

# 论文《Dual-Task Learning for Multi-Behavior Sequential Recommendation》复现报告

## 摘要

近年来，序列推荐成为研究热点，而利用用户异构交互序列信息的多行为序列推荐受到的关注相对较少。现有工作在解决多行为序列推荐问题时往往忽视了不同建模视角的互补效应。此外，仍有两个挑战需要解决：一是用户意图和上下文信息的行为异构型，二是购买行为交互记录的稀疏性。为发挥多行为交互序列的潜力，作者提出了一个新颖的模型NextIP，该模型采用双任务学习策略，同时考虑下一物品预测任务和购买行为预测任务。在下一物品预测任务中，作者设计了一个目标行为感知的上下文聚合器，通过利用下一个行为来指导各类特定行为的物品子序列共同预测下一个交互的物品。在购买行为预测任务中，作者设计了一个行为感知的自注意力机制，通过自监督对比学习任务帮助模型更有效地学习用户的购买偏好。本报告在原模型的基础上，加入了一个行为感知的加权二元交叉熵损失函数，以更好地学习用户的购买偏好，并在两个公开数据集上进行了实验，验证了模型各模块的有效性和改进方法所带来的性能提升。

**关键词：**多行为序列推荐；自注意力机制；对比学习

## 1 引言

在电子商务 [21]、教育 [25]和招聘 [12]等各个领域，推荐系统可以缓解信息过载的问题并为用户推荐他们感兴趣的商品、文章、视频等。在推荐系统的不同研究分支中，基于用户对某类行为的先前交互记录来预测下一交互物品的序列推荐问题是一个研究热点。

然而，在用户行为数据稀疏的场景下，仅仅利用单一类型行为的交互数据进行序列推荐难以捕获用户精确的偏好，导致推荐效果不佳。以电子商务场景为例，一方面，企业的最终目标是引导用户购买商品，而购买数据通常非常稀少。另一方面，现实中的电子商务平台往往包含一些更为丰富的辅助行为，例如，浏览、加入购物车和收藏等。因此，从各种行为类型的交互记录中挖掘用户的偏好和意图对于提高用户购买行为的预测准确率是非常有必要的。

关于多行为序列推荐的先前工作大致可以分为两个分支：一个是对行为无关的物品序列和相应的行为类型序列进行建模，另一个是将一个多行为交互序列划分为一些行为特定的物品子序列，以便于后期的联合建模。此外，对用户一些辅助行为到目标行为的转移模式的建模有利于理解用户的购买导向偏好。例如，用户在选择一款智能手机之前，可能会比较不同品牌的智能手机，其中从先前的浏览到最终购买的转变通常表明了用户真正的购买偏好。

本报告复现的NextIP模型 [16]从一个新的角度对多行为交互序列进行建模，同时考虑下一交互物品的预测任务和购买行为的预测任务。该模型将用户的多行为交互序列，视为一条行为无关的物品序列、一条行为类型序列和多条特定行为的物品子序列。首先，为了解决意图的行为异构型问题，并在不同程度上传递行为特异性的知识，该模型设计了一个目标行为感知的上下文聚合器（**target-behavior aware context aggregator**, TBCG）通过利用下一个行为来指导各类特定行为的物品子序列共同预测下一个交互的物品。其次，该模型侧重于将真实的购买行为与之前其他的辅助行为进行区分。具体来说，作者设计了一个行为感知的自注意力（**behavior-aware self-attention**, BSA）机制来提取用户在一定时间间隔内的特定行为兴趣，该模型将用户在某一时间步的偏好抽象为一个虚拟物品的表征向量，并将其作为负样本，而购买行为作为自监督对比学习任务的正样本，以帮助模型更有效地学习购买偏好。

## 2 相关工作

### 2.1 单行为序列推荐

早期的序列推荐工作主要基于马尔科夫链和矩阵分解模型 [10, 20]。随着深度学习的发展，各种技术被应用到序列推荐中，例如基于门控循环单元的GRU4Rec [7]、基于卷积神经网络的Caser [23]、基于自注意力机制的SASRec [11]等。BERT4Rec [22]借助NLP中的完形填空任务使用自监督的方式双向训练模型。FISSA [14]在SASRec的基础上引入了一个全局偏好的学习模型，对序列推荐任务十分有效。

### 2.2 多行为推荐

多行为推荐的一些工作基于矩阵分解模型。例如，ABPR [19]自适应地学习不同辅助行为的权重。近年来，基于深度学习的方法在多行为推荐中表现出色。例如，NMTR [4]扩展了NCF [6]，通过多任务框架处理多行为数据。此外，许多研究者提出利用图神经网络来捕捉用户和物品之间的不同关系，例如MBGCN [9]构建了一个包含多种边的（用户，物品）二部图，并使用图卷积网络增强表示学习。MB-GMN [27]提出了图元网络来捕捉不同行为语义和关系，在多行为推荐中表现出色。VAE++ [17]是一种基于变分自编码器的方法，同时建模目标行为、辅助行为及混合行为。

### 2.3 多行为序列推荐

早期的多行为序列推荐主要使用循环神经网络 [1, 2, 8]进行建模，例如RLBL [15]通过行为转移矩阵捕获异构行为的相互影响，RIB [28]同时使用GRU和注意力机制的优势进行建模。为了更好地将多行为信息融入到模型中，更多的基于图卷积网络 [13]的方法被提出，例如MGNN-Spred [26]通过构建行为转移子图辅助学习用户的表征，MKM-SR [18]采用门控图神经网络对会话的混合行为物品序列进行建模，并对相应的行为序列进行建模，以获得每一步的行为表示。基于注意力机制 [24]的方法在多行为序列推荐相关的任务中有较好的表现，例如DMT [5]提出了一个深度兴趣Transformer，为后续的点击率预测任务建模每个特定行为的物品序列。

### 3 本文方法

#### 3.1 本文方法概述

本文复现的NextIP模型框架如图 1 所示。该模型首先使用一个嵌入层将异构交互序列编码为嵌入向量序列，即嵌入矩阵，然后将嵌入矩阵用于两个不同的任务，在下一物品预测任务中学习用户下一交互物品的偏好，在购买行为预测任务中区分不同行为对用户购买意图的影响。在下面的章节中，本文将详细介绍NextIP模型中各个组件的设计。

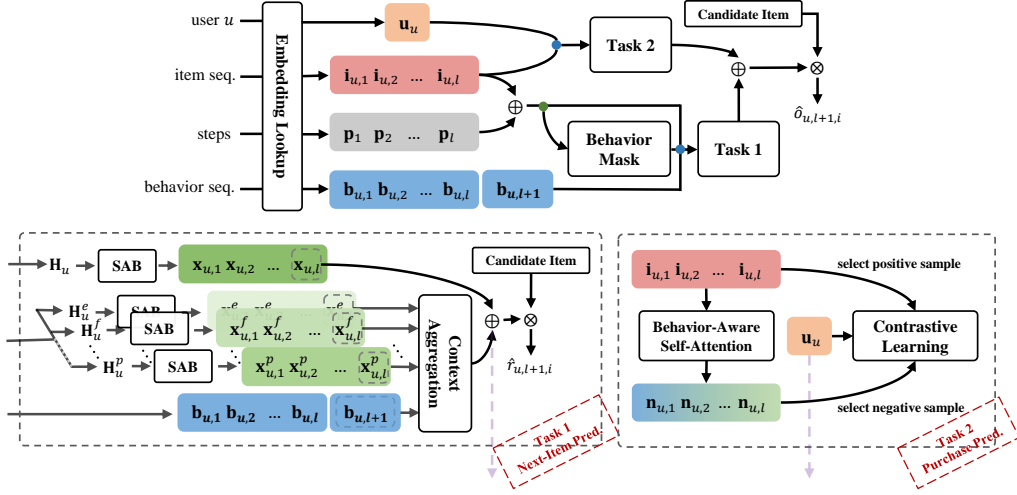


图 1. NextIP模型示意图

#### 3.2 嵌入层

该模型使用了一个嵌入层，将多行为交互序列 $S_u$ 转化为多种不同的嵌入矩阵。对于用户集合 $\mathcal{U}$ 、物品集合 $\mathcal{I}$ 、行为集合 $\mathcal{B}$ ，分别使用了三个可学习的嵌入矩阵 $\mathbf{U} \in \mathbb{R}^{|\mathcal{U}| \times d}$ 、 $\mathbf{I} \in \mathbb{R}^{|\mathcal{I}| \times d}$ 、 $\mathbf{B} \in \mathbb{R}^{|\mathcal{B}| \times d}$ ，将用户、物品和行为映射到隐向量空间中，另外为了区分不同时间步的交互记录，还引入了一个位置嵌入矩阵 $\mathbf{P} \in \mathbb{R}^{L \times d}$ ，其中 $L$ 表示序列长度。因此，对于用户 $u$ 而言，其多行为交互序列 $S_u$ 可以表示为，位置无关的嵌入矩阵 $\mathbf{S}_u = [\mathbf{i}_{u,1}, \mathbf{i}_{u,2}, \dots, \mathbf{i}_{u,L}]$ ，其中 $\mathbf{i}_{u,l} \in \mathbf{I}$ 表示用户 $u$ 在时间步 $l$ 所交互的物品的嵌入向量，加入位置相关性后，嵌入矩阵为 $\mathbf{H}_u = \mathbf{S}_u + \mathbf{P}$ 。

为了捕获用户不同行为的偏好，该模型进一步通过行为特定的掩码获取用户行为特定的同构项目序列。不失一般性，以浏览行为为例，

$$\mathbf{H}_u^e = \mathbf{H}_u \otimes \mathbf{M}^e, \quad (1)$$

其中 $\otimes$ 是按位乘法。 $\mathbf{M}^e = [\mathbf{m}_{u,1}^e, \mathbf{m}_{u,2}^e, \dots, \mathbf{m}_{u,L}^e] \in \mathbb{R}^{L \times d}$ 是一个行为特定的掩码矩阵， $\mathbf{m}_{u,l}^e \in \{0, 1\}^d$ 是一个二值向量，若用户 $u$ 在时间步 $l$ 的行为类型是浏览，则 $\mathbf{m}_{u,l}^e$ 为一个全1向量，否则为一个全0向量。使用类似地方法可以收藏、加购和购买行为相应的同构项目序列的嵌入矩阵 $\mathbf{H}_u^f$ 、 $\mathbf{H}_u^c$ 和 $\mathbf{H}_u^p$ 。

### 3.3 任务一：下一物品预测

第一个任务是以行为感知的方式预测下一个物品，通过统一和互补地利用行为序列以及行为特定和行为无关的物品序列。在这个任务中，该方法首先在物品序列编码器中编码行为无关的偏好和特定行为的上下文，然后引入一个新颖的目标行为感知上下文聚合器（TBCG）来聚合不同类型行为的独特知识。

#### 3.3.1 物品序列编码器

为了在所有行为无关和行为特定的序列中获取不同的上下文信息，该模型选择了自注意力块（self-attention block, SAB）[11]作为物品序列编码器，因为它已被证明是有效的。此外，每个自注意力网络的参数在不同类型的物品序列中是独立的，用于捕捉不同行为所特有的语义信息和转移模式。例如，浏览行为的物品序列中的转移通常表示物品之间的相似性，而购买物品序列中的转移则通常意味着互补性。省略随机丢弃（dropout）、层归一化（layer normalization）和残差连接（skip connect）的公式，每个SAB由一个自注意力层（self-attention layer, SAL）和一个前馈层（feed-forward layer, FFL）组成，

$$\text{SAB}(\mathbf{X}) = \text{FFL}(\text{SAL}(\mathbf{X})), \quad (2)$$

$$\mathbf{X}' = \text{SAL}(\mathbf{X}) = (\text{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}) \otimes \Delta) \mathbf{V}, \quad (3)$$

$$\text{FFL}(\mathbf{X}') = \text{ReLU}(\mathbf{X}'\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2, \quad (4)$$

其中 $\mathbf{Q} = \mathbf{X}\mathbf{W}_Q$ 、 $\mathbf{K} = \mathbf{X}\mathbf{W}_K$ 、 $\mathbf{V} = \mathbf{X}\mathbf{W}_V$ ，并且 $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d \times d}$ 是可学习的参数矩阵， $\sqrt{d}$ 是缩放因子用于避免内积的值过大， $\Delta$ 是一个掩码矩阵，用于屏蔽未来信息。 $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d \times d}$ 和 $\mathbf{b}_1, \mathbf{b}_2 \in \mathbb{R}^d$ 是前馈层中可学习的权重和偏置。

通过将 $\mathbf{X}_u^{(0)} = \mathbf{H}_u$ 输入到 $K$ 层SAB中，该模型可以得到一个行为无关的物品序列上下文表征矩阵 $\mathbf{X}_u^{(K)} = [\mathbf{x}_{u,1}^{(K)}; \dots; \mathbf{x}_{u,l}^{(K)}; \dots; \mathbf{x}_{u,L}^{(K)}]$ ，其中 $\mathbf{x}_{u,l}^{(K)}$ 表示用户 $u$ 在第 $l$ 个时间步行为无关的上下文。将特定行为 $b$ 的同构物品序列 $\mathbf{X}_u^{b(0)} = \mathbf{H}_u^b$ 放入到 $K$ 层的自注意力块中，得到了行为特定的物品序列上下文 $\mathbf{X}_u^{b(K)}$ ，其中， $\mathbf{x}_{u,l}^{b(K)}$ 表示用户 $u$ 在第 $l$ 个时间步特定行为 $b$ 的上下文， $b \in \{e, f, c, p\}$ ， $e$ 表示浏览行为， $f$ 表示收藏行为， $c$ 表示加购行为， $p$ 表示购买行为。

#### 3.3.2 目标行为感知的上下文聚合器（TBCG）

不同特定行为类型的物品序列可以提供不同的上下文信息，因此应明确不同行为的区别并加以利用。例如，用户最近浏览运动鞋的行为是一个有用的信号，这表明下一个购买可能就是运动鞋，而知道用户上个月购买了一套健身器材可以帮助模型预测用户可能会购买其互补产品如运动服或健身手环。然而，在预测不同行为时，需要平衡上下文信息，因为不同行为之间的依赖关系是复杂的。因此，该模型使用了一个目标行为感知的上下文聚合器，它利用用户的目标行为嵌入来引导不同行为上下文的聚合。具体来说，在当前步骤 $l$ ，该模块将不同行为的上下文表示（即 $[\mathbf{x}_{u,l}^e; \mathbf{x}_{u,l}^f; \mathbf{x}_{u,l}^c; \mathbf{x}_{u,l}^p]$ ）堆叠起来作为注意力机制的键（key）和值

(value), 同时利用用户在下一步的行为 $\mathbf{b}_{u,l+1}$ 的嵌入作为查询 (query), 自适应地提取与待预测的下一个物品更相关的上下文,

$$\mathbf{Q}'_l = \mathbf{b}_{u,l+1} \mathbf{W}_{Q'}, \quad (5)$$

$$\mathbf{K}'_l = [\mathbf{x}_{u,l}^e; \mathbf{x}_{u,l}^f; \mathbf{x}_{u,l}^c; \mathbf{x}_{u,l}^p] \mathbf{W}_{K'}, \quad (6)$$

$$\mathbf{V}'_l = [\mathbf{x}_{u,l}^e; \mathbf{x}_{u,l}^f; \mathbf{x}_{u,l}^c; \mathbf{x}_{u,l}^p] \mathbf{W}_{V'}, \quad (7)$$

$$\mathbf{g}_{u,l} = \text{softmax}\left(\frac{\mathbf{Q}'_l \mathbf{K}'_l{}^T}{\sqrt{d}}\right) \mathbf{V}'_l, \quad (8)$$

其中 $\mathbf{Q}'_l$ ,  $\mathbf{K}'_l$ 和 $\mathbf{V}'_l$ 是查询、键和值的线性变换,  $\mathbf{W}_{Q'}$ ,  $\mathbf{W}_{K'}$ ,  $\mathbf{W}_{V'} \in \mathbb{R}^{d \times d}$ 是可学习的权重矩阵,  $\mathbf{g}_{u,l}$ 是用户 $u$ 在时间步 $l$ 的考虑了目标行为的门控上下文表征。

接着, 该模型根据下一交互物品的行为类型\*选出一个下一行为的上下文 $\mathbf{X}_u^{*(K)}$ , 将其与门控上下文表征 $\mathbf{g}_{u,l}$ 进行平均池化,

$$\mathbf{g}'_{u,l} = \frac{\mathbf{x}_{u,l}^{*(K)} + \mathbf{g}_{u,l}}{2}, \quad (9)$$

最后, 该模型将混合行为上下文特征和目标行为导向的特征进行融合, 并与物品 $i$ 的原始表征向量 $\mathbf{i}_i$ 进行内积操作, 得到用户 $u$ 在时间步 $l+1$ 对物品 $i$ 的预测得分,

$$\hat{r}_{u,l+1,i} = (\mathbf{x}_{u,l} + \mathbf{g}'_{u,l})(\mathbf{i}_i)^T, \quad (10)$$

任务一采用与SASRec [11]相同的二元交叉熵损失函数,

$$\mathcal{L}_1 = - \sum_{u \in \mathcal{U}} \sum_{l=2}^{L+1} \delta(i_u^l) [\log(\sigma(\hat{r}_{u,l,i_u^l})) + \log(1 - \sigma(\hat{r}_{u,l,neg}))], \quad (11)$$

其中 $neg$ 是在 $\mathcal{I} \setminus \mathcal{S}_u$ 中随机采样的负样本。当 $i_u^l$ 不是一个填充项时,  $\delta(i_u^l) = 1$ , 否则为0。

### 3.4 任务二：购买行为预测

任务二主要考虑对用户以购买为导向的长期偏好的建模, 用户的购买行为在其交互历史中通常非常稀疏, 这使得模型难以有效地学习用户的购买偏好。为了解决上述问题, 该模块以自监督的方式学习用户购买导向的偏好。例如, 用户在做出购买决策之前, 通常会通过一些辅助行为对比不同品牌, 如果用户最终选择购买某个物品而不是其他交互过的物品, 这说明相较于之前其他交互过的物品, 用户更喜欢购买的这个物品。

#### 3.4.1 行为感知的自注意力机制

用户交互历史中行为转移模式所反映的意图通常是多样且不确定的, 例如, 用户可能在购买智能手表之前浏览了书籍, 因此假设用户更喜欢智能手表而不是这本书可能是不合理的。为了解决这一不确定性问题, 该模块设计了一种行为感知的自注意力机制, 将某种行为



的历史交互物品集合聚合成一个虚拟物品。首先，该模型创建了一个行为感知的因果注意力掩码矩阵，

$$\mathbf{A} = \mathbf{M} \otimes \Delta \quad (12)$$

其中 $\mathbf{M} \in \mathbb{R}^{L \times L}$ 是一个行为感知的注意力掩码矩阵，当 $b_u^l = b_u^{l'}$ 并且 $l-l' < L_{clip}$ 时， $\mathbf{M}_{l,l'} = 1$ ，否则为0。 $L_{clip}$ 用于保证聚合的物品与用户的购买意图相关，避免久远的交互记录对当前的购买行为产生干扰，接着使用注意力机制将用户的意图融入到虚拟物品当中，

$$\mathbf{N}_u = (\text{softmax}(\frac{\mathbf{Q}_u'' \mathbf{K}_u''^T}{\sqrt{d}}) \otimes \mathbf{A}) \mathbf{V}_u'' \quad (13)$$

其中 $\mathbf{Q}_u'' = \mathbf{S}_u \mathbf{W}_{Q''}$ ， $\mathbf{K}_u'' = \mathbf{S}_u \mathbf{W}_{K''}$ 和 $\mathbf{V}_u'' = \mathbf{S}_u \mathbf{W}_{V''}$ 是注意力机制中查询、键和值的线性变换， $\mathbf{W}_{Q''}, \mathbf{W}_{K''}, \mathbf{W}_{V''} \in \mathbb{R}^{d \times d}$ 是可学习的权重矩阵。该模块可以得到一条虚拟物品嵌入向量序列，即嵌入矩阵 $\mathbf{N}_u = [\mathbf{n}_{u,1}, \mathbf{n}_{u,2}, \dots, \mathbf{n}_{u,L}]$ ，其中 $\mathbf{n}_{u,l}$ 表示用户 $u$ 在时间步 $l$ 的虚拟物品表征。

然后，该模块将购买物品之前的辅助行为虚拟物品视为负样本，将当前时间步的购买物品作为正样本，用于学习用户购买导向的偏好。该模型采用对比损失函数来训练模型，如下所示，

$$\text{sim}(\mathbf{u}_u, \mathbf{i}_y) = \exp(\frac{\mathbf{u}_u \mathbf{W} \mathbf{i}_y^T}{\rho}) \quad (14)$$

$$\text{sim}(\mathbf{u}_u, \mathbf{n}_{u,t}) = \exp(\frac{\mathbf{u}_u \mathbf{W} \mathbf{n}_{u,t}^T}{\rho}) \quad (15)$$

$$\mathcal{L}_2 = - \sum_{u \in \mathcal{U}} \sum_{y \in \mathcal{Y}_u} \log \frac{\text{sim}(\mathbf{u}_u, \mathbf{i}_y)}{\text{sim}(\mathbf{u}_u, \mathbf{i}_y) + \sum_{t \in \mathcal{T}(u,y)} \text{sim}(\mathbf{u}_u, \mathbf{n}_{u,t})} \quad (16)$$

其中 $\text{sim}$ 用于计算用户和物品的相关度， $\mathbf{W}$ 用于对用户和物品的表征进行线性变换， $\rho$ 是一个温度系数。 $y \in \mathcal{Y}_u$ 是用户 $u$ 购买过的物品集合， $\mathcal{T}(u,y)$ 是用户 $u$ 与购买物品 $y$ 时间步相近的辅助行为交互物品的集合。

### 3.5 模型训练和预测

该模型使用端到端的训练方法同时最小化**任务一：下一物品预测**的损失函数 $\mathcal{L}_1$ 和**任务二：购买行为预测**的损失函数 $\mathcal{L}_2$ ，即

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2. \quad (17)$$

最后，该模型可以预测用户 $u$ 在第 $l+1$ 个时间步对物品 $i$ 的评分，

$$\hat{o}_{u,l+1,i} = \mathbf{z}_{u,l}(\mathbf{i}_i)^T, \quad (18)$$

其中 $\mathbf{z}_{u,l} = \mathbf{x}_{u,l} + \mathbf{g}_{u,l}' + \mathbf{u}_u$ ， $\mathbf{x}_{u,l}$ 和 $\mathbf{g}_{u,l}'$ 在任务一 $\mathcal{L}_1$ 中进行优化，而是 $\mathbf{u}_u$ 在任务二 $\mathcal{L}_2$ 中进行优化。

## 4 复现细节

### 4.1 与已有开源代码对比

本报告实现的NextIP模型是基于作者提供的代码进行复现的，完成了模型的性能评估、模块消融、行为消融三个实验任务，并且可以达到原论文报告的实验结果。

本文在原模型的基础上，考虑到多行为序列推荐任务的目标是预测下一购买物品，因此在任务一中加入了一个行为感知的加权二元交叉熵损失函数，以更好地学习用户的购买偏好。在原论文使用的两个公开数据集上进行对比实验，均有效提升了模型的推荐性能，具体的改进细节见[创新点——行为感知的二元交叉熵损失函数](#)。

### 4.2 数据集和评估方式

本报告采用原论文使用的两个公开数据集Tmall和UB进行实验，这两个数据集是电子商务领域的真实数据集，包含了用户的多种行为类型，如浏览、收藏、加购和购买。每个数据集预处理之后的统计信息如表1所示。与原论文一致，本报告使用留一法划分数据集，进行多次实验。

在这两个数据集上，本文使用与原论文一致的评估指标， $\text{top-k}$ 的命中率（HR@k）和归一化折损累积（NDCG@k）。

表 1. 数据集统计信息

数据集	用户数	物品数	平均序列长度	行为类型
Tmall	17209	16139	43.49	{浏览, 收藏, 购买}
UB	20858	30617	28.65	{浏览, 收藏, 加购, 购买}

### 4.3 模型实现的参数细节

与原论文一致，本文在实验中将序列长度固定为50，对原始序列长度过长或者不足的部分进行截断和填充。一批次样本的数量（batch size）为128，随机丢失概率（dropout rate）为0.5，学习率（learning rate）为0.001，优化器采用Adam，任务二行为感知的自注意力块中裁剪大小 $L_{clip}$ 设置为10，对比学习温度系数 $\rho$ 设置为0.07，序列编码器的自注意力层数在 $K \in \{1, 2, 3\}$ 中选择。

### 4.4 创新点——行为感知的二元交叉熵损失函数

考虑到多行为序列推荐任务的目标是预测下一购买物品，本文在任务一中加入了一个行为感知的加权二元交叉熵损失函数，从而解决不同输入行为的重要性 [3]。具体地，首先将行为类型通过独热码的方式进行编码，

$$\begin{aligned} \mathbf{b}_{hot}^p &= [1, 0, 0, 0], \mathbf{b}_{hot}^c = [0, 1, 0, 0], \\ \mathbf{b}_{hot}^f &= [0, 0, 1, 0], \mathbf{b}_{hot}^e = [0, 0, 0, 1]. \end{aligned} \tag{19}$$

接下来将用户的行为类型编码映射为一个行为感知的权重向量，使用权重调整的方式缓解购买行为的稀疏性问题，

$$w^* = \mathbf{b}_{\text{hot}}^* \mathbf{W}^*, \quad (20)$$

其中 $*$ 表示目标物品对应的行为类型，即目标行为，最后为二元交叉熵损失函数的正项加以权重 $w^*$ ，即可得到行为感知的二元交叉熵损失函数，

$$\mathcal{L}_1^* = - \sum_{u \in \mathcal{U}} \sum_{l=2}^{L+1} \delta(i_u^l) [w^* \log(\sigma(\hat{r}_{u,l,i_u^l})) + \log(1 - \sigma(\hat{r}_{u,l,neg}))], \quad (21)$$

在这一部分的实验中，本文尝试两种映射矩阵 $\mathbf{W}^*$ ，一种为可学习的权重矩阵，另一种为静态的权重矩阵，其中辅助行为（即浏览、收藏和加购）的权重设置为0.1，目标行为（即购买）的权重在[0.25, 0.5, 0.75, 1.0]中选择。

## 5 实验结果分析

### 5.1 模块消融与性能比较

本文进行了三部分实验，在第一部分实验中，本文对NextIP模型的两个模块进行了消融实验，验证了TBCG模块和BSA模块的有效性，并尝试在模型中加入行为感知的二元交叉熵损失函数，测试是否有性能的提升。在第二部分实验中，本文分析了行为感知的二元交叉熵损失函数对模型性能的影响。第三部分实验，我分析了不同的行为感知权重对模型性能的影响程度及稳定性。实验结果如表2所示。

表 2. 模块消融实验与性能比较

Model	Tmall		UB	
	HR@10	NDCG@10	HR@10	NDCG@10
NextIP(w/o BSA)	0.664387	0.43284	0.626485	0.423454
NextIP(w/o TBCG)	0.623812	0.409364	0.571316	0.392285
NextIP	<u>0.678656</u>	<u>0.443626</u>	<u>0.630485</u>	<u>0.423612</u>
NextIP*( $\mathbf{W}_d^*$ )	0.623644	0.409101	0.528155	0.364891
NextIP*( $\mathbf{W}_s^*$ )	<b>0.687736</b>	<b>0.455135</b>	<b>0.636456</b>	<b>0.435008</b>

从表2中可以得到以下结论：

- 第一部分的消融实验中，可以发现在两个数据集中，TBCG模块和BSA模块都对模型的性能有一定的提升，并且TBCG模块的提升效果更为明显，说明下一物品预测任务在多行为序列推荐任务中的起着更大的作用。



- 第二部分的改进实验中，可以发现在两个数据集中，静态权重对模型性能有所提升，而使用可学习的动态参数会出现过拟合现象，导致模型推荐性能下降。

## 5.2 行为类型消融实验

为了研究不同特定行为的交互物品序列对NextIP性能的影响，本文进行了额外的行为消融实验。通过去除任务一中的某一类特定行为的项目序列，可以得到NextIP的四种变体，即-e, -p, -c和-f，分别表示在模型中去除了浏览、购买、加购和收藏行为。需要注意Tmall数据集中不包括收藏行为，实验结果如图2所示，实验根据消除了特定行为的模型性能与原模型性能的性能差异，计算每种行为的贡献度。

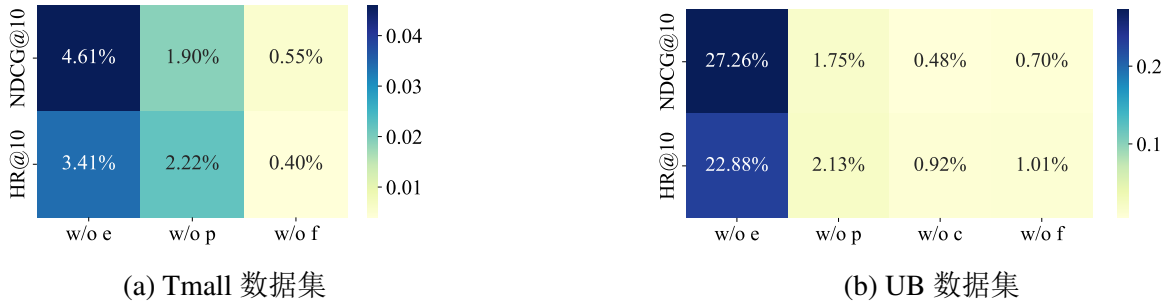


图 2. 不同行为对模型性能的影响

从图2中可以得到以下结论：

- 所有特定行为的序列对NextIP的性能都有一定的贡献，其中Tmall数据集中加购和收藏行为对模型的性能影响较小，而UB数据集中购买行为对模型的性能影响也较小。
- 由于浏览行为是用户最常见的行为，在数据集中浏览行为往往比较稠密，所以将浏览行为从模型中去除会对模型的性能产生较大的影响。
- 由于购买行为直接地反映了用户的购买意图，而多行为序列推荐任务的目标是预测下一个购买物品，因此Tmall数据集中购买行为对模型的性能影响较大，而UB数据集中由于购买行为相对稀疏，购买行为的贡献相对较小。

## 5.3 行为感知权重对模型性能的影响

为了探究不同的行为感知权重对模型性能的影响，本文在两个数据集上进行了实验，将辅助行为的权重设置为0.1，目标行为的权重在 $w^p \in \{0.25, 0.5, 0.75, 1.0\}$ 中选择，进行了多次实验取平均值的结果如图3所示，可以看出在两个数据集中，权重为0.75时模型性能最好，权重设置为0.25和0.5时模型性能低于原模型。

从图中也能看出实验结果的波动性较低，权重为0.75时多次实验结果中模型性能的最小值也高于原始模型的性能。另外，各种权重下的多次实验结果的波动性均较小。因此该实验说明行为感知的二元交叉熵损失函数能有效提升模型的推荐性能。

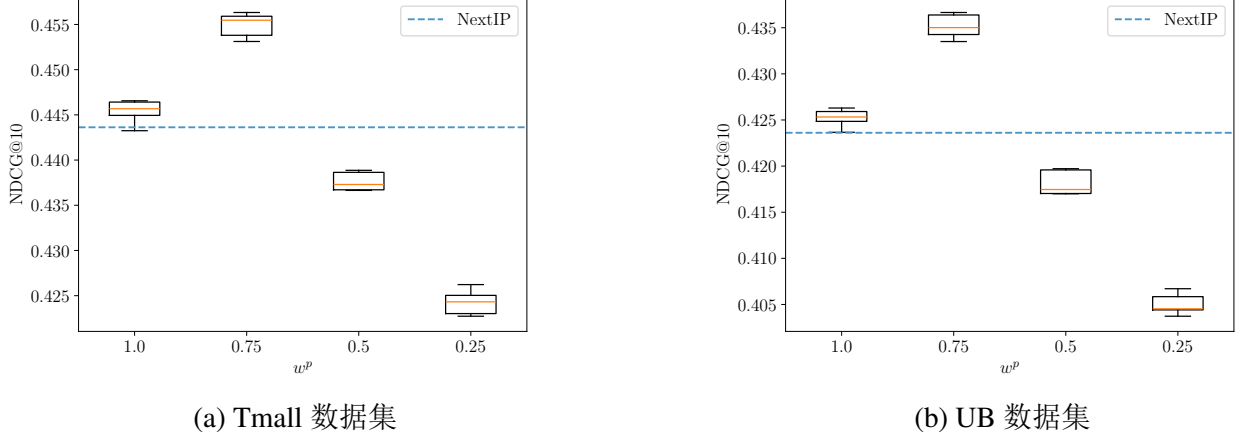


图 3. 行为感知权重对模型性能的影响

## 6 总结与展望

本报告复现了多行为序列推荐模型NextIP，该模型同时考虑两个任务进行建模。在任务一中，模型使用了目标行为感知的上下文聚合器（TBCG），聚合不同行为的上下文信息。在任务二中，模型引入行为感知的自注意力机制（BSA），将特定行为的历史交互物品抽象成一个虚拟物品，以通过自监督的方式学习用户的购买导向偏好。

在实验部分，本报告进行了消融实验和性能比较，验证了TBCG模块和BSA模块的有效性。实验结果表明，这两个模块在提高模型性能方面起到了重要作用，尤其是TBCG模块在多行为序列推荐任务中表现尤为突出。本文还进行了行为消融实验，分析每种行为类型对模型性能的影响。此外，本文引入了一个行为感知的二元交叉熵损失函数，结果显示静态权重可以稳定提升模型的推荐性能。

在后续的研究中，可以考虑使用两个方式对模型进行进一步的改进：（1）使用购买行为划分混合行为序列得到多条连续的混合子序列，深入分析用户购买行为模式；（2）引入物品的属性信息，同时使用物品ID和属性信息，更好地表示物品的特征，提高模型的推荐性能。

## 参考文献

- [1] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *Workshop on Syntax, Semantics and Structure in Statistical Translation at the Conference on Empirical Methods in Natural Language Processing*, 2014.
- [2] Jeffrey L. Elman. Finding structure in time. *Cognitive Science*, 14:179–211, 1990.
- [3] Shereen Elsayed, Ahmed Rashed, and Lars Schmidt-Thieme. Multi-behavioral sequential recommendation. In *Proceedings of the 18th ACM Conference on Recommender Systems, RecSys '24*, page 902–906, New York, NY, USA, 2024.

- [4] Chen Gao, Xiangnan He, Dahua Gan, Xiangning Chen, Fuli Feng, Yong Li, Tat-Seng Chua, and Depeng Jin. Neural multi-task recommendation from multi-behavior data. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, pages 1554–1557, 2019.
- [5] Yulong Gu, Zhuoye Ding, Shuaiqiang Wang, Lixin Zou, Yiding Liu, and Dawei Yin. Deep multifaceted transformers for multi-objective ranking in large-scale e-commerce recommender systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management, CIKM '20*, page 2493–2500, New York, NY, USA, 2020.
- [6] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web, WWW '17*, page 173–182, Republic and Canton of Geneva, CHE, 2017.
- [7] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *CoRR*, abs/1511.06939, 2015.
- [8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- [9] Bowen Jin, Chen Gao, Xiangnan He, Depeng Jin, and Yong Li. Multi-behavior recommendation with graph convolutional networks. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '20*, page 659–668, New York, NY, USA, 2020.
- [10] Santosh Kabbur, Xia Ning, and George Karypis. Fism: factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13*, page 659–667, New York, NY, USA, 2013.
- [11] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 197–206, 2018.
- [12] Krishnaram Kenthapadi, Benjamin Le, and Ganesh Venkataraman. Personalized job recommendation system at linkedin: Practical challenges and lessons learned. In *Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys '17*, page 346–347, New York, NY, USA, 2017.
- [13] Thomas Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *ArXiv*, abs/1609.02907, 2016.
- [14] Jing Lin, Weike Pan, and Zhong Ming. Fissa: Fusing item similarity models with self-attention networks for sequential recommendation. In *Proceedings of the 14th ACM Conference on Recommender Systems, RecSys '20*, page 130–139, New York, NY, USA, 2020.
- [15] Qiang Liu, Shu Wu, and Liang Wang. Multi-behavioral sequential prediction with recurrent log-bilinear model. *IEEE Trans. on Knowl. and Data Eng.*, 29(6):1254–1267, June 2017.

- [16] Jinwei Luo, Mingkai He, Xiaolin Lin, Weike Pan, and Zhong Ming. Dual-task learning for multi-behavior sequential recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, page 1379–1388, New York, NY, USA, 2022.
- [17] Wanqi Ma, Xiancong Chen, Weike Pan, and Zhong Ming. Vae++: Variational autoencoder for heterogeneous one-class collaborative filtering. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM '22, page 666–674, New York, NY, USA, 2022.
- [18] Wenjing Meng, Deqing Yang, and Yanghua Xiao. Incorporating user micro-behaviors and item knowledge into multi-task learning for session-based recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1091–1100, New York, NY, USA, 2020.
- [19] Weike Pan, Hao Zhong, Congfu Xu, and Zhong Ming. Adaptive bayesian personalized ranking for heterogeneous implicit feedbacks. *Know.-Based Syst.*, 73(1):173–180, January 2015.
- [20] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, page 811–820, New York, NY, USA, 2010.
- [21] Brent Smith and Greg Linden. Two decades of recommender systems at amazon.com. *IEEE Internet Computing*, 21(3):12–18, 2017.
- [22] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 1441–1450, New York, NY, USA, 2019.
- [23] Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, WSDM '18, page 565–573, New York, NY, USA, 2018.
- [24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017.
- [25] Chao Wang, Hengshu Zhu, Chen Zhu, Xi Zhang, Enhong Chen, and Hui Xiong. Personalized employee training course recommendation with career development awareness. In *Proceedings of The Web Conference 2020*, WWW '20, page 1648–1659, New York, NY, USA, 2020.
- [26] Wen Wang, Wei Zhang, Shukai Liu, Qi Liu, Bo Zhang, Leyu Lin, and Hongyuan Zha. Beyond clicks: Modeling multi-relational item graph for session-based target behavior prediction. In

*Proceedings of The Web Conference 2020*, WWW '20, page 3056–3062, New York, NY, USA, 2020.

- [27] Lianghao Xia, Yong Xu, Chao Huang, Peng Dai, and Liefeng Bo. Graph meta network for multi-behavior recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 757–766, New York, NY, USA, 2021.
- [28] Meizi Zhou, Zhuoye Ding, Jiliang Tang, and Dawei Yin. Micro behaviors: A new perspective in e-commerce recommender systems. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, WSDM '18, page 727–735, New York, NY, USA, 2018.