带偏差连续反馈的无偏学习排序

Yi Ren*腾讯 中国北京
h
enrybjren
@tencent.com

Hongyan Tang*腾讯
中国北京
violatang
@tencent.com

Siwen Zhu 腾讯 中国
北京
si
wenzhu@tencent.com

arXiv:2303.04335v1 [cs.IR] 8 Mar 2023

摘要

如何基于有偏反馈学习无偏排序器是一个众所周知的挑战。无偏学习排序（LTR）算法已被证实能够基于噪声反馈准确地建模相对相关性，因此颇具吸引力，并已应用于许多采用单一类别标签的场景，例如用户点击信号。然而，现有的无偏LTR方法无法妥善处理连续反馈，而这对许多工业应用至关重要，例如内容推荐系统。

为了提供个性化的高质量推荐结果，推荐系统需要对类别型和连续型有偏反馈（如点击和停留时间）进行建模。由于无偏学习排序方法无法处理这类连续反馈，而且在有偏反馈下，未经去偏的成对学习通常不如逐点学习，这一点也在我们的实验中得到了验证，因此，在主要的工业推荐应用中，主流做法是训练多个逐点排序器来预测多个目标的绝对值，并利用一个独立的浅层塔来估计和减轻位置偏差的影响。然而，这种训练范式的优化目标与排序指标大相径庭：排序指标关注的是排名靠前的项目之间的相对顺序，而非每个物品的预测精度。此外，由于现有系统倾向于在较高位置推荐相关性更高的物品，基于浅层塔的方法难以准确区分用户反馈究竟是由位置还是相关性所致。因此，如果能够解决上述问题，我们就有望进一步提升性能。

因此，我们设计了一种新颖的无偏学习排序算法 [1] 来应对这些挑战。该算法创新性地以成对方式建模位置偏差，并引入成对信任偏差，以明确区分位置偏差、信任偏差和用户相关性，同时适用于连续反馈和类别反馈。在公开基准数据集以及腾讯新闻大规模推荐系统的内部实时流量上进行的实验结果表明，所提出的方法在连续标签任务上表现出色，在类别标签任务上也具有竞争力。

*两位作者对本研究的贡献相同。[1]源代码已发布于 https://github.com/phyllist/ULTRA。

特此授予许可，可免费为个人或课堂使用而制作本作品的全部或部分数字或纸质副本，但前提是不得为营利或商业利益而制作或分发副本，且副本首页必须附有本声明及完整引文。对于本作品中由非 ACM 所有的组件，必须尊重其版权。注明出处后允许摘编。若要进行其他复制、重新发表、在服务器上发布或向列表重新分发，必须事先获得具体许可和/或支付费用。请通过 permissions@acm.org 申请许可。
CIKM '22，*2022*年*10*月*17*日至*21*日，美国佐治亚州亚特兰大。© 2022年美国计算机协会。ACM ISBN 978-1-4503-9236-5/22/10...$15.00
https://doi.org/10.1145/3511808.3557483

CCS 概念

• **信息系统** →**学习排序**。

关键词

无偏学习排序、学习排序、位置偏差、连续反馈、推荐系统、成对学习

ACM 引用格式：
Yi Ren、Hongyan Tang 和 Siwen Zhu。2022年。《利用有偏连续反馈进行无偏学习排序》。载于第*31*届 *ACM* 国际信息与知识管理会议论文集（*CIKM '22*），*2022*年*10*月*17*日至*21*日，美国佐治亚州亚特兰大。ACM，纽约，纽约州，美国，10页。https://doi.org/10.1145/3511808.3557483

1 引言

目前，无偏学习排序算法已广泛应用于搜索引擎等存在点击信号偏差的场景中。 [1, 2, 10] 这些算法将位置偏差建模为一种反事实效应，并通过使用各自的逆倾向权重对每个含噪点击信号进行加权，估计无偏相关性偏好，从而为相应应用带来显著的性能提升。然而，现有方法只能处理类别标签，无法处理连续标签，因为相关性与连续用户反馈之间的关系不符合其核心假设：如公式 3 所示，二元点击概率等于二元相关性概率与检查概率的乘积。此外，连续反馈（例如停留时间和视频观看比例）对于在内容推荐系统等许多其他场景中准确建模用户偏好至关重要，因此，设计一种适用于连续标签的无偏学习排序方法十分必要。

推荐系统需要预测类别型和连续型用户反馈（例如点击、购买、分享和停留时间），以学习用户偏好并向相应用户推荐排名靠前的物品。在推荐场景中，用户行为往往存在偏差。例如，用户点击并观看视频，可能仅仅是因为该视频排名靠前，而非其相关性高。因此，为了准确估计用户的相对偏好并提升推荐系统的排序性能，必须纠正有偏用户行为所带来的位置偏差 [7] 。由于现有无偏LTR方法无法妥善处理连续反馈，而未经去偏的成对学习在有偏反馈上的表现往往不如逐点学习（我们在第 5 节的实验也验证了这一点），主流工业排序模块通常会训练多个逐点排序器，以学习多种用户行为的绝对概率 [25, 31] ，并添加一个浅层塔，直接建模并消除位置偏差 [8, 31]。然而，训练目标与

实际推荐场景中的排序目标之间存在差距：排序关注的是物品的相对顺序，而非绝对值。此外，基于浅层塔的模型将偏差特征与相关性特征耦合在一起，使得有效消除偏差颇为困难 [27]。因此，通过这种学习范式，很难纠正位置偏差并有效学习用户偏好。

为实现有效的偏差校正并融入成对学习范式，我们提出了一种新颖的无偏成对学习排序方法。该方法以成对方式对基于位置的检查偏差建模，并引入成对信任偏差，以更好地刻画潜在偏差并更准确地学习相对偏好。与最先进的无偏学习排序方法相比，我们的方法直接对无偏相关性对与有偏用户反馈对之间的相关性建模，不依赖上述关于标签的假设，且适用于类别标签和连续标签。此外，成对去偏与排序目标更加一致，能够取得更好的偏好学习效果。进一步地，引入成对信任偏差后，可明确区分依赖位置的检查偏差和依赖位置对的相对信任偏差，从而实现更好的校正。在学习排序和推荐基准数据集上的离线实验，以及腾讯新闻大规模视频推荐系统中的在线 A/B 测试表明，我们的方法不仅在连续标签场景下优于最先进的排序模型，在类别标签场景下也具有竞争力。

本文的主要贡献如下：

- 我们提出了一种新颖的无偏成对学习排序方法，以更好地校正偏差并实现无偏的相对偏好学习。据我们所知，所提出的方法是首个既适用于类别标签也适用于连续标签的无偏学习排序方法。
- 我们将基于回归的EM算法 [27] 扩展到成对设置，以有效估计所提出方法的参数。
- 我们通过离线和在线实验评估了我们的方法的性能。基准数据集上的结果表明，我们的方法在连续标签和类别标签上均取得了有竞争力的表现。腾讯新闻大规模推荐系统中的在线 A/B 测试显示，业务指标显著提升了 2.08%。

2 相关工作

本节将讨论无偏学习排序以及推荐系统中的去偏方法相关工作。

2.1 无偏排序算法学习

近年来，无偏学习排序已被积极研究，作为一种利用有偏反馈进行学习以校正位置偏差的有前景方法 [7] 。Joachims [15] 首先提出了反事实框架，通过逆倾向加权（IPW）估计器学习理论上无偏的排序器。为了估计倾向得分，Wang [26] 和 Joachims [15] 都提出了结果随机化和干预方法。为减少对用户体验的负面影响，一些研究提出直接从有偏反馈中学习倾向得分，例如 DLA [2], 、回归 EM [27],以及无偏 LambdaMART [10]。这些方法忽略了位置相关的点击噪声 [14] ，而 [1]对此进行了处理。不过，由于其核心假设是方程 3 中所示的二元点击概率和二元相关性概率，以上所有方法都只能用于类别标签。相比之下，根据本文的实验结果，我们提出的方法既适用于类别标签，也适用于连续标签，并且在两者上都取得了有竞争力的性能。

2.2 从非随机缺失的隐式反馈中学习无偏推荐器

另一条相关但有所不同的研究路线 [9, 18, 19, 23, 24]侧重于基于非随机缺失（MNAR）的隐式反馈学习无偏推荐器。这些研究主要关注消除由 MNAR 反馈 [7] 导致的曝光偏差 [7] 和选择偏差，并针对基于协同过滤的推荐方法[16, 17, 20]展开研究。协同过滤适用于推荐系统中的候选生成模块，可在低延迟下 [11], 从所有候选项中检索一组相关物品；它将用户交互过的物品视为正样本，将其他所有物品视为潜在负样本，但往往忽略困难负样本。相比之下，我们的工作针对排序模块，并将已曝光但未交互的困难负样本项作为负样本，以便将用户最想要的物品排在前面。此外，我们的工作侧重于纠正位置偏差 [7] ，而非选择偏差和曝光偏差。

2.3 推荐系统排序模块的位置偏差建模

如上所述，排序模块会将用户浏览过但未交互的物品作为负样本，以实现准确推荐。传统排序模块通常采用逐点学习方法来估计交互概率 [25, 31] ，因为无偏学习排序方法无法处理有偏连续反馈。为了校正逐点排序模型中的位置偏差 [7] ，Zhao [31]和 Guo [8]都提出采用浅层塔，通过将浅层塔的输出与主模型的输出相加或相乘，来估计并消除偏差。然而，由于偏差特征与相关性特征相互耦合，基于浅层塔的方法无法有效发挥作用 [27]。Wu [29]将 [27]的思路应用于推荐，采用逆倾向加权框架和回归期望最大化来学习无偏排序器。然而，这种方法仍然只能用于类别标签，不适用于典型的推荐系统。我们的方法将逆倾向加权框架扩展到成对设置，以处理连续标签，在去偏和偏好学习方面取得了更优的性能。

3 方法论

本节将介绍针对类别标签和连续标签的无偏学习排序通用框架。

3.1 类别标签的无偏学习排序

假设有 N 个位置，对于用户请求 u，物品 x_i 显示在位置 $i \in [1, N]$，r_i 是用户-物品对（u，x_i）的无偏相关性。为简单起见，我们只考虑二元相关性，

并且可以轻松将其扩展到多级情形。然后我们可以按如下方式计算风险函数:

$$R_{rel}(f) = \int L(f(u, x_i), r_i) dP(r_i = 1, u, x_i) \tag{1}$$

其中，$f(u,\ x_i)$表示用户-物品对（$u,\ x_i$）的排序分数，$L(f(u,\ x_i),\ r_i)$表示基于性能指标的损失函数，$P(r_i{=}1,\ u,\ x_i)$表示物品 x_i上正r_i的概率分布。为简单起见，损失函数中省略了物品的位置信息。由于折损累计增益（DCG） [12, 13] 和平均相关位置（ARP） [28] 等大多数性能指标只考虑相关物品，因此这里的风险函数仅基于具有正相关性标签的物品计算。

$$\hat{f}_{rel} = \arg\min_{f} \sum_{u, x_i, r_i > 0} L(f(u, x_i), r_i) \tag{2}$$

如果能够获得无偏相关性标签，我们就可以通过最小化公式 2 中所示的经验风险函数来学习排序器。然而，在大规模应用中，收集足够多的真实相关性标签并不可行，因为我们无法为所有用户找到足够多合适的评审人员。因此，如果能够弥合用户行为与相关性标签之间的差距，现实应用中大量有偏的用户行为（例如点击、购买、点赞）就能成为宝贵的标签。基于位置的模型（PBM） [22], 假设，用户当且仅当检查到相关物品时才会点击该物品，并且检查概率仅取决于位置，与相关性无关。如表 1 所示，令 e_i 表示用户是否检查了物品 x_i，c_i 表示物品 x_i的用户行为标签，我们可以对 c_i和 r_i之间的关系建模如下:

$$\begin{aligned} P(c_i = 1|u, x_i, i) &= P(r_i = 1|u, x_i, i) P(e_i = 1|u, x_i, i) \\ &= P(r_i = 1|u, x_i) P(e_i = 1|i) \end{aligned} \tag{3}$$

然后，我们可以基于用户行为标签推导风险函数和经验风险函数:

$$\begin{aligned} R_{unbiased}(f) &= \int \frac{L(f(u, x_i), c_i)}{P(e_i = 1|i)} dP(c_i = 1, u, x_i, i) \\ &= \int L(f(u, x_i), c_i) d \frac{P(c_i = 1, u, x_i, i)}{P(e_i = 1|i)} \\ &= \int L(f(u, x_i), c_i) dP(r_i = 1, u, x_i) \\ &= \int L(f(u, x_i), r_i) dP(r_i = 1, u, x_i) \\ &= R_{rel}(f) \end{aligned} \tag{4}$$

$$\hat{f}_{rel} = \hat{f}_{unbiased} = \arg\min_{f} \sum_{u, x_i, i, c_i > 0} \frac{L(f(u, x_i), c_i)}{P(e_i = 1|i)} \tag{5}$$

如公式 4 所示，采用逆倾向加权（IPW）损失时， [15] 实际上等于 R_{rel}。换句话说，如果我们能够准确估计检查概率 $P(e_i = 1|i)$，就可以基于用户行为标签学习无偏排序器。

由于公式 3 中对相关性标签与用户行为标签的概率分布所作的假设，上述方法只能应用于标签为类别型而非连续型的任务，而连续型标签中，用户行为标签与真实相关性之间的关系往往复杂得多。然而，停留时间等连续型标签以及视频观看比例是反映用户满意度的重要反馈，在工业推荐系统中也很常用。为消除连续标签中的位置偏差，我们提出了一种新颖的无偏学习排序方法，该方法绕过了此前关于无偏相关性标签与有偏用户行为标签之间关系的假设。该方法的具体细节将在后续章节中介绍。

符号说明	描述
u	包含用户档案和上下文的用户请求。
x_i	显示在位置 i的物品特征。
r_i	用户-物品对（$u,\ x_i$）的无偏相关性。
c_i	用户-物品对（$u,\ x_i$）的有偏反馈标签。
e_i	x_i的检查。
θ_i	$P(e_i = 1|i)$，检查概率。
θ_i^-	$P(e_i = 1|i,\ c_i = 0)$
ϵ_{ij}^+	$P\ (c_i > c_j|e_i = 1,\ e_j = 1,\ r_i > r_j,\ i,\ j)$
ϵ_{ij}^-	$P(c_i > c_j|e_i = 1, e_j = 1, r_i <= r_j, i, j)$
$\gamma_{u,x_b x_j}$	$P(r_i > r_j|u, x_i, x_j)$
β_{u,x_i}	$P(r_i > 0|u, x_i)$，（u, x_i）的相关概率。

表 1：符号说明及描述

3.2 连续标签的无偏学习排序

对于连续用户反馈，前述无偏学习排序方法无法奏效。为克服无偏相关性标签与存在偏差的用户行为标签之间强假设的局限，我们从成对学习的角度提出了一种新颖的无偏学习排序方法，该方法适用于类别标签和连续标签。

在成对设置中，损失函数定义在物品对上。设 $\hat{y}_i = f(u,\ x_i)$ 表示用户物品对 $(u,\ x_i)$ 的预测分数，$L(\hat{y}_i,\ r_i,\ \hat{y}_j,\ r_j)$ 表示成对损失函数，$P(r_i{>}r_j,\ u,\ x_i,\ x_j)$ 表示用户 u偏好物品 x_i而非物品 x_j的概率分布，则风险函数以及通过最小化经验风险函数学得的排序器分别定义在公式 6 和公式 7 中。与针对类别标签的无偏学习排序类似，我们也只考虑正向相关性对，因为负向对和中性对不应参与优化 DCG 和 ARP 等常见排序指标。

$$R_{rel}(f) = \int L(\hat{y}_i, r_i, \hat{y}_j, r_j) dP(r_i > r_j, u, x_i, x_j) \tag{6}$$

$$\hat{f}_{rel} = \arg\min_{f} \sum_{u, x_i, x_j, r_i > r_j} L(\hat{y}_i, r_i, \hat{y}_j, r_j) \tag{7}$$

3.2.1 基于成对位置偏置模型的无偏学习排序。为了对无偏相关性对（$r_i,\ r_j$）与有偏连续标签对（$c_i,\ c_j$）之间的关系进行建模，我们将 PBM 扩展到成对设置。具体而言，我们假设对物品 x_i和 x_j的检查是相互独立的；如果 x_i和x_j都经过检查，那么当且仅当 x_i比x_j更相关时，x_i才会呈现出比x_j更大的连续标签，即:

$$P(c_i > c_j, e_i = 1, e_j = 1|u, x_i, x_j, i, j) = \theta_i \theta_j \gamma_{u,x_i,x_j} \tag{8}$$

，其中 $\gamma_{u,\ x_i,\ x_j} = P\ (r_i > r_j|u,\ x_i,\ x_j)$。然后，我们可以基于正连续标签对推导出风险函数：

$$
\begin{aligned}
&R_{unbiased}(f)\\
&=\int\frac{L(\hat{y}_i,c_i,\hat{y}_j,c_j)}{\theta_i\theta_j}P(e_j=1|u,i,j,x_i,x_j,c_i>c_j)dP(c_i{>}c_j,u,x_i,x_j,i,j)\\
&=\int\frac{L(\hat{y}_i,c_i,\hat{y}_i,c_j)}{\theta_i\theta_j}dP(c_i{>}c_j,u,x_i,x_j,i,j,e_i=1,e_j=1)\\
&=\int\frac{L(\hat{y}_i,c_i,\hat{y}_i,c_j)}{\theta_i\theta_j}dP(r_i{>}r_j,u,x_i,x_j,i,j,e_i=1,e_j=1)\\
&=\int\frac{L(\hat{y}_i,r_i,\hat{y}_j,r_j)}{\theta_i\theta_j}dP(r_i{>}r_j,u,x_i,x_j,i,j)P(e_i=1|i)P(e_j=1|j)\\
&=\int L(\hat{y}_i,r_i,\hat{y}_i,r_j)dP(r_i{>}r_j,u,x_i,x_j)=R_{rel}(f)
\end{aligned}
\tag{9}
$$

基于 $R_{unbiased}$ 是 R_{rel}的无偏估计量这一依据，我们可以通过最小化公式 10 中所示的经验风险函数 $loss_{IPW}$来学习无偏排序器，其中 $L_{ij} = L(\hat{y}_i,\ c_i,\ \hat{y}_j,\ c_j)$。值得注意的是，为了提高经验风险函数的精度，正标签样本和零标签样本分别进行了明确处理，因为不同标签样本的后验检查概率有所不同。具体而言，正标签物品的后验检查概率必定为 1，而零标签物品的后验检查概率则不确定。

$$
loss_{IPW} = \sum_{u,x_i,x_j,i,j,c_i>c_j,c_j>0}\frac{L_{ij}}{\theta_i\theta_j} + \sum_{u,x_i,x_j,i,j,c_i>c_j,c_j=0}\frac{L_{ij}h_{ij}}{\theta_i\theta_j}, \tag{10}
$$

其中 $h_{ij} = P(e_j = 1|u,\ x_i,\ x_j,\ i,\ j,\ c_i > c_j,\ c_j = 0)$。

3.2.2 基于成对信任偏差的无偏学习排序。基于成对位置偏置模型的无偏学习排序方法假设，如果 x_i 的相关性高于x_j，且两者都经过检查，那么 x_i 的连续标签会大于x_j。然而，根据 [14],中的用户研究，用户往往更信任排名靠前的物品，因此，即使不同位置上的物品都经过检查且相关性相同，排名靠前的物品仍会有更高的点击率，这种现象称为信任偏差。此外，连续反馈值通常遵循某种分布，而不是由相关性标签到连续反馈值的特定映射，因此有偏标签对与无偏相关性对之间的关系会更加复杂。为了更准确地估计无偏相关性对（$r_i,\ r_j$）与有偏连续标签对（$c_i,\ c_j$）之间的关系，本文纳入上述信任偏差，并将其扩展到成对设置。由于信任偏差仅取决于曝光位置，我们在公式 11 和公式 12 中对成对信任偏差进行建模，并假设 P（$c_i > c_j|e_i = 1,\ e_j = 1,\ r_i > r_j,\ u,\ x_i,\ x_j,\ i,\ j$）和 P（$c_i > c_j|e_i = 1,\ e_j = 1,\ r_i \le r_j,\ u,\ x_i,\ x_j,\ i,\ j$）与（$u,\ x_i,\ x_j$）相互独立。具体而言，对于正相关性对 $r_i > r_j$，曝光为正连续标签对 $c_i > c_j$的概率并非 1，而是 ϵ^+_{ij}；而非正相关性对也有ϵ^-_{ij}的概率表现为正连续标签对。 ϵ^+_{ij}和ϵ_{ij}仅取决于位置，并满足约束 $0 < \epsilon^-_{ij} < \epsilon^+_{ij} < 1$。

$$
P(c_i > c_j|e_i = 1, e_j = 1, r_i > r_j, i, j) = \epsilon^+_{ij} \tag{11}
$$

$$
P(c_i > c_j|e_i = 1, e_j = 1, r_i \le r_j, i, j) = \epsilon^-_{ij} \tag{12}
$$

接下来可以计算正连续标签对的概率。如公式 13 所示，曝光为正连续标签对 $c_i{>}c_j$有两种情况。第一，如果x_j经过检查，那么 c_i与c_j之间的关系取决于 r_i与r_j之间的关系以及信任偏差。第二，如果x_j未经过检查，正标签对$c_i > c_j$会退化为正数据点 $c_i > 0$；根据 PBM，这只取决于 x_i是否经过检查及其相关性。

$$
\begin{aligned}
&P(c_i > c_j|u, x_i, x_j, i, j)\\
&= P(c_i > c_j, e_i = 1, e_j = 1|u, x_i, x_j, i, j)\\
&+ P(c_i > 0, e_i = 1, e_j = 0|u, x_i, x_j, i, j)\\
&= \theta_i\theta_j(\epsilon^+_{ij}\gamma_{u,x_i,x_j} + \epsilon^-_{ij}(1 - \gamma_{u,x_i,x_j})) + \theta_i(1 - \theta_j)\beta_{u,x_i}
\end{aligned}
\tag{13}
$$

，其中 $\beta_{u,\ x_i} = P(r_i > 0|u,\ x_i)$ 是 $(u,\ x_i)$ 的相关概率。

由于我们无法再根据 $c_i > c_j$的 $e_i = 1$、 $e_j = 1$ 的状态推断 $r_i > r_j$，引入成对信任偏差后，无法从 $R_{unbiased}$ 推导出 R_{rel}。为解决这一问题，我们计算在给定正连续标签对和真实检查的情况下，相关性对为正的概率，即：

$$
\begin{aligned}
&P(r_i > r_j|u, x_i, x_j, i, j, c_i > c_j, e_i = 1, e_j = 1)\\
&= \frac{P(c_i{>}c_j|e_i{=}1,e_j{=}1,r_i{>}r_j,u,x_i,x_j,i,j)P(r_i{>}r_j|u,x_i,x_j)}{P(c_i > c_j,e_i = 1,e_j = 1|u, x_i, x_j, i, j)}\\
&= \frac{\epsilon^+_{ij}\gamma_{u,x_i,x_j}}{\epsilon^+_{ij}\gamma_{u,x_i,x_j} + \epsilon^-_{ij}(1 - \gamma_{u,x_i,x_j})} = m_{ij}
\end{aligned}
\tag{14}
$$

最后，我们基于 $loss_{IPW}$ 和 m_{ij} 定义了贝叶斯逆倾向加权损失，以补偿并校正基于位置的检查偏差和信任偏差。由公式15和公式14可知，在成对位置偏置模型无噪声的情况下，即当 $\epsilon^+_{ij} = 1$ 和 $\epsilon^-_{ij} = 0$ 时，贝叶斯逆倾向加权损失会退化为 $loss_{IPW}$。

$$
loss_{Bayes-IPW} = \arg\min_f \sum_{\substack{u,x_i,x_j,i,j,\\ c_i>c_j,c_j>0}}\frac{L_{ij}m_{ij}}{\theta_i\theta_j} + \sum_{\substack{u,x_i,x_j,i,j,\\ c_i>c_j,c_j=0}}\frac{L_{ij}m_{ij}h_{ij}}{\theta_i\theta_j}, \tag{15}
$$

其中， h_{ij} 的计算方式如下：

$$
\begin{aligned}
&h_{ij} = P(e_j = 1|u, x_i, x_j, i, j, c_i > c_j, c_j = 0)\\
&= \frac{P(e_j{=}1|u,x_j,j,c_j{=}0)P(c_i{>}c_j|u,x_i,x_j,i,j,e_j{=}1,c_j{=}0)}{P(c_i > c_j|u, x_i, x_j, i, j, c_j = 0)}\\
&= \frac{\theta_i\theta^-_j(\epsilon^+_{ij}\gamma_{u,x_i,x_j}{+}\epsilon^-_{ij}(1{-}\gamma_{u,x_i,x_j}))}{\theta_i\theta^-_j(\epsilon^+_{ij}\gamma_{u,x_i,x_j}{+}\epsilon^-_{ij}(1{-}\gamma_{u,x_i,x_j}))+\theta_i(1{-}\theta^-_j)\beta_{u,x_i}},
\end{aligned}
\tag{16}
$$

其中， θ^-_j 表示已知物品 x_j 的连续标签为 0 时的后验检查概率。

3.2.3 排序指标的直接优化。由于成对损失 [4]着重于最小化成对错误的数量，而忽略了相关性差异不同的物品对的相对重要性，因此与折损累计增益和 ARP 等常见性能指标并不完全匹配。为了在模型中直接优化排序指标，我们参照 LambdaRank [3]的做法改进了损失函数。LambdaRank 的主要思想是引入 NDCG 变化量，以直接优化 NDCG 评估指标；其中，NDCG 变化量表示在排序列表 [3, 5]中交换物品 x_i和x_j 后 NDCG 分数的差异。由于无法获得无偏的真实相关性，我们假设有偏反馈

$$P(e_i=1,e_j=1,r_i>r_j|u,x_i,x_j,i,j,c_i>c_j)=\frac{\theta_i\theta_j\epsilon_{ij}^{+}\gamma_{u,x_i,x_j}}{\theta_i\theta_j(\epsilon_{ij}^{+}\gamma_{u,x_i,x_j}+\epsilon_{ij}^{-}(1-\gamma_{u,x_i,x_j}))+\theta_i(1-\theta_j)\beta_{u,x_i}}$$

$$P(e_i=1,e_j=1,r_i\le r_j|u,x_i,x_j,i,j,c_i>c_j)=\frac{\theta_i\theta_j\epsilon_{ij}^{-}(1-\gamma_{u,x_i,x_j})}{\theta_i\theta_j(\epsilon_{ij}^{+}\gamma_{u,x_i,x_j}+\epsilon_{ij}^{-}(1-\gamma_{u,x_i,x_j}))+\theta_i(1-\theta_j)\beta_{u,x_i}}$$

$$P(e_i=1,e_j=0,r_i>0|u,x_i,x_j,i,j,c_i>c_j)=\frac{\theta_i(1-\theta_j)\beta_{u,x_i}}{\theta_i\theta_j(\epsilon_{ij}^{+}\gamma_{u,x_i,x_j}+\epsilon_{ij}^{-}(1-\gamma_{u,x_i,x_j}))+\theta_i(1-\theta_j)\beta_{u,x_i}}$$

$$P(e_i=1,e_j=1,r_i\le r_j|u,x_i,x_j,i,j,c_i\le c_j)=\frac{\theta_i\theta_j(1-\epsilon_{ij}^{-})(1-\gamma_{u,x_i,x_j})}{1-(\theta_i\theta_j(\epsilon_{ij}^{+}\gamma_{u,x_i,x_j}+\epsilon_{ij}^{-}(1-\gamma_{u,x_i,x_j}))+\theta_i(1-\theta_j)\beta_{u,x_i})}$$

$$P(e_i=1,e_j=1,r_i>r_j|u,x_i,x_j,i,j,c_i\le c_j)=\frac{\theta_i\theta_j(1-\epsilon_{ij}^{+})\gamma_{u,x_i,x_j}}{1-(\theta_i\theta_j(\epsilon_{ij}^{+}\gamma_{u,x_i,x_j}+\epsilon_{ij}^{-}(1-\gamma_{u,x_i,x_j}))+\theta_i(1-\theta_j)\beta_{u,x_i})}$$

$$P(e_i=1,e_j=0,r_i\le 0|u,x_i,x_j,i,j,c_i>c_j)=0 \quad P(e_i=0|u,x_i,x_j,i,j,c_i>c_j)=0$$

$$P(e_i=0,r_i>0|u,x_i,i,c_i=0)=\tfrac{(1-\theta_i)\beta_{u,x_i}}{1-\theta_i\beta_{u,x_i}} \qquad P(e_i=1,r_i>0|u,x_i,i,c_i=0)=0$$

$$P(e_i=1,r_i\le 0|u,x_i,i,c_i=0)=\tfrac{\theta_i(1-\beta_{u,x_i})}{1-\theta_i\beta_{u,x_i}} \qquad P(e_i=1,r_i>0|u,x_i,i,c_i>0)=1$$

图1：期望步骤中的公式

与无偏相关性正相关，并将成对损失 $loss_{Bayes-IPW}$改进 $loss_{opt}$如下：

$$loss_{opt}=\sum_{\substack{u,x_i,x_j,i,j,\\ c_i>c_j,c_j>0}}\frac{L_{ij}m_{ij}|\Delta Z_{ij}|}{\theta_i\theta_j}+\sum_{\substack{u,x_i,x_j,i,j,\\ c_i>c_j,c_j=0}}\frac{L_{ij}m_{ij}h_{ij}|\Delta Z_{ij}|}{\theta_i\theta_j}, \tag{17}$$

其中， ΔZ_{ij}表示当排序列表中的物品 x_i和x_j互换时，基于有偏标签计算得到的评估指标之差。

因此，如果我们能够估计 θ_i、 θ_i^{-}、 ϵ_{ij}^{+}和ϵ_{ij}^{-}的参数，就可以通过最小化损失函数来学习无偏排序器 $f_{unbiased}$。本文采用基于回归的期望最大化（EM）方法 [27] 来估计这些参数。估计过程将在下一节中介绍。

3.3 多目标应用的监督标签合成

有些应用（例如内容推荐系统）通常会同时考虑多个实时指标 [25]。例如，在视频推荐场景中，我们希望最大化用户的视频观看次数、停留时间以及其他满意度相关指标（例如点赞和分享）。为了对多种行为进行建模并同时最大化多个实时指标，传统推荐系统通常会训练多个排序器来预测多个目标的绝对值，并结合这些预测结果进行最终排序。然而，这种训练范式的优化目标与排序指标大相径庭：排序指标关注的是排名靠前的项目之间的相对顺序，而非绝对分数。此外，如果我们通过不同的无偏学习排序模型训练多个排序器，就无法以理论上合理的方式组合预测结果，因为不同无偏学习排序模型的预测值可能并不处于同一尺度。

为了解决这些问题并获得更好的多目标优化性能，我们提出基于多种用户行为定义合成标签，从而在单个模型中直接对多个实时指标进行建模。如公式 18 所示，合成标签通过组合函数 {v1} 合并多种用户行为（例如点击、停留时间和满意度），其中 M 是实时指标所考虑的用户行为数量。组合函数通常根据业务目标设定，其形式可以非常灵活，以捕捉实时指标之间的复杂关系。例如，加权求和和加权乘积常用于工业推荐系统。借助合成连续标签，我们可以自然地应用第 3.2 节所述的方法。

$$c=\Phi(action_1,action_2,...,action_M) \tag{18}$$

将多种用户行为结合到合成标签中有一些优势。首先，通过学习合成标签，可以弥合训练目标与实时指标之间的差距，因为合成标签与最终业务目标密切相关。其次，我们可以通过单个模型获得实时指标的无偏预测，并直接将其用于最终排序，无需进行组合，因此更加便捷高效。

4 通过期望最大化进行估计

为估计上一节介绍的 θ_i、 θ_i^{-}、 ϵ_{ij}^{+} 和 ϵ_{ij}^{-} 的参数并学习无偏排序器，本文将基于回归的 EM 方法 [27] 扩展到成对设置。在基于回归的EM算法中，通过交替执行期望步骤和最大化步骤，直至收敛，来估计参数。本节将分别说明期望步骤和最大化步骤的过程。

4.1 期望步骤

在期望步骤中，我们需要估计给定参数 $\theta_i, \epsilon_{ij}^{+}$和 ϵ_{ij}^{-}和 γ_{u,x_i,x_j}时，隐变量 e_i, e_j和相关性对 (r_i, r_j) 的分布。为实现这一目标，我们首先计算图1中所示所有隐变量的联合概率，其中所有公式均直接由贝叶斯法则推导得出。例如，在

$$\epsilon_{ij}^{+}=\frac{\sum_{u,x_i,x_j,c_i>c_j}P(e_i=1,e_j=1,r_i>r_j|u,x_i,x_j,i,j,c_i>c_j)}{\sum_{u,x_i,x_j,i,j,c_i>c_j}P(e_i=1,e_j=1,r_i>r_j|u,x_i,x_j,i,j,c_i>c_j)+\sum_{u,x_i,x_j,i,j,c_i\le c_j}P(e_i=1,e_j=1,r_i>r_j|u,x_i,x_j,i,j,c_i\le c_j)}$$

$$\epsilon_{ij}^{-}=\frac{\sum_{u,x_i,x_j,c_i>c_j}P(e_i=1,e_j=1,r_i\le r_j|u,x_i,x_j,i,j,c_i>c_j)}{\sum_{u,x_i,x_j,i,j,c_i>c_j}P(e_i=1,e_j=1,r_i\le r_j|u,x_i,x_j,i,j,c_i>c_j)+\sum_{u,x_i,x_j,i,j,c_i\le c_j}P(e_i=1,e_j=1,r_i\le r_j|u,x_i,x_j,i,j,c_i\le c_j)}$$

$$\gamma_{u,x_i,x_j}=\frac{\sum_{u'=u,x'_i=x_i,x'_j=x_j,c_i>c_j}P(r_i>r_j|u,x_i,x_j,i,j,c_i>c_j)+\sum_{u'=u,x'_i=x_i,x'_j=x_j,c_i\le c_j}P(r_i>r_j|u,x_i,x_j,i,j,c_i\le c_j)}{\sum_{u'=u,x'_i=x_i,x'_j=x_j}1}$$

$$\beta_{u,x_i}=\frac{\sum_{u'=u,x'_i=x_i,c_i>0}P(r_i>0|u,x_i,i,c_i>0)+\sum_{u'=u,x'_i=x_i,c_i\le 0}P(r_i>0|u,x_i,i,c_i\le 0)}{\sum_{u'=u,x'_i=x_i}1}$$

$$\theta_k^{-}=\frac{\sum_{u,x_i,i,c_i=0,i=k}P(e_i=1|u,x_i,i,i,c_i=0)}{\sum_{u,x_i,i,c_i=0}1\{i=k\}}\qquad\theta_k=\frac{\sum_{u,x_i,i,c_i>0,i=k}P(e_i=1|u,x_i,i,c_i>0)+\sum_{u,x_i,i,c_i=0,i=k}\theta_k^{-}}{\sum_{u,x_i,i}1\{i=k\}}$$

Figure 2: Formulas in Maximization Step

第一个方程中，我们有：

$$\begin{aligned}&P(e_i=1,e_j=1,r_i>r_j|u,x_i,x_j,i,j,c_i>c_j)\\&=\frac{P(c_i>c_j|e_i=1,e_j=1,r_i>r_j)P(e_i=1,e_j=1,r_i>r_j)}{P(c_i>c_j|u,x_i,x_j,i,j)}\\&=\frac{\theta_i\theta_j\epsilon_{ij}^{+}\gamma_{u,x_i,x_j}}{\theta_i\theta_j(\epsilon_{ij}^{+}\gamma_{u,x_i,x_j}+\epsilon_{ij}^{-}(1-\gamma_{u,x_i,x_j}))+\theta_i(1-\theta_j)\beta_{u,x_i}}\end{aligned}\tag{19}$$

值得注意的是，我们以逐点方式估计隐变量 e_i、 e_j 和真实相关性 $r_i>0$，如图1最后4个公式所示，因为它们仅取决于单个物品。计算出联合概率后，我们便可以计算隐变量的边缘概率，并在最大化步骤中使用。例如，边缘概率$P(e_i=1|c_i=0)$等于 $P(e_i=1,r_i>0|c_i=0)$与$P(e_i=1,r_i\le 0|c_i=0)$之和。

4.2 最大化步骤

在最大化步骤中，根据训练样本以及期望步骤得到的后验概率，更新所有参数以最小化损失函数。图 2 展示了所有参数更新公式。在标准 EM 算法的过程中，$\gamma_{ux_ix_j}$ 和 β_{ux_i} 分别按照 $3rd$ 公式和 $4th$ 公式进行更新，这要求（u、 x_i、 x_j）重复出现并展示在不同的位置对中。由于在大规模真实世界系统中，用户-物品对（u、 x_i、 x_j）的样本高度稀疏，且合成标签对（c_i、 c_j）含有噪声，因此很难通过自由参数 $\gamma_{ux_ix_j}$ 和 β_{ux_i} 来最小化损失。为解决这一问题，我们采用基于回归的 EM [27]算法，通过学习回归函数来估计 $\gamma_{ux_ix_j}$ 和 β_{ux_i} 的参数。具体而言，我们假设存在特征向量 $X_{ux_ix_j}$ 和 X_{ux_i}，分别表示用户-物品对（u、 x_i、 x_j）样本和用户-物品（u、 x_i）样本，并使用函数 g 计算相关性偏好 $\gamma_{ux_ix_j}=g(X_{u,x_i,x_j})$，使用函数 h 计算绝对相关性 $\beta_{ux_i}=h(X_{u,x_i})$。因此，在最大化步骤中，我们旨在寻找回归函数 $g(X)$ 和 $h(X)$，根据训练样本和期望步骤中计算出的隐变量分布来最小化损失函数。例如，我们可以将特征向量 $X_{ux_ix_j}$ 回归到概率 $P(r_i>r_j|u,x_i,x_j,i,j,c_i,c_j)$。与 [27],类似，我们基于伯努利采样将此类回归问题转换为分类问题。具体而言，我们根据 $P(r_i>r_j|u,x_i,x_j,i,j,c_i,c_j)$ 和 $P(r_i>0|u,x_i,i,c_i)$，分别采样二元相关性对标签 $\gamma\in\{0,1\}$和二元相关性标签 $\beta\in\{0,1\}$。然后，我们可以基于训练集 $\{X,\gamma\}$ 和 $\{X,\beta\}$，采用分类模型学习 $g(X)$ 和 $h(X)$。$g(X)$ 和 $h(X)$ 可以灵活地采用任何分类模型，本文选择了常用的深度神经网络模型。

请注意，原始EM算法在最大化步骤中通过对整个数据集进行计算来更新参数，这给训练数据规模庞大的工业系统带来了挑战。为解决这一问题，我们按照在线EM的思路采用小批量EM [6]。因此，我们基于单个小批量中的数据计算图2中的公式，并在每个批次中增量更新参数，具体如下：

$$\epsilon_{ij}^{+}=\epsilon_{ij}^{+}\times(1-\alpha)+\widehat{\epsilon_{ij}^{+}}\times\alpha\tag{20}$$

其中，α是 ϵ_{ij}^{+}的预定学习率，$\widehat{\epsilon_{ij}^{+}}$是当前批次中 ϵ_{ij}^{+}的估计值。

5 实验

本节首先介绍实验数据和实验设置，然后开展大量离线和在线实验，以回答以下研究问题：

- RQ1：与最先进的去偏方法相比，所提出的方法在连续标签上的表现如何?
- RQ2：与最先进的无偏学习排序方法相比，所提出的方法在类别标签上的表现如何?
- RQ3：所提出的方法在不同程度的位置偏差以及连续标签相对于类别标签的不同相对权重下是否具有鲁棒性?
- RQ4：所提出的方法在真实世界推荐系统中的表现如何?

5.1 实验数据

为回答上述研究问题，并评估所提出方法在推荐和搜索场景中的有效性，我们在学习排序和推荐领域的公开基准数据集上进行了实验，具体如下：

	Yahoo 数据集			MovieLens		
	奖励@3	奖励@5	奖励@10	奖励@3	奖励@5	奖励@10
我们的方法 ($loss_{opt}$)	**2.3319** ± 6.9e-3	**2.7185** ± 7.5e-3	**2.9735** ± 5.9e-3	5.8553 ± 1.1e-2	**7.6690** ± 1.5e-2	**9.1400** ± 1.6e-2
我们的方法（$loss_{Bayes-IPW}$）	2.3046 ± 8.5e-3	2.6864 ± 7.5e-3	2.9423 ± 8.0e-3	5.8415 ± 1.4e-2	7.6486 ± 7.8e-3	9.1217 ± 9.7e-3
我们的方法 ($loss_{IP\ W}$)	2.2447 ± 9.9e-3	2.6239 ± 1.0e-2	2.8828 ± 8.9e-3	**5.8584** ± 2.0e-2	7.6635 ± 2.1e-2	9.1285 ± 2.5e-2
求和-合成标签	2.2599 ± 5.0e-3	2.6380 ± 8.1e-3	2.8913 ± 8.7e-3	5.7863 ± 2.5e-2	7.5704 ± 3.1e-2	9.0289 ± 3.5e-2
PAL-合成	2.2612 ± 8.7e-3	2.6388 ± 6.0e-3	2.8914 ± 5.6e-3	5.7484 ± 3.1e-2	7.5271 ± 3.7e-2	8.9682 ± 4.0e-2
求和-点击	2.2776 ± 1.2e-2	2.6590 ± 9.7e-3	2.9146 ± 8.9e-3	5.7283 ± 2.1e-2	7.5570 ± 2.3e-2	9.0264 ± 2.8e-2
PAL-Click	2.2907 ± 9.9e-3	2.6719 ± 6.6e-3	2.9274 ± 6.7e-3	5.7385 ± 1.9e-2	7.5838 ± 1.8e-2	9.0586 ± 1.9e-2
PAL-Click + PAL-停留时间	2.2707 ± 7.7e-3	2.6504 ± 7.3e-3	2.9058 ± 6.9e-3	5.7569 ± 2.7e-2	7.5491 ± 3.5e-2	9.0031 ± 3.7e-2
下界（逐点深度神经网络）	2.2548 ± 4.4e-3	2.6353 ± 6.5e-3	2.8892 ± 5.3e-3	5.7106 ± 1.3e-2	7.5392 ± 1.6e-2	9.0113 ± 1.9e-2
下界（LambdaRank）	2.1619 ± 1.2e-2	2.5370 ± 8.5e-3	2.7957 ± 8.4e-3	5.6800 ± 2.1e-2	7.4805 ± 2.9e-2	8.9387 ± 4.7e-2

表2：连续标签上的实验结果

• 学习排序数据集：Yahoo! LETOR set 1 和 MSLR-Web30K [3] 是规模最大的公开学习排序基准数据集中的两个，包含多个查询-文档对，这些查询-文档对由特征向量表示，并带有 5 级相关性标签。具体而言，Yahoo 数据集包含 29.9k 个查询和 710k 个文档，MSLR-Web30K 包含 31.2k 个查询。我们采用这些语料库中的训练、验证和测试划分。

• 推荐数据集：MovieLens [4] 是一个推荐基准数据集，包含 6,000 位用户对 3,900 部电影的 100 万条无偏评分。由于 MovieLens 中的每位用户和每部电影都仅以 ID 表示，没有其他信息，因此我们通过以下步骤为所有用户-电影对生成特征向量。首先，根据时间顺序将每位用户的样本划分为训练集（50%）、验证集（25%）和测试集（25%）。然后，我们基于训练集训练贝叶斯个性化排序（BPR）[20] 模型，为每位用户和每个物品分别获取一个 60 维特征向量。最后，将用户特征向量、物品特征向量以及用户和物品特征向量的点积拼接起来，为每个用户-物品对生成一个 180 维特征向量。

由于这些数据集中没有用户行为，我们将每个查询或用户视为一个用户请求，并生成点击和停留时间标签，以模拟有偏反馈。

5.1.1 点击模拟。我们按照 [2] 中的流程对点击进行采样。首先，使用带有相关性标签的训练数据中的 1% 训练一个排序支持向量机 [5] 模型，为每位用户生成初始排序列表。然后，通过模拟 PBM 的浏览过程，按如下方式均匀采样点击标签$click_{u,\ x_i,\ i}$：

$$click_{u,x_i,i} \sim U(P(c_i = 1|u, x_i, i)),$$
$$P(c_i = 1|u, x_i, i) = P(e_i = 1|i)P(r_i = 1|u, x_i), \quad (21)$$

其中，$P(e_i = 1|i) = \theta_i^{\eta}$ 表示检查概率，θ_i是由初始 Ranking-SVM排序器确定的位置 i 上点击的位置偏差，η控制位置偏差的严重程度。θ_i取自 [14] 中的一项眼动追踪实验，η 默认设为 1。物品 x_i 被感知为相关物品的概率计算如下：

$$P(r_i = 1|u, x_i) = \epsilon + (1 - \epsilon)\frac{2^{y_{u,x_i}} - 1}{2^{y_{max}} - 1},$$

其中，$y_{u,\ x_i} \in [0, 4]$ 是 5 级相关性标签，y_{max}是最高级 4。我们将 MovieLens 中的评分从 [1-5] 映射到 [0-4]，作为相关性标签。ϵ对点击噪声进行建模，因此不相关物品（$y_{u,\ x_i}$=0）也具有非零点击概率。ϵ 默认设为 0.1。

5.1.2 停留时间模拟。接着，我们根据采样得到的点击数据，对连续停留时间标签进行采样，因为在实际情况中，只有被点击的样本才具有非零停留时间。如 [30], 所示，停留时间与相关性和项目长度相关；长度相同的物品，其停留时间服从对数高斯分布。因此，我们分析了真实视频推荐系统中时长约为 100 秒的视频的停留时间，也发现其服从对数高斯分布。此外，我们假设不同相关性等级下的停留时间分别服从不同的对数高斯分布，而总体停留时间的对数高斯分布由这些分布组成。然后，我们对经过对数变换的停留时间训练一个高斯混合模型（高斯混合模型）[21], 以获得对应不同相关性等级的 5 个高斯分布，并按如下方式采样停留时间：

$$dt_{u,x_i,i} = click_{u,x_i,i} \cdot e^{\omega_{u,x_i}}, \quad \omega_{u,x_i} \sim N(\mu_{y_{u,x_i}}, \sigma_{y_{u,x_i}}), \quad (22)$$

其中$click_{u,\ x_i,\ i}$是采样得到的点击标签，$\omega_{u,\ x_i}$表示点击后的采样对数缩放停留时间，$\mu_{y_{u,\ x_i}}$和$\sigma_{y_{u,\ x_i}}$分别是通过高斯混合模型获得的不同相关性等级下的均值和标准差。

5.2 实验设置

本文在第5.3小节的合成标签和第5.4小节的点击标签上开展实验，分别评估我们的方法在连续标签和类别标签上的性能。基于第3.3小节，我们采用典型推荐场景中常用的加权求和函数，并将合成标签$c_{u,\ x_i,\ i}$定义如下：

$$c_{u,x_i,i} = click_{u,x_i,i} + \frac{dt_{u,x_i,i}}{e^{\delta}},$$

其中，δ控制连续标签在合成标签中的权重。δ默认设为3，以使停留时间的权重与我们的真实世界推荐系统中的权重相近。

在实验中，我们采用一个具有 [512, 256, 128]个隐藏单元、使用 ELU 激活函数和 Adagrad 优化器的深度神经网络模型；对于二元分类排序器和成对排序器，采用交叉熵损失

[2]https://webscope.sandbox.yahoo.com/catalog.php?datatype=c [3] https://www.microsoft.com/en-us/research/project/mslr/ [4] https://files.grouplens.org/datasets/movielens/ml-1m-README.txt [5] http://www.cs.cornell.edu/people/tj/svm_light/svm_rank.html

	Yahoo 数据集			MSLR-Web30K		
	NDCG@3	NDCG@5	NDCG@10	NDCG@3	NDCG@5	NDCG@10
上界	0.6960 ± 1.4e-3	0.7175 ± 1.3e-3	0.7644 ± 1.0e-3	0.3960 ± 2.3e-3	0.4047 ± 1.8e-3	0.4286 ± 1.8e-3
我们的方法 ($loss_{opt}$)	0.6895 ± 2.5e-3	**0.7115** ± 2.1e-3	**0.7589** ± 1.4e-3	**0.3811** ± 2.6e-3	**0.3899** ± 1.7e-3	**0.4132** ± 1.0e-3
我们的方法 ($loss_{Bayes-IP\ W}$)	**0.6900** ± 2.4e-3	**0.7116** ± 1.7e-3	**0.7589** ± 1.5e-3	0.3800 ± 2.5e-3	0.3888 ± 1.8e-3	0.4125 ± 1.1e-3
我们的方法 ($loss_{IP\ W}$)	0.6860 ± 1.6e-3	0.7079 ± 1.3e-3	0.7557 ± 1.2e-3	0.3784 ± 1.4e-3	0.3882 ± 1.4e-3	0.4119 ± 1.5e-3
DLA [2]	0.6892 ± 2.1e-3	0.7106 ± 2.2e-3	0.7578 ± 1.6e-3	0.3792 ± 2.7e-3	0.3870 ± 2.4e-3	0.4102 ± 2.9e-3
回归 EM [27]	0.6893 ± 2.1e-3	0.7108 ± 2.1e-3	0.7577 ± 1.6e-3	0.3707 ± 3.9e-3	0.3766 ± 3.7e-3	0.3977 ± 3.6e-3
成对去偏 [10]	0.6600 ± 3.5e-3	0.6847 ± 2.8e-3	0.7365 ± 2.0e-3	0.3481 ± 3.3e-3	0.3623 ± 2.8e-3	0.3912 ± 2.4e-3
下界 (LambdaRank)	0.6541 ± 2.5e-3	0.6790 ± 2.2e-3	0.7320 ± 1.7e-3	0.3433 ± 2.2e-3	0.3579 ± 2.1e-3	0.3874 ± 1.7e-3

表3：类别标签上的实验结果

，对于回归排序器，则采用均方误差损失。此外，我们针对每种方法仔细调节学习率，范围为

[5 3, e-5 e-2] 。为进行公平比较，我们对每种方法训练并评估 10 次，并报告评估指标的均值和标准差。为了评估我们的方法在无偏学习排序方面的表现，我们在类别标签实验中采用 NDCG@k 作为评估指标。对于连续标签，我们采用典型推荐应用中常用的加权求和函数，将点击和停留时间结合起来，并将评估指标定义为 $Reward@k$，以模拟组合实时指标：

$$Reward@k = \sum_{i=1}^{k} click'_{u,x_i,i} + \frac{dt'_{u,x_i,i}}{e^{\delta}},$$

其中，$click_{u、x_i、i}$和 $dt_{u、x_i、i}$分别表示在排序位置 i处估计的点击和停留时间反馈。不同于 $click_{u、x_i、i}$，此处的位置 i由最终的无偏学习排序排序器决定，而不是由初始的 Ranking-SVM排序器决定。具体而言，$click_{u、x_i、i}$和 $dt_{u、x_i、i}$分别基于公式21和公式22进行采样。

5.3 连续标签上的实验（RQ1）

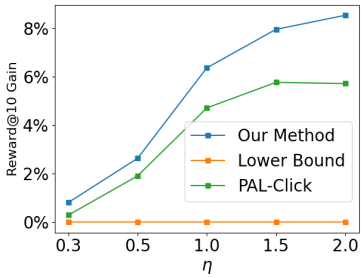
我们首先在 Yahoo 数据集和 MovieLens 的连续合成标签上开展实验，以评估我们的方法在实际应用中的性能。为进行全面比较，我们使用合成标签训练一个逐点深度神经网络和 LambdaRank 模型，但不进行去偏，将其作为下界。此外，我们从我们的方法中移除成对信任偏差校正和用于直接指标优化的损失精炼，以探究它们的有效性。需要注意的是，最先进水平的工业推荐系统会分别训练多个逐点模型来预测多个目标，并采用 PAL [8] 或基于求和的浅层塔 [31] 进行去偏。我们考虑以下基线方法：

- 求和-合成标签：我们使用合成标签训练排序器，并采用基于求和的浅层塔进行去偏。
- PAL-合成：我们使用合成标签训练排序器，并采用 PAL 进行去偏。
- 求和-点击：我们分别训练点击和停留时间任务，并在点击任务中采用基于求和的浅层塔进行去偏。
- PAL-Click：我们分别训练点击和停留时间任务，并在点击任务中采用 PAL 进行去偏。
- PAL-Click + PAL-停留时间：我们分别训练点击和停留时间任务，并在两个任务中都采用 PAL。

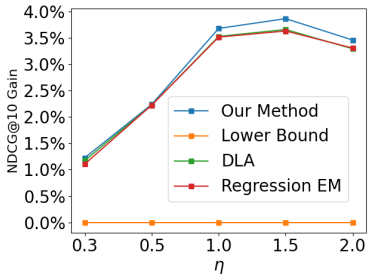
对于包含多个任务的基线模型，我们将预测的点击概率和停留时间通过与合成标签相同的加权求和函数进行组合，用于排序。如表2所示，我们的方法在两个数据集上都显著优于所有基线方法。可以看出，使用有偏合成标签的 LambdaRank 表现不如逐点深度神经网络，这表明在推荐场景中，去偏对成对学习更为重要，也可以解释为什么逐点方法曾是推荐系统的主流排序器。此外，在停留时间任务中采用浅层塔会显著降低性能，这说明浅层塔无法很好地处理连续标签。相比之下，我们基于概率图的方法能够清晰地区分偏差和相关性，在不同应用的去偏和偏好学习方面表现更优。此外，直接优化排序指标能在两个数据集上带来显著提升。除此之外，成对信任偏差校正在 Yahoo 数据集上的表现更好，但在 MovieLens 上的表现相近。为了探究成对信任偏差校正在 Yahoo 数据集和 MovieLens 上表现不同的原因，我们分析了这些数据集中m_{ij}公式14中定义的统计量，发现m_{ij}在 MovieLens 的不同位置对中大致相似，而在 Yahoo 数据集中差异很大，这表明成对信任偏差校正在 MovieLens 上没有提升空间。

5.4 类别标签实验（RQ2）

我们还在 Yahoo 数据集和 Web30K 的采样二元点击标签上进行了实验，将我们的方法与最先进的无偏学习排序方法进行比较，包括 DLA [2], 回归 EM [27] 和成对去偏 [10] ，以回答 RQ2。为进行全面比较，我们使用无偏相关性标签和未经去偏的有偏点击标签训练 LambdaRank 模型，分别作为上界和下界。表3显示，我们的方法在类别标签上与最先进的无偏学习排序方法相比表现具有竞争力。此外，成对信任偏差校正使 Yahoo 数据集和 Web30K 上的性能均得到持续提升。此外，排序指标的直接优化在 Web30K 上表现更好，但在 Yahoo 数据集上的表现相近，因为我们只能根据噪声点击信号计算 ΔZ_{ij}



(a) 连续标签



(b) 类别标签

图 3：不同位置偏差程度下相对于下界的性能提升

总的来说，我们的方法在连续标签和类别标签上都取得了具有竞争力的表现。

5.5 鲁棒性分析（RQ3）

本节中，为了评估所提方法的鲁棒性，我们在不同程度的位置偏差和不同连续标签权重下，将其与 Yahoo 数据集上的最佳基线方法（即 DLA、回归 EM 和 PAL-Click）进行比较。

5.5.1 位置偏差的严重程度。图 3 展示了在由 η 控制的不同程度的位置偏差下，各去偏方法在连续标签和类别标签上的表现相对于下界（LambdaRank）的提升幅度。表现提升幅度定义为评估指标相对于下界的相对改善。可以看出，我们的方法对位置偏差的严重程度具有鲁棒性，并且在不同的 η 下始终优于其他竞争方法。

5.5.2 连续标签的相对权重。根据图 4，在由 δ 控制的不同连续标签权重下，我们的方法始终表现最佳。总体而言，我们的方法对于不同程度的位置偏差和不同的连续标签权重均具有鲁棒性。

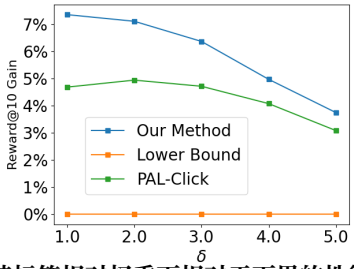


图 4：不同连续标签相对权重下相对于下界的性能提升

5.6 在线 A/B 测试（RQ4）

为评估所提出方法在真实系统中的性能，我们在腾讯新闻大规模视频推荐系统的排序模块中进行了为期 7 天的严谨在线 A/B 测试。系统中的实时指标和合成标签定义为多种用户行为的组合（例如点击、停留时间、点赞和分享）。由于 PAL-Click 是在连续标签上表现最佳的基线方法，我们分别针对多种用户行为进行训练，并在点击任务中采用 PAL 进行去偏，作为在线 A/B 测试的基线。我们在自己的深度学习框架中实现了所有模型，将用户随机分配到多个分桶中，并将每个模型部署到其中一个分桶。与 PAL-Click 相比，我们的方法使实时指标提升了 2.08%，这证明了我们的方法在真实应用中进行去偏和偏好学习的有效性。

为进一步研究我们的方法在在线 A/B 测试中的表现，我们分析了我们的方法在每个位置学习得到的倾向概率，记为 θ_i，并将其与系统中每个位置的正标签统计概率（记为 $P(c_i > 0)$）进行比较。图 5 显示，θ_i 在较低位置时更小，这与 $P(c_i > 0)$ 的下降趋势一致。我们还根据公式 3，分析了每个位置的相关性估计概率，记为$P(c_i>0)\ \theta_i$。结果表明，$P(c_i>0)\ \theta_i$也呈现出阶梯式下降趋势，尤其是在较高位置，这同样与推荐机制一致，即相关性更高的物品通常会展示在排名更高的位置。总而言之，我们的方法能够在真实世界的推荐应用中有效地学习并纠正位置偏差。

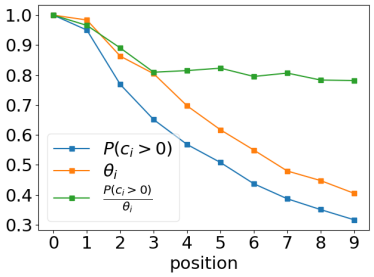


图 5：统计概率与学习得到的倾向概率

6 结论

本文提出了一种新颖的无偏成对学习排序方法，以成对方式对基于位置的检查偏差和信任偏差进行建模，从而更好地校正偏差并学习偏好。这是首个同时适用于类别标签和连续标签的无偏学习排序方法。在公开基准数据集上的实验结果和在线 A/B 测试表明，对于连续标签，该方法显著优于最先进的排序模型；对于类别标签，其表现则可与最先进的无偏学习排序方法相媲美。未来工作将重点研究如何校正位置偏差以外的其他类型偏差。

参考文献

[1] Aman Agarwal、Xuanhui Wang、Cheng Li、Michael Bendersky 和 Marc Najork。2019。解决无偏学习排序中的信任偏差。载于 *万维网大会*。4–14。[2] Qingyao Ai、Keping Bi、Cheng Luo、Jiafeng Guo 和 W Bruce Croft。2018。通过无偏倾向性估计实现无偏学习排序。载于 *第41届 ACM 国际信息检索研究与发展会议*。385–394。[3] Christopher Burges、Robert Ragno 和 Quoc Le。2006。使用非光滑成本函数进行学习排序。*神经信息处理系统进展* 19（2006），193–200。[4] Chris Burges、Tal Shaked、Erin Renshaw、Ari Lazier、Matt Deeds、Nicole Hamilton 和 Greg Hullender。2005。使用梯度下降进行学习排序。载于 *第22届国际机器学习会议论文集*。89–96。[5] Christopher JC Burges。2010。从 RankNet 到 LambdaRank 再到 LambdaMART：综述。*学习* 11，23-581（2010），81。[6] Olivier Cappé 和 Eric Moulines。2009。潜在数据模型的在线期望最大化算法。*皇家统计学会杂志B辑（统计方法论）* 71，3（2009），593–613。[7] Jiawei Chen、Hande Dong、Xiang Wang、Fuli Feng、Meng Wang 和 Xiangnan He。2020。推荐系统中的偏差与去偏：综述及未来方向。*arXiv 预印本 arXiv:2010.03240*（2020）。[8] Huifeng Guo、Jinkai Yu、Qing Liu、Ruiming Tang 和 Yuzhou Zhang。2019。PAL：面向在线推荐系统中点击率预测的位置偏差感知学习框架。载于 *第13届 ACM 推荐系统会议论文集*。452–456。[9] Yifan Hu、Yehuda Koren 和 Chris Volinsky。2008。隐式反馈数据集的协同过滤。载于 *2008年第八届 IEEE 数据挖掘国际会议*。IEEE，263–272。[10] Ziniu Hu、Yang Wang、Qu Peng 和 Hang Li。2019。无偏 LambdaMART：一种无偏成对学习排序算法。载于 *万维网大会*。2830–2836。[11] Jui-Ting Huang、Ashish Sharma、Shuying Sun、Li Xia、David Zhang、Philip Pronin、Janani Padmanabhan、Giuseppe Ottaviano 和 Linjun Yang。2020。Facebook搜索中的基于嵌入的检索。载于 *第26届 ACM SIGKDD知识发现与数据挖掘国际会议论文集*。2553–2561。[12] Kalervo Järvelin 和 Jaana Kekäläinen。2002。基于累积增益的信息检索技术评估。*ACM信息系统汇刊（TOIS）* 20，4（2002），422–446。[13] Kalervo Järvelin 和 Jaana Kekäläinen。2017。检索高度相关文档的信息检索评估方法。载于 *ACM SIGIR论坛*，第51卷。ACM，纽约，纽约州，美国，243–250。[14] Thorsten Joachims、Laura Granka、Bing Pan、Helene Hembrooke 和 Geri Gay。2017。准确地将点击数据解读为隐式反馈。载于 *ACM SIGIR论坛*，第51卷。ACM，纽约，纽约州，美国，4–11。[15] Thorsten Joachims、Adith Swaminathan 和 Tobias Schnabel。2017。基于有偏反馈的无偏学习排序。载于 *第十届 ACM 网络搜索与数据挖掘国际会议论文集*。781–789。[16] Christopher C Johnson。2014。用于隐式反馈数据的逻辑矩阵分解。*神经信息处理系统进展* 27，78（2014），1–9。[17] Yehuda Koren、Robert Bell 和 Chris Volinsky。2009。推荐系统的矩阵分解技术。*计算机*42，8（2009），30–37。[18] Jae-woong Lee、Seong min Park 和 Jongwuk Lee。2021。面向隐式反馈的双重无偏推荐学习。载于 *第44届 ACM SIGIR 信息检索研究与开发国际会议论文集*。1647–1651。[19] Dawen Liang、Laurent Charlin、James McInerney 和 David M Blei。2016。推荐中的用户曝光建模。载于 *第25届万维网国际会议论文集*。951–961。[20] Steffen Rendle、Christoph Freudenthaler、Zeno Gantner 和 Lars Schmidt-Thieme。2012。BPR：基于隐式反馈的贝叶斯个性化排序。*arXiv 预印本 arXiv:1205.2618*（2012）。[21] Douglas A Reynolds。2009。高斯混合模型。*生物识别百科全书*741，659-663（2009）。[22] Matthew Richardson、Ewa Dominowska 和 Robert Ragno。2007。点击预测：估算新广告的点击率。载于 *第16届万维网国际会议论文集*。521–530。[23] Yuta Saito。2020。基于有偏隐式反馈的无偏成对学习。载于 *2020年 ACM SIGIR 信息检索理论国际会议论文集*。5–12。[24] Yuta Saito、Suguru Yaginuma、Yuta Nishino、Hayato Sakata 和 Kazuhide Nakata。2020。基于非随机缺失隐式反馈的无偏推荐学习。载于 *第13届网络搜索与数据挖掘国际会议论文集*。501–509。[25] Hongyan Tang、Junning Liu、Ming Zhao 和 Xudong Gong。2020。渐进式分层提取（PLE）：一种用于个性化推荐的新型多任务学习（MTL）模型。载于 *第十四届 ACM 推荐系统会议*。269–278。[26] Xuanhui Wang、Michael Bendersky、Donald Metzler 和 Marc Najork。2016。个性化搜索中考虑选择偏差的学习排序。载于 *第39届 ACM SIGIR 信息检索研究与开发国际会议论文集*。115–124。[27] Xuanhui Wang、Nadav Golbandi、Michael Bendersky、Donald Metzler 和 Marc Najork。2018。个性化搜索中无偏学习排序的位置偏差估计。载于 *第十一届 ACM 网络搜索与数据挖掘国际会议论文集*。610–618。[28] Xuanhui Wang、Cheng Li、Nadav Golbandi、Michael Bendersky 和 Marc Najork。2018。用于排序指标优化的 LambdaLoss 框架。载于 *第27届 ACM 信息与知识管理国际会议论文集*。1313–1322。[29] Xinwei Wu、Hechang Chen、Jiashu Zhao、Li He、Dawei Yin 和 Yi Chang。2021。信息流推荐中的无偏学习排序。载于 *第14届 ACM 网络搜索与数据挖掘国际会议论文集*。490–498。[30] Peifeng Yin、Ping Luo、Wang-Chien Lee 和 Min Wang。2013。沉默也是证据：从心理学角度解读推荐中的停留时间。载于 *第19届 ACM SIGKDD 知识发现与数据挖掘国际会议论文集*。989–997。[31] Zhe Zhao、Lichan Hong、Li Wei、Jilin Chen、Aniruddh Nath、Shawn Andrews、Aditee Kumthekar、Maheswaran Sathiamoorthy、Xinyang Yi 和 Ed Chi。2019。推荐接下来要观看的视频：一种多任务排序系统。载于 *第13届 ACM 推荐系统会议论文集*。43–51。