后处理方法旨在通过重新排序［140、193、194,195] 或多臂老虎机［196、197、198] 等技术在训练后的推理阶段实现公平性。为了衡量不公平性，提出了许多不同的公平性指标。例如，绝对差异（AD）[66]衡量保护组和未保护组表现之间的绝对差异； Normalized Discounted KLdivergence［199] 计算每个位置的 KL-divergence 的归一化折扣累积值等。更多可能的公平性指标可以在［32] 中找到。

最近，研究人员注意到，仅通过相关或关联无法很好地检测公平性。具体来说，公平标准仅基于随机变量的联合分布［200]，例如结果、特征、敏感属性等。然而，最近的工作［201] 表明，任何仅取决于联合概率分布的公平定义都是不一定能够检测到歧视。因此，许多方法［91、93、200、202、203] 被提出来通过因果关系的镜头来解决不公平问题。

在一般的机器学习中，基于因果的公平性符号主要是根据干预或反事实定义的。要衡量因果公平中的不公平，一个挑战是理解导致不同结果的因果关系。因果图作为因果推理的有力工具，通常用于表示变量之间的因果关系。给定捕捉因果关系的因果图，许多因果效应被用来衡量不公平性。例如，ATE（如式（3），也称为总效应［27]）用于衡量改变敏感属性对结果的影响，Kilbertus 等人［204] 衡量间接因果效应［205] 来自敏感属性到结果，并消除从敏感属性到结果的有向路径，除非通过解析变量，其中解析变量是指因果图中以非歧视方式受敏感属性影响的任何变量。在［76] 中可以找到更多基于因果的公平性符号的细节。反事实公平是因果公平中常用的公平定义。反事实公平是一种个人层面的基于因果的公平概念，它要求预测的结果在反事实世界中应该与任何个人在现实世界中的结果相同［91]。基本思想是最小化 ATT（如方程式（4），一些参考文献也将其命名为 ETT [27,68,132])，条件是所有特征在事实和反事实世界中接收相同的概率分布。对于推荐中的反事实公平性，定义如下[139]：

定义 15 (推荐中的反事实公平性)如果对于具有敏感属性 $Z = z$ 且剩余特征 $X = x$ 的任何用户 u ，则推荐模型满足反事实公平性：

$$P(L_z \mid X = x, Z = z) = P(L_{z'} \mid X = x, Z = z) \quad (21)$$

对于所有的 L 和 Z 可达到的任意值 z'，其中 L 表示用户 u 的 top-k 推荐列表。

在下一节中，我们将介绍一些实现推荐公平性的因果方法。

8.2 因果方法

正如我们之前提到的，偏见是一个被广泛接受的不公平原因，因此一些现有工作采用逆倾向评分（IPS）方法来解决推荐中的偏见。例如，流行度偏差会导致项目端的不公平，流行的项目可能会获得更多的曝光机会。IPS 接近偏差是由非随机分配的治疗引起的，因此使用逆倾向对样本重新加权以消除偏差。例如，Schnabel 等人［90] 将

推荐视为治疗，并在经验风险最小化框架中应用 IPS 估计器来学习解决推荐中的偏差。Saito 等人 [70] 设计了一个基于 IPS 的无偏配对学习估计器。Wang 等人 [106] 使用一小部分无偏数据来训练倾向模型，并使用有偏数据来训练基于 IPS 的评级模型。基于 IPS 的方法易于实施，但它需要准确的倾向估计器并且存在高方差 [206、207]。

尽管数据偏差通常被认为是推荐不公平的主要原因，但偏差与公平之间的关系尚未得到明确理解或讨论。更具体地说，通常会提出去偏方法来通过消除偏差来提高推荐性能，因此模型是通过推荐指标而不是公平指标来评估的。许多关于公平性的工作不是通过去偏方法实现的，而是直接根据公平性要求设计的，这可能会导致准确性和公平性之间的权衡。在接下来的部分中，我们将重点关注公平方法。更多关于去偏方法的讨论可以在第 12 节中找到。

反事实公平作为一种基于因果关系的公平定义，要求预测的结果在反事实世界中与在事实世界中相同。为了实现反事实公平，公平模型预测反事实世界中的结果（即敏感属性已更改）很重要但具有挑战性。Ma 等人 [208] 提出了一个反事实数据增强模块，该模块是基于训练的在具有公平约束的变分自动编码器上，生成具有不同敏感属性的反事实数据。通过最大化从原始数据中学习到的表示与不同的反事实数据之间的相似性，所设计的模型能够实现反事实公平。Mehrotra 等人 [209] 使用反事实估计根据相关性和公平性之间的权衡来评估推荐策略，并提出了一个考虑用户对公平性的容忍度的推荐模型。反事实思想不仅用于公平模型设计，还用于公平诊断。具体来说，公平性诊断旨在找出导致模型不公平的原因。受反事实解释 [130、210] 思想的启发，Ge 等人 [137] 提出了一种反事实推理方法来学习显着影响公平效用权衡的关键特征，并将它们用作基于特征的推荐的公平解释。

结构因果模型由捕捉变量之间直接因果关系的因果图和建立变量之间定量关系的一组结构方程组成。正如我们在第 6 节中介绍的，如果给定结构方程，则可以通过替换结构方程中的变量值来获得干预或反事实结果。受这个想法的启发，一些关于公平的工作利用了学习或预定义的结构方程。例如，一些工作 [92、211、212、213、214、215] 从学习的结构方程中模拟不同的因果效应，以发现歧视并进一步消除它们。Kilbertus 等人 [204] 开发了一个实用的程序来消除给定结构方程模型的歧视。

其他一些基于因果关系的方法利用因果图来捕获底层数据生成机制并应用其他技术来实现公平性。例如，Huang 等人 [216] 使用从因果图中识别的 d-separation 集来设计用于在线推荐的公平置信上限 bandit 算法。Li 等人 [139] 设计了一个基于因果图的模型，通过对抗学习生成与特征无关的用户表示。具体来说，该模型同时训练一个预测器和一个对抗性分类器，其中预测器学习推荐的表示，而分类器最小化预测器预测敏感特征的能力。

8.3 未决问题

正如我们上面介绍的，研究人员开始意识到在推荐中考虑基于因果关系的公平性的重要性 [76, 201]。然而，推荐中因果公平性的基础还没有很好地建立起来。具体来说，公平技术在分类任务中得到了很好的探索，但是，即使在某些情况下可以将推荐视为分类任务，这些技术也可能不会直接迁移到推荐问题。例如，一种在分类中实现反事实公平性的直接方法 [91] 是从输入中删除敏感属性，以保证结果与敏感特征之间的独立性。然而，在推荐系统中，一些现有的方法不使用特征进行推荐，例如大多数基于协同过滤的模型 [217]，但仍然存在不公平性。原因是交互信息包含敏感特征和用户-项目交互之间的隐藏关系，这种潜在关系会在协同学习过程中被模型捕获，从而导致不公平。因此，对不公平的潜在因果机制进行更多探索至关重要。此外，它可以帮助社区在偏见和公平之间建立联系。

9 推荐中的因果鲁棒性

最近，机器学习的鲁棒性变得越来越重要。由于模型时间非常耗时，推荐系统模型在实践中并没有经常重新训练。传统上，推荐系统假设训练数据集和测试数据集的模式相同。但是，训练数据集和真实世界数据之间存在差异。这种差异可能是由自然分布转移或意图攻击引起的[218]。当我们将模型应用于真实世界数据时，对这样的训练数据集进行训练将导致性能下降。在这种情况下，如何构建一个健壮的模型就显得非常重要了。

9.1 问题介绍

首先，我们需要知道哪些方面会损害推荐系统的稳健性。一般来说，数据集在训练过程中会被分成三个子集（训练集、验证集和测试集）。大多数鲁棒性发生在训练集和测试集上。例如，如果训练数据集不够大，这可能会导致过拟合或欠拟合问题。在这种情况下，我们可能会在测试集上得到不好的结果。具体来说，鲁棒性问题可以分为以下几类：

- 分配转移：许多现有的推荐系统假设训练集和测试集的分布是相同的。然而，这种假设不符合现实世界的场景，这使得许多现有的推荐模型无法达到我们在线部署它们时所期望的性能[219]。许多当前的推荐系统模型都是基于现有收集的数据集进行训练的。部署新模型时，数据的分布可能会有所不同[220]。即使数据是新收集的，仍然存在转换风险 [218]。因为训练模型和部署模型之间有一个时间段。由于处理错误，收集的信息被攻击或更改的时间足够长。
- 转换：大多数推荐系统都使用用户和项目的特征进行训练。基于该特征，推荐系统可以为用户提供一组推荐项目。但是，用户和项目的特征可能会被破坏或误导。例如，如果用户在购买商品时使用 VPN，则位置信息可能是错误的。有了这些数据，推荐系统可能会向用户提供错误的项目。
- 攻击：大多数推荐系统都使用用户和项目的特征进行训练。基于该特征，推荐系统可以为用户提供一组推荐项目。但是，用户和项目的特征可能会被破坏或误导。例如，如果用户在购买商品时使用 VPN，则位置信息可能是错误的。有了这些数据，推荐系统可能会向用户提供错误的项目。
- 稀疏：大多数推荐系统都使用用户和项目的特征进行训练。基于该特征，推荐系统可以为用户提供一组推荐项目。但是，用户和项目的特征可能会被破坏或误导。例如，如果用户在购买商品时使用 VPN，则位置信息可能是错误的。有了这些数据，推荐系统可能会向用户提供错误的项目。

9.2 鲁棒性方法

推荐模型存在数据稀疏性问题。例如，在电子商务应用中，相对于大量的用户和物品，用户的购买历史是稀疏的。因此，模型无法获得足够的数据进行训练，导致预测性能低下。解决此问题的一种方法是反事实数据增强。通过使用反事实推理生成新的训练数据，并与原始数据一起可以增强推荐模型的性能。反事实数据增强序列推荐（CASR）为我们提供了一个解决问题的框架[221]。对于训练样本 $(\{u,t^1,t^2,\dots t^l\},t^{l+1})$，模型将首先指示索引 d，并将 t^d 替换为项目 t^a 。假设 $e_t \in R^D$ 是项目 t 的嵌入，D 是项目的嵌入大小。对于给定的样本 $(\{u,t^1,t^2,\dots t^l\},t^{l+1})$，模型将优化以下对象：

$$\min_{t^a \in C} \|\mathbf{e}_{t^a} - \mathbf{e}_{t^d}\|_2^2$$
$$s.t.\ t^{l+1} \neq \arg\max_{t \in I} \mathcal{S}(t \mid u, t_1, \dots, t^{d-1}, t^a, t^{d+1}, \dots, t^l) \quad (22)$$

其中 $\mathcal{I}$ 是所有项目的集合。C 是要替换的项目集，可以指定为 $\mathcal{I}$ 或其他涉及一些先验知识的集合。$\mathcal{S}$ 是用于生成新的顺序数据的采样器。在此函数中，对象尝试最小化原始项目和替换项目之间的距离。并且约束确保更改的项目不是原始项目。在这种情况下，我们可以生成与原始数据不同但与原始数据相似的数据

一些现有的工作引入了因果表示学习的思想来缓解分布转移问题。该模型将用户特征分为观察组和未观察组，并根据是否受到观察特征的影响设置两种类型的偏好[222]。根据因果图，创建了一个框架来模拟交互生成过程。为了处理未观察到的特征，他们设计了一种新的变分自动编码器 (VAE)，以从历史交互和观察到的特征中推断出未观察到的特征。

9.3 未决问题

现有的关于鲁棒性问题的工作只能关注一些特定的问题。例如，使用反事实数据扩充问题来缓解稀疏性问题，使用因果表示学习来解决分布偏移问题。如果我们将这些方法应用于其他鲁棒性问题，实验性能可能会下降很多。而目前现存的大部分工作都是无法解释的。他们可能在解决某些问题上有很好的表现，但他们无法解释模型的哪一部分提高了性能，模型的哪一部分可以有更多的改进。一种可应用于多个问题的解释稳健性方法是一个巨大的挑战。

10 基于提升的推荐

现代推荐系统通常旨在推荐用户最有可能与之交互的项目（例如，点击、购买等）。但是，即使没有推荐，用户也可能会与某些项目进行交互。基

于这个事实，一些现有的工作建议推荐具有高交互概率提升而不是高交互概率值的项目。

一个密切相关的领域是提升建模，它指的是用于估计处理对结果的增量影响的技术。提升建模既是因果推理又是机器学习问题[225]。这是一个因果推理问题，因为计算增量影响所需的两个结果（即接受治疗或不接受治疗）对个人而言是唯一的。这也是一个机器学习问题，因为它需要模型来预测可靠的提升值以进行决策。从理论上讲，提升模型旨在估计治疗对结果的影响 [225]。现有文献中主要有三种方法：Two-Model 方法分别在已处理数据和受控数据上训练两个模型，并使用两个预测之间的差异来计算提升值[226]；类别转换方法基于一些假设 [227] 建立了治疗组和控制组之间的联系；直接估计方法设计了一个模型来直接估计提升值 [228, 229]。

10.1 问题介绍

推荐系统已应用于多个工业领域，以增加企业利润并提高用户参与度。为了实现这一目标，大多数推荐模型旨在通过推荐具有最高交互概率的项目来增加用户操作（例如，点击、购买等）。然而，大多数推荐系统忽略了一个事实，即用户可能会对某些项目采取行动，而不管系统是否推荐它们[75, 230]。例如，如果系统推荐用户购买瓶装水的概率为 95%，能量饮料的概率为 50%。在大多数传统推荐系统看来，推荐瓶装水会更好，因为它更有可能被用户购买。但是，如果系统不推荐的话，瓶装水作为日常用品，可能还有 90%的概率被购买，而能量饮料可能只有 20%的概率被购买。推荐能量饮料似乎是一个更好的选择，因为它具有更高的购买概率提升（即 30% 对 5%），这反过来可能会带来更多利润。基于这种动机，有一种趋势是设计基于提升的推荐系统，旨在推荐具有高提升的项目。

之前的一些工作已经意识到推荐的影响，但没有从因果关系的角度解决它。例如，Bodapati [231] 提出了一个两阶段模型，分别训练项目的意识和满意度阶段。通过基于公司发起的购买数据（即，由于推荐而进行的购买）和自发购买数据（即，公司发起的购买以外的购买）训练模型，该模型旨在推荐能够最大化预期增量数量的项目从推荐购买。Sato 等人 [230] 提出了一种购买预测模型，该模型结合了推荐响应中的个体差异。更具体地说，该模型包括用户特定和项目特定的响应，以最大化推荐的影响。

推荐系统的提升被定义为由推荐引起的用户操作（例如，点击、购买等）的增加。考虑到提升被定义为有推荐和没有推荐的情况之间的差异，从因果推理的角度来看，提升可以用潜在结果在数学上表示。更具体地说，以推荐为处理，令 $Y(1)$ 为有推荐的潜在结果，$Y(0)$ 为无推荐的潜在结果。考虑二元情况，$Y(1) = 1$ 和 $Y(0) = 1$ 分别意味着用户将在有推荐和无推荐的情况下对项目采取行动。一个项目对用户的提升是 $Y(1) - Y(0)$。在下面的小节中，我们将介绍一些现有的基于提升的推荐模型的因果推理工作。

10.2 因果方法

估计提升值的一个挑战是每个人都无法观察到事实和反事实结果（即有和没有建议的结果）。因此，提升值没有观察到的基本事实（即推荐的因果效应）。为了克服这个问题，一种可能的解决方案是关于训练数据。Sato 等人 [75] 提出了一种基于提升优化的观测数据采样方法。具体来说，通过观察购买和推荐日志，对于给定的用户，可以购买或不购买商品，推荐或不推荐商品。提议的优化从四类项目(即推荐和购买、推荐和不购买、不推荐和购买、不推荐和不购买)中抽取特定于提升任务的正面和负面实例。因此，通过将提升任务的采样标签作为基本事实，所提出的优化能够学习用户-项目对的提升值。除了观测数据的采样方法外，实验数据的训练也是一种可用的选择。Shang 等人 [86] 提出了一种基于强化学习的方法，该方法结合了深度提升网络来学习不同动作的因果效应作为奖励函数。提升网络从随机实验收集的训练数据中学习。

根据提升值的计算，一种直接的方法是估计反事实结果。尽管随机实验是估计因果效应的理想选择，但由于耗时且昂贵，将随机实验应用于所有推荐场景是不切实际的。因此，必须仅根据观察数据来估计反事实结果。受协同过滤思想的启发，Xie 等人 [87] 认为相似的用户对物品的品味相似，在推荐下的处理效果也相似。所提出的方法是基于张量分解设计的，具有用户、项目和治疗三个维度。更具体地说，对于具有 m 个用户、n 个项目和 l 个处理的三维张量，可以按如下方式预测元素 $y_{u,i,t}$。

$$\hat{y}_{u,i,t} = p_u^T q_i + p_u^T d_t + q_i^T d_t \tag{23}$$

其中 p_u、q_i、d_t 分别是用户 u、项目 i 和处理 t 的潜在表示。$\hat{y}_{u,i,t}$ 的预测值用于推断处理 t 下用户-项目对 (u,i) 的潜在结果。以二元处理

设置为例，用户-项目对 (u,i) 的提升值可以通过 $\hat{y}_{u,i,t=1}-\hat{y}_{u,i,t=0}$ 来估计。Sato 等人 [88] 应用匹配估计器 [232] 来估计未观察到的反事实结果，并进一步估计推荐的因果效应。更具体地说，根据推荐系统中的邻域方法，所提出的方法将潜在结果替换为一组邻居的观察结果的加权平均值，以计算因果效应，其中邻居可以是邻域用户或邻域项目。

仅根据观测数据估计因果效应具有挑战性，因为基本事实是不可观测的，而且估计容易出现观测数据的偏差。为了克服这个问题，一些现有的工作设计了基于 IPS 的方法来估计推荐或评估的无偏因果效应。提升值的无偏估计（即因果效应）可以由 IPS [233] 制定。在实践中，IPS 容易出现高方差问题。为了解决这个问题，Sato 等人 [233] 应用上限逆倾向评分（CIPS）来训练一个无偏的基于提升的模型;Sato 等人 [75] 提出了一种无偏估计量，用于使用自归一化逆倾向评分(SNIPS)[234] 进行基于提升的评估； Xiao 和 Wang [235] 应用双重鲁棒技术 [107、236] 来训练一个无偏且鲁棒的模型，用于基于提升的推荐。

10.3 未决问题

现有的基于提升的推荐工作主要集中在表示提升值和使用潜在结果框架估计因果效应。结构因果模型作为因果推理的强大工具，很少用于基于提升的推荐。如果系统推荐某个项目，则使用结构因果模型的现有工作正在尝试估计用户的偏好，这可以通过对设计的因果图进行操作来估计。然而，目前还不清楚如何使用 do-operation 在没有推荐的情况下估计偏好。首先，结构因果模型需要设计的因果图。现有的使用结构因果模型进行因果推荐的工作很少明确地将推荐的影响纳入因果图中，但是对于基于提升的推荐，是否需要特定的因果图来明确描述推荐的影响仍然需要讨论。其次，使用 do-operation 在没有推荐的情况下进行偏好的数学表示也是一个挑战。最后，对于基于提升的推荐，如果设计的因果图和没有推荐的偏好的数学表示被确定，那么在没有推荐的偏好上应用因果技术可能与现有工作不同。

11 推荐中的因果公正性

如今，推荐算法已广泛应用于多种应用程序，以减轻我们日常生活中的信息过载。尽管推荐系统 (RS) 在广泛的现实应用中获得了巨大的影响，但它仍然面临许多具有挑战性的偏差问题，如果置之不理，将影响推荐系统的长期效益。偏差问题在 RS 中很常见，因为 RS 的一个特性是反馈回路。按照普遍接受的理解 [35, 36]，RS 中的反馈回路从鸟瞰图上可以分为三个部分：1）数据收集部分（用户→数据）； 2）模型训练部分（数据→模型）； 3）模型服务部分（模型→用户）。每个部分和整个反馈循环中都存在对偏差问题的不同定义。我们将在以下部分介绍更多细节。

11.1 问题介绍

正如我们所提到的，偏差问题存在于每个部分以及整个反馈回路中。下面分别介绍 RS 的反馈回路中 bias 的不同定义：

- **数据偏差**：是指收集到的用于训练的数据与理想的测试数据之间的分布差异。通常，RS 的训练数据是观察性的而不是实验性的。用户决策可能受到 RS 暴露机制等多种因素的影响，因此训练分布与测试分布不同。此外，训练数据可能无法真正代表用户偏好，误导推荐模型以做出不准确的预测。下面我们将介绍四种数据偏差：
  - **选择偏差**：选择偏差源于用户的明确反馈（即评级）。选择偏差是指由于用户的选择，观察到的评分并不代表所有评分。它也被称为非随机缺失 (MNAR)。
  - **曝光偏差**：曝光偏差通常发生在带有隐式反馈的推荐中。由于观察数据中没有关于用户不喜欢哪个项目的信息，因此学习过程将使用未观察到的交互来表示负面偏好。曝光偏差意味着未观察到的交互不一定代表用户的负面偏好，因为用户只接触到一小部分项目。
  - **从众偏差**：从众是指用户倾向于与群体中的其他人有相似的行为，即使他们的行为违背了他们自己的判断，这使得反馈可能不能代表用户的真实偏好。
  - **位置偏差**：位置偏差在推荐中很常见，尤其是结果是由排名列表呈现的。位置偏差意味着用户倾向于与推荐列表中较高位置的项目进行交互，即使较高位置的项目可能不是高度相关的。
- **模型偏差**：模型偏差是指模型设计中的偏差。偏见并不总是有害的。事实上，模型中的偏差使模型能够实现将预测泛化

到未观察到的例子的能力。

- 归纳偏差：归纳偏差表示使模型设计者更好地学习目标并在训练数据之外进行概括的假设。

- **结果偏差：**是指推荐算法在呈现给用户的推荐结果中倾向于表现出偏差的现象。通常，推荐结果的偏差是从两个角度来研究的，一个是流行度偏差，另一个是不公平性。我们在第 7 节中介绍了公平性和相关方法，因此在本节中，我们将结果中的偏差限制为流行度偏差。
  - 人气偏差：人气偏差是指热门商品被推荐的频率高于其人气保证的现象。
- **反馈回路偏差：**是指整个 RS 反馈回路机制引入的放大偏置。数据偏差会导致数据不平衡，导致推荐结果出现偏差问题，而有偏差的推荐又会影响用户的行为，进一步放大未来推荐中的偏差。以流行偏差为例，流行的项目在观察数据中得到更多的曝光，进而获得更多的推荐机会，导致偏差放大，流行的项目变得更受欢迎，而不流行的项目变得更不受欢迎［237，238，239］。这些由反馈回路引起的放大偏差，如果无人看管，将导致回声室［115，240］或过滤气泡［241，242，243，244］，这将降低多样性并增加同质化。

一般来说，推荐系统的去偏有两种方式，一种是在训练时去偏，一种是在评估时去偏。近年来，将因果推理引入去偏推荐取得了巨大成功。在接下来的部分中，我们将介绍基于因果推理的去偏推荐模型的现有工作。

11.2 因果方法

为了解决推荐系统中的偏差问题，一种直接的解决方案是利用无偏差数据［82、96、97、98、99、100、101、102］。正如我们在部分中提到的，在宽松的可忽略性假设下，结合有限的实验数据和观测数据是一种可能的解决方案。在推荐系统中，实验数据（也称为无偏数据）通过使用随机推荐策略而不是常规推荐策略来干预系统。更具体地说，对于每个用户，他们不使用推荐模型来展示物品，而是随机选择一些物品来展示。利用无偏数据有助于实现无偏预测，因为应用随机推荐会打破反馈循环。关键的挑战是如何将一小部分无偏见的数据纳入模型设计。例如，Rosenfeld 等人［96］和 Bonner 和 Vasile［82］分别对有偏数据和无偏数据应用两个推荐模型，并通过正则化连接两个模型。Yuan 等人［97］学习了一种具有无偏数据的插补模型，用于广告点击预测。Chen 等人［101］通过元学习利用无偏数据。尽管使用无偏数据可以有效地处理偏差，但收集无偏数据会随机向用户推荐项目，而不是使用个性化推荐模型，这将不可避免地损害用户体验和平台收入。

另一种常用的方法是基于重新加权，它使用逆倾向得分为不同的偏差问题重新加权数据样本，例如选择偏差［90，106］，暴露偏差［70，72，73，245］，位置偏差［246］，247，248，249，250，251］等。关键的挑战是如何估计倾向得分以及如何将其应用于优化。一些工作［70，252］使用基于流行度的倾向估计器。一些工作［73、248、250、251］提出了一个双重问题来优化倾向估计器和推荐模型。一些工作［249］建议从观察数据中学习倾向得分。一些工作［207、235］使用双重稳健模型来处理不准确的倾向估计。逆向倾向评分方法有一些局限性，例如倾向评分不准确和存在高方差问题[206]。

因果调整是解决偏差问题的另一个有希望的方向［77、79、81、114、115］。在 do-operator 的帮助下，设计的模型旨在估计因果偏好 $P(Y|U,do(V))$ 与干预项目曝光，而不是通过传统推荐估计的纯关联偏好 $P(Y|U,V)$ 楷模。直观上，可以理解为回答一个反事实的问题：如果我们干预将项目暴露给用户，偏好是什么？因果调整用于估计观察数据的因果偏好。更具体地说，因果调整包括后门调整[68]、前门调整[68]等。基于设计的代表推荐系统中数据生成底层机制的因果图，首先要识别一组满足相应标准的变量（例如，后门调整的后门标准，前门调整的前门标准），然后对已识别的变量集应用因果调整以估计因果偏好。例如，Zhang 等人［79］应用后门调整来减轻由项目流行度引起的曝光偏差；Xu 等人［77］利用前门调整来消除未观察到的混杂因素的影响；Wang 等人［114］利用后门调整来减轻流行偏差的影响。因果调整需要识别一组满足相应标准的变量，然而，给定一个合理的推荐系统因果图，并不总是能找到一组满足该标准的变量。但设计的因果图将从其他方面指导模型设计。

因果图作为因果建模的有效且强大的工具，用于描述推荐系统中的数据生成过程。基于设计的因果图，研究人员将以此为指导来设计去偏因果模型［80，253］。例如，Zhao 等人［253］和 Zheng 等人［80］基于设计的因果图解开了偏见

和用户偏好的影响，并仅根据用户的偏好推荐项目；Wei 等人[95]和 Wang 等人[94]基于设计的因果图表示反事实世界，并进行反事实推理以进行推荐。

11.3 未决问题

逆向倾向评分(IPS)是一种很有价值的去偏方法。然而，IPS 方法的有效性高度依赖于倾向得分的正确性。如何获得正确的倾向得分仍然是一个重要但尚未解决的问题。现有工作通常基于某些项目特征设计简单的倾向估计器，例如基于流行度的倾向[70]，或从数据中学习倾向得分[73、248、249、250、251]。是否使用正确的倾向得分只能通过推荐性能的改进来间接估计。因此，倾向得分正确性的定量评价仍然是一个悬而未决的问题，需要进一步探索。

12 未解决的问题和未来的方向

12.1 潜在的因果机制

回想一下我们上面介绍的现有工作，其中大部分都是基于推荐系统的底层因果机制，这些机制由预定义的因果关系表示。通常，存在三个级别的预定义因果关系。第一级仅识别因果关系。例如，IPS 方法只研究两个变量之间的数量关系，一个是因，另一个是果。例如，在某些基于 IPS 的推荐去偏模型中，原因是项目曝光，效果是交互概率。第二个是定义因果图，识别所有变量对之间的因果关系（即是否存在因果关系，如果存在因果关系的方向）。通过预先定义因果图，现有的一些作品在因果图的指导下设计模型。例如，一些作品 [80, 253] 基于定义的因果图解开多重因果关系以实现无偏性。最后一层是结构因果模型，它不仅定义了因果关系，还定义了数量关系（即结构方程以原因为输入并返回结果值）。所提出模型的有效性与潜在因果机制的正确性高度相关。目前，大多数现有工作都是通过专家知识来定义潜在的因果机制。预定义因果机制的正确性只能通过推荐性能间接反映。因此，对定义的因果机制进行直接和定量的评估值得进一步探索。另一个观察是，即使在相同的实际场景下，不同的模型也可能具有不同的预定义因果机制。因此，我们认为应该提出一个普遍的因果机制。

12.2 因果发现

除了对预定义因果机制的准确性的担忧外，预定义因果机制的另一个局限性是专家知识预定义的因果机制通常非常简单，只考虑很少的因素。然而，在现实世界中，决策过程（即推荐系统的潜在因果机制）可能涉及更多因素，超出了领域专家的理解范围。因此，从数据中学习因果关系是推荐系统中一个重要但尚未解决的问题。很少有作品 [125, 126] 设计基于连续优化 [127, 254] 的因果发现方法用于推荐。学习到的因果机制可用于可解释的推荐或用于改进推荐。因此，提出用于推荐的因果发现方法是一个很有前途的方向。评估推荐的因果发现方法也是一个挑战。由于现实世界数据中不存在真实的因果机制，推荐中的因果发现方法通常通过推荐性能间接评估。为了直接评估推荐的因果发现方法，一种可能的解决方案是使用模拟（我们将在下一节中介绍）。

12.3 因果关系驱动的模拟

模拟是构建可以测量和分析推荐系统的环境的最强大方法之一。为推荐建立模拟将有利于工业界和学术界。例如，对于工业，仿真为从业者提供了可控的环境来分析感兴趣的目标，例如一些商业目的，以加快应用程序开发的步伐，而没有道德风险。对于学术界的研究人员来说，由于现实世界推荐系统的可访问性限制，一些提出的方法无法评估。这个问题可以通过使用模拟来解决。现有的模拟利用强化学习技术来模拟设计环境下的决策过程。然而，没有潜在因果机制的现有模拟可能会导致不准确和不稳定的决策。将因果机制用于模拟将为长期分析和因果相关分析实现更稳定的系统。例如，因果关系驱动的模拟可用于评估推荐中的因果发现方法。因此，因果关系驱动的模拟将在推荐系统中发挥重要作用，值得进一步探索。

13 结论

在本次调查中，我们对推荐的因果推理方法进行了全面审查。我们首先提供推荐系统的基础知识。然后，我们从因果推理和推荐系统的角度介绍现有工作。更具体地说，一方面，我们介绍了因果推理的知识并展示了它与推荐系统的联系，另一方面，我们介绍了推荐系统中的不同问题以及因果推理如何应用。最后，我们进一步列出了一些未解决的问题和未来的方向。我们希望这项调查能够使该领域的研究人员和从业者受益，并激发更多因果推理推荐方面的研究工作。

参考文献

[1] S. Ghemawat, H. Gobioff, and S.-T. Leung, “The google file system,” in ACM SIGOPS operating systems review, vol. 37, no. 5. ACM, 2003,pp. 29–43.

[2] J. Dean and S. Ghemawat, “Mapreduce: simplified data processing on large clusters,” Communications of the ACM, vol. 51, no. 1, pp. 107–113,2008.

[3] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, “The hadoop distributed file system,” in Mass Storage Systems and Technologies(MSST), 2010 IEEE 26th Symposium on. IEEE, 2010, pp. 1–10.

[4] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, “Spark: cluster computing with working sets,” in Proceedings of the 2nd USENIX conference on Hot topics in cloud computing, vol. 10, 2010, p.10.

[5] K. Ashton, “That internet of things thing,” RFiD Journal, vol. 22, no. 7, pp. 97–114, 2009.

[6] H. Sundmaeker, P . Guillemin, P . Friess, and S. Woelfflé, “Vision and challenges for realising the internet of things,” 2010.

[7] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of things(iot): A vision, architectural elements, and future directions,” Future Generation Computer Systems, vol. 29, no. 7, pp. 1645–1660, 2013.

[8] A. Greenberg, J. Hamilton, D. A. Maltz, and P . Patel, “The cost of a cloud: research problems in data center networks,” ACM SIGCOMM computer communication review, vol. 39, no. 1, pp. 68–73, 2008.

[9] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, “The case for vm-based cloudlets in mobile computing,” Pervasive Computing, IEEE, vol.8, no.4, pp.14–23, 2009.

[10] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog computing and its role in the internet of things,” in Proceedings of the first edition of the MCC workshop on Mobile cloud computing. ACM, 2012, pp. 13–16.

[11] M. Satyanarayanan, P . Bahl, R. Caceres, and N. Davies, “The case for vm-based cloudlets in mobile computing,” Pervasive Computing, IEEE, vol. 8, no. 4, pp. 14–23, 2009.

[12] “Openfog architecture overview,” OpenF og Consortium Architecture Working Group, 2016.

[13] L. U. Khan, I. Yaqoob, N. H. Tran, S. M. A. Kazmi, T. N. Dang, and C. S. Hong, “Edge Computing Enabled Smart Cities: A Comprehensive Survey,” IEEE Internet of Things Journal, pp. 1–1, 2020.

[14] N. Bizanis and F. A. Kuipers, “SDN and Virtualization Solutions for the Internet of Things: A Survey,” IEEE Access, vol. 4, pp. 5591–5606, 2016.

[15] K. Kimani, V. Oduol, and K. Langat, “Cyber security challenges for IoT-based smart grid networks,” International Journal of Critical Infrastructure Protection, vol. 25, pp. 36–49, Jun. 2019.

[16] P. Robertson, “Software-Defined Networking Changes the Paradigm for Mission-Critical Operational Technology Networks.”

[17] R. Durner, A. Blenk, and W. Kellerer, “Performance study of dynamic QoS management for OpenFlow-enabled SDN switches,” in Proc. IEEE 23rd Int. Symp. Qual. Service (IWQoS), Portland, OR, USA, 2015, pp. 177–182.

[18] A. Lara, A. Kolasani, and B. Ramamurthy, “Network innovation using OpenFlow: A survey,” IEEE Commun. Surveys Tuts., vol. 16, no. 1, pp. 493–512, 1st Quart., 2014.

[19] J. Bailey and S. Stuart, “Faucet: Deploying SDN in the enterprise,” Queue, vol. 14, no. 5, pp. 54–68, 2016.

[20] D. Amendola, N. Cordeschi, and E. Baccarelli, “Bandwidth management VMs live migration in wireless fog computing for 5G networks,” in Proc. 5th IEEE Int. Conf. Cloud Netw. (Cloudnet), Pisa, Italy, 2016, pp. 21–26.

[21] W.-C. Lin, C.-H. Liao, K.-T. Kuo, and C. H.-P . Wen, “Flow-and-VM migration for optimizing throughput and energy in SDN-based cloud datacenter,” in Proc. IEEE 5th Int. Conf. Cloud Comput. Technol. Sci. (CloudCom), vol. 1. Bristol, U.K., 2013, pp. 206–211.

[22] Á. L. V . Caraguay, A. B. Peral, L. I. B. López, and L. J. G. Villalba, “SDN: Evolution and opportunities in the development IoT applications,” Int. J. Distrib. Sensor Netw., vol. 10, no. 5, May 2014, Art. no. 735142.

[23] Geni. Accessed on Jan. 2017. [Online].

[24] Open Networking Foundation, SDN Definition. Accessed on Jan. 2017. [Online]. Available: https://www.opennetworking.org/sdnresources/sdn-definition

[25] S. Jain et al., “B4: Experience with a globally-deployed software defined WAN,” ACM SIGCOMM Comput. Commun. Rev., vol. 43, no. 4, pp. 3–14, 2013.

[26] S. Sezer et al., “Are we ready for SDN? Implementation challenges for software-defined networks,” IEEE Commun. Mag., vol. 51, no. 7, pp. 36–43, Jul. 2013.

[27] S. Ortiz, “Software-defined networking: On the

verge of a breakthrough?" IEEE Comput., vol. 46, no. 7, pp. 10–12, Jul. 2013.