```
消除推荐系统中流行偏见的模型无关的反事实推理
▼ ● 一、简介
  ▼ • 1. 经典训练范式
    ▼ ● 现状
        • 经典的训练范式偏向于推荐热门项目,无法揭示用户的真实偏好
    ▼ ● 数据表现
        训练数据中更受欢迎的项目被推荐的频率远高于预期,这表明存在严重的受欢迎程度偏差
    ▼ ● 模型缺陷
        • 模型倾向于简单地从项目的受欢迎程度来推荐项目,而不是用户项目匹配
    ▼ ● 导致缺陷的原因
        • 由训练范式引起,该范式认为更频繁地推荐流行项目可以实现更低的损失,从而朝着这个方向更新参数
    ▼ ● 后果
        • 1. 阻碍推荐者准确理解用户偏好, 并降低推荐的多样性
        • 2. 马太效应——流行商品被推荐得更多,甚至变得更受欢迎
  ▼ • 2. 传统解决方案
    ▼ ● 解决思路
        • 训练强调长尾项目,降低流行项目对推荐人训练的影响
    ▼ ● 解决方案
        • 1. 在训练损失中重新加权它们的交互
       • 2. 合并平衡的训练数据
        • 3. 解开用户和项目嵌入
    ▼ ● 缺陷
        • 1. 缺乏对物品流行度如何影响每一次特定互动的精细考虑
        • 2. 缺乏对流行度偏见机制的系统观点
        • 举例:流行项目上的交互总是比长尾项目权重低,而不管哪个流行项目更符合用户的偏好
  ▼ • 3. 本文观点
    ▼ ● 解决思路
        • 消除流行偏见的关键不是盲目地将推荐者推到长尾,而是了解物品流行如何影响每次互动
    ▼ ● 传统推荐模型
        • 1. 影响互动概率的主要因素有三个: 用户-项目匹配、项目受欢迎程度和用户一致性
        • 2. 现有的推荐模型主要关注用户-项目匹配因素,忽略了项目受欢迎程度如何影响交互概率
    ▼ ● 解决方案
        • 1. 构建因果图:添加了来自用户节点的直接边缘 (U) 到排名分数 (Y)
      ▼ • 2. 消除流行偏见: 推断从项目节点 (I) 到排名分数 (Y) 的直接影响
          • 从总体排名得分中扣除反事实世界中的排名得分
        ● 3. 训练方法:在主要任务和两个辅助任务上进行联合训练,两个辅助任务负责捕捉U→Y和I→Y
    ▼ ● 模型部署

    MF

    LightGCN

        • 注意: 本文方法是模型无关的 (不依赖与某些具体的模型)
  ▼ • 4. 主要贡献
      • 1. 提出推荐系统中流行度偏差的因果观点,并制定推荐的因果图
      • 2. 提出了一个模型无关的反事实推理 (MACR) 框架, 该框架根据因果图训练推荐模型, 并进行反事实推理, 以消除推荐
       推理阶段的流行偏见
      • 3. 在五个真实世界的推荐数据集上进行评估,以证明MACR的有效性和合理性
  ▼ • 5. 行文结构
      • 第1节:介绍传统实现方案的缺陷,介绍作者的实现思路和解决方法
      • 第2节: 定义模型解决的问题
      • 第3节: 介绍解决方案
      • 第4节: 阐述实验过程和实验结果, 并结合案例进行分析
      • 第5节:介绍与作者工作相关的其他工作
      • 第6节: 介绍结论和未来工作
▼ ● 二、问题定义
  ▼ • 1. 变量定义
      • U和I分别表示用户和项目的集合
      • n 是用户数量,m 是项目数
      • Y表示用户-项目交互

    Y中,每个条目 y 的计算公式 (式1)

  ▼ • 2. 训练方法
      • 从 Y 中学习评分函数 f, 预测用户 u 对于项目 i 的偏好
  ▼ • 3. 评估方法
    ▼ • 传统方法: 在一组拒绝holdout (例如, 随机或按时间划分) 交互上进行评估的
        • 可能无法反映预测用户真实偏好的能力
    ▼ • 本文方法: 遵循前人的工作[7, 30]来执行去偏评估,其中测试交互被采样为在项目上的均匀分布
        • 更多地关注用户偏好
        • 也可以检查模型处理流行度偏差的能力
▼ ● 三、解决方案
  ▼ • 1. 反事实推理的关键概念
    ▼ ● 因果图
        • 因果图 G={V, E} 是一个有向无环图

    V表示变量集

        • E 表示变量之间的因果关系
        • 大写字母 (例如 I) 表示变量
        • 小写字母 (例如 i) 表示变量的观测值
      ▼ • 边缘意味着祖先节点是原因 I, 后继结点是一种效果 Y

    I→Y: 存在 I 到 Y 的直接影响

    I→K→Y: 存在 I 通过中介变量 K 到 Y 的间接影响

      ▼ • Y的值可以根据其祖先结点的值来计算
          • Y_{i,k} 的计算公式 (式2)
    ▼ ● 因果关系
        ● 因果影响的定义: I 对 Y 的因果影响是目标变量 Y 因祖先变量 I 的单位变化而变化的幅度
       ▼ ● 总效应 TE: 计算公式 (式3)
          TE = Y_{i,Ki} - Y{i*,Ki*}
          • 理解: 两种假设情况 | = i 和 | = i* 之间的差异
          ● | = i* 值 | 的值从现实中静默的情况 (null) , Ki* 表示 | = i* 时 K 的值
          • TE 可以分解为自然直接效应 (NDE) 和总间接效应 (TIE)
      ▼ • 自然直接效应 NDE: 计算公式 (式4)
          NDE = Y_{i,Ki*} - Y_{i*,Ki*}

    含义:通过直接路径 I→Y 的影响

    理解:在 I→Y 的直接路径上, I 从 i* 到 i 时 Y 的值变化, K 设定为 I = i* 时的值

    Y_{i,Ki*}的计算是一个反事实推断,因为它要求同一变量 I的值在不同的路径上设置不同的值

      ▼ ● 总间接效应 TIE: 计算公式 (式5)
          TIE = TE - NDE = Y_{i,Ki} - Y{i,Ki*}

    含义:通过间接路径 I→K→Y 的影响

  ▼ • 2. 推荐过程的因果观点
    ▼ ● 大多数现有推荐模型的因果图 (图2a)
        ● U: 用户嵌入, I: 项目嵌入, K: 用户项目匹配特征, Y: 排名得分
        • K(U,I): 学习用户和项目之间的匹配特征, Y(K): 评分函数
        • 因果视图: Y_{K_{u,i}}
        • 缺陷: 丢弃了直接影响排名得分的用户一致性和项目受欢迎程度
    ▼ ● 更完整的推荐因果图 (图2c)
        ● U→Y: 用户对评分的直接影响, I→Y: 项目对评分的直接影响
        因果视图: Y_{u,i,K_{u,i}}
    ▼ ● 变量定义
          • 排名分数,通过一个推荐损失 (如BCE损失) 来监督恢复历史交互 (式6)
          • 指 Y_{K_{u,i}} 或 Y_{u,i,K_{u,i}}
          • 是用户 u 与物品 i 交互的可能性,不可避免地偏向数据中的热门物品
        • D: 训练集
        σ(·): sigmoid函数
        • 在测试阶段,将排名得分较高的项目推荐给用户
    ▼ ● 因果视角

    项目流行度通过 I→Y 直接影响y_{u,i},从而影响流行项目的排名得分

        阻断流行项目对Y的直接影响将消除流行度偏差
  ▼ • 3. MACR框架的引入
    ▼ ● 框架结构
        • 总体思路:通过多任务学习进行推荐器训练,通过反事实推理进行去偏推荐
        • 遵循图2 (c) 中的因果图
      ▼ ● 变量定义
          • 用户-项目匹配: y_{k} 是来自现有推荐器的排序得分,反应了项目 i 在多大程度上匹配了用户 u 的偏好

    项目模块: y_{i}表示项目受欢迎程度的影响,越受欢迎的项目得分越高

          • 用户模块: y_{u}表示无论偏好是否匹配,用户 u 在多大程度上会与物品进行交互
        • 训练目标:恢复历史交互y_{ui}},将三个分支聚合成一个最终的预测分数 (式7)
    ▼ ● 训练推荐器
        • 在整体排名分数 y_{ui} 上施加一个推荐损失
        • 设计了一种多任务学习模式,对 y_{u}和 y_{i}进行额外的监督 (式8)
     反事实推断
        思路: 消除流行度偏差的关键在于从排名分数 y_{ui} 中去除路径 I→Y 的直接影响
        • 计算公式: 式9
  ▼ • 4. MACR消除流行偏见的合理性
      • 通过反事实推理计算从 I 到 Y 的 NDE
      • 消除流行度偏差可以通过将 TE 减去 NDE 来实现 (式10)

    本文提出的反事实推断与通常推断的关键区别在于使用 TIE 对项目进行排序而非 TE (附录A)

  ▼ • 5. MACR的可能扩展
    ▼ ● 本文局限
        • 在这项工作中,我们关注的是相互作用频率所揭示的偏差
        • 作为从因果角度解决问题的初步尝试,我们忽略了其他因素的影响
    ▼ ● 可能扩展
        • 我们可以简单地扩展所提出的 MACR 框架,将这些信息作为附加节点纳入因果图中
        • 我们可以揭示导致具体建议的原因,并尝试进一步消除偏见,这将留给未来的探索
▼ • 四、实验过程与结果
  ▼ ● 提出问题
      • RQ1: MACR 是否优于现有的去偏方法?
      • RQ2: 不同的超参数设置如何影响推荐表现?
      • RQ3: 我们框架中的不同组件如何对性能做出贡献?

    RQ4: MACR 如何消除流行偏差?

  ▼ • 1. 实验设置
    ▼ ● 数据集

    Adressa, Globo, ML10M, Yelp2018, Gowalla

        • ML10M: 是一个显式反馈数据集,但我们将其转换为隐式数据,其中每个条目都标记为0或1,指示用户是否对项目进
         行了评级
        • Adresaa、Globo: 是两个流行的新闻推荐数据集
        • Gowalla、Yelp: LightGCN 中的数据集,用于公平比较
    ▼ ● 评价指标
      ▼ ● 传统策略
          • 对一组拒绝交互的传统评估策略并不能反映预测用户偏好的能力
          • 传统的评估策略不适合测试模型是否存在流行性偏差,我们需要对去偏数据进行评估
      ▼ ● 本文策略
          • 遵循前人的工作来模拟去偏推荐, 其中测试交互被采样为在项目上的均匀分布
          • 以相同概率随机抽取10%的交互项目作为测试集,另10%作为验证集,其余作为有偏差的训练数据
      ▼ ● 指标
          • 命中率 (HR) 、召回率 (Rec) 和归一化贴现累积增益 (NDCG)
    ▼ ● 基线
        ● 使用经典的 MF (MACR_MF) 和最先进的 LightGCN (MACR_LightGCN) 来实现我们的 MACR
        ● 与以下基线进行比较: MF、LightGCN、ExpoMF、CausE_MF,CausE_LightGCN、BS_MF,BS_LightGCN、Reg_
         MF, Reg_LightGCN、IPW_MF, IPW_LightGCN、DICE_MF, DICE_LightGCN
        • 由于我们的目标是对用户和项目之间的交互进行建模,因此我们不会与使用辅助信息的模型进行比较
        • 省略了与其他协同过滤模型的比较,因为 LightGCN 是目前最先进的协同过滤方法
  ▼ • 2. 实验结果 (RQ1)
      • 我们的 MACR 在所有数据集的所有指标上始终优于所有比较方法 (表2)
      • 1. 在所有情况下,我们的 MACR 都会大幅提升 MF 或 LightGCN
      • 2. 在大多数情况下,LightGCN 的性能比 MF 差,但在常规数据集拆分中,LightGCN 通常是一种性能更好的方法
      • 3. 在数据集方面,我们还可以发现,与Globo数据集相比,改进幅度非常大

    4. 关于流行度去偏的基线, Reg 方法在基本模型上的改进有限, 有时甚至表现得更糟

      • 我们还报告了我们在 Adressa 数据集上的实验结果 (附录C.1) , 为了更全面的评估, 我们使用了不同的 K 值
  ▼ • 3. 案例研究
    ▼ • 1. 超参数的影响 (RQ2)

    三个超参数: α、β、c

      ▼ ● 敏感性分析

    α、β: 附录C.2

→ C
             • 以 MACR_LightGCN 为例,随着 c 从 0 到 29 的变化,模型的性能越来越好
             • 而进一步增加 c 会适得其反
             • 这说明适当程度的阻塞中间匹配偏好有利于流行度去偏并提高推荐性能
             ● 与 MACR_MF 相比,MACR_LightGCN 对 c 更敏感,因为它的性能在优化后下降得更快
             • 这表明 LightGCN 更容易受到流行偏见的影响
    ▼ • 2. 用户分支和项目分支的影响 (RQ3)
        • 总体思路:为了考察用户分支和项目分支的整体效应,我们在Adressa数据集上对MACR_MF进行了消融研究,并一次
         去除不同的成分进行比较
      ▼ ● 具体方法:将MACR与其4种特殊情况进行比较

 MACR_MF w/o 用户 (项目) 分支

    MACR_MF w/o LI (LU)

      ▼ ● 结果
          • 用户分支和项目分支都提高了推荐性能,与删除用户分支相比,该模型在删除项目分支时的性能要差得多
         ▼ ● 与移除 LU 相比,移除 LI 也会对性能造成更严重的损害
             • 验证了项目流行度偏见对推荐的影响大于用户一致性
         ▼ ● 与简单地移除 LI 和 LU 相比,删除用户/项目分支会使模型的性能变差
             • 验证了在预测中进一步融合项目和用户影响的重要性
    ▼ • 3. 去偏能力 (RQ4)
```

• 实验方法: 我们将 MACR_MF 和 MACR_LightGCN 与它们的基本模型 MF 和 LightGCN 进行了比较

• 实验结果:我们的方法确实降低了热门项目的推荐频率,并推荐了更多不太受欢迎的项目

• 最受欢迎的项目组的召回率增长最大,受欢迎项目的推荐频率降低了

• 结论: 我们的模型能够捕捉商品的流行度和用户的一致性, 从而使去偏受益

• 探索目的: 为了研究为什么我们的框架有利于推荐中的去偏

• 我们回顾了与这项工作最相关的关于推荐中的流行偏见和推荐中的因果推理的现有工作

• 探索内容: 什么用户分支和项目分支实际在模型中学习

• 不受欢迎的项目组有相对较小的改进

▼ ● 第一条研究线是基于上一节中描述的反向倾向加权 (IPW)

• 这些先前的方法忽略了流行度如何影响每一个特定的互动

• 这些方法导致在推荐准确性和不受欢迎项目的覆盖范围之间进行权衡

• 创建一个去偏训练数据集作为辅助任务,以帮助在偏斜数据集中训练的模型更好地泛化

▼ ● ExpoMF 联合建模了用户对某个项目的接触,以及他们由此产生的点击决策

• 一个由两个模型组成的框架: 一个暴露模型和一个偏好模型

• 核心思想是重新加权训练损失中的相互作用

• 权重被设置为物品受欢迎程度的倒数

▼ ● 另一条研究路线试图通过排名调整来解决这个问题

• 2. 可应用于推荐系统输出的重新排序方法

• 1. 基于正则化的方法

• 用于缓解流行度偏差

• 也可以用来缓解流行度偏差

• 缓解受欢迎程度偏差

▼ ● 我们提出了模型无关框架 MACR

• 2. 探索如何结合各种辅助信息

• 3. 如何扩展我们的框架以缓解推荐系统中的其他偏见

• 5. 将探索因果关系和其他关系领域知识的结合

• 我们认为他们的方法并没有明确消除流行偏见

▼ ● Zheng 等人也试图通过因果方法来缓解流行偏见

• 缺乏对流行偏见机制的系统观点

• 是基于概率的,因此不能推广到更普遍的环境中

• 我们提出了第一个因果观点来缓解推荐系统中的流行偏见问题

• 根据因果图进行多任务训练,以评估不同原因对排名得分的贡献

在五个真实世界的推荐数据集上进行的大量实验已经证明了 MACR 的有效性

• 1. 把因果观扩展到推荐系统中的更多应用,并探索用户和项目模块的其他设计

• 4. 研究通过反事实推理同时消除流行偏见和暴露偏见等多种类型的偏见

• 进行反事实推理以估计项目属性对排名分数的直接影响,并将其去除以消除流行度偏差

▼ • 2. 以因果关系为导向的数据

▼ ● 实验发现

▼ ● 探索

▼ • 1. 推荐中的流行偏差

▼ • 2. 推荐中的因果推理

▼ • 3. 因果嵌入

• 六、结论与未来工作

▼ • 2. 未来工作

▼ • 1. 结论

▼ • 1. 反向倾向权重

▼ ● 五、相关工作