基于压缩交互图的多行为推荐框架

Wei Guo*
guowei67
@huawei.com 华为诺亚方舟实验室 中国深圳

孟畅*†
mengc21@mails.tsinghua.edu.cn 清华大学深圳国际研究生院，清华大学
中国深圳

袁恩铭†
yem19@mails.tsinghua.edu.cn
跨学科研究院
清华大学信息科学系
大学
中国北京

何志成
hezhicheng9@huawei.com
华为诺亚方舟实验室
深圳 ，中国

Huifeng Guo
huifeng.guo@huawei.com
华为诺亚方舟实验室
深圳 ，中国

Yingxue Zhang
yingxue.zhang@huawei.com
华为加拿大技术公司
蒙特利尔，加拿大

Bo Chen
chenbo116@huawei.com
华为诺亚方舟实验室
深圳 ，中国

Yaochen Hu
yaochen.hu@huawei.com
华为加拿大技术公司
蒙特利尔，加拿大

Ruiming Tang‡
tangruiming@huawei.com
华为诺亚方舟实验室
深圳 ，中国

Xiu Li‡
li.xiu@sz.tsinghua.edu.cn 清华大学深圳国际研究生院，清华大学，深圳，中国

Rui Zhang
rayteam@yeah.net
ruizhang.info 深圳，中国

arXiv:2303.02418v1 [cs.IR] 4 Mar 2023

摘要

在大多数真实世界的推荐场景中，都会记录多种类型的用户行为数据（例如点击、加入购物车和购买），这些数据有助于学习用户的多方面偏好。然而，由于数据分布不平衡且目标行为稀疏，探索多行为数据颇具挑战：将多行为数据视为“特征”时，会导致高阶关系建模不足；将多行为数据视为“标签”时，则会在多任务学习中引发梯度冲突。本文提出 CIGF，即一种基于压缩交互图的框架，以克服上述局限。具体而言，我们设计了一种新颖的压缩交互图卷积网络（CIGCN），以显式建模实例级 高阶关系。为缓解将多行为数据视为“标签”时可能出现的梯度冲突，我们在 CIGCN 之上提出一种用于多任务学习的分离输入的多专家（MESI）网络，并采用分离输入 。在三个大规模真实世界数据集上的全面实验表明，CIGF 具有优越性。消融研究和深入分析进一步验证了我们的模型在捕捉高阶关系和缓解梯度冲突方面的有效性。源代码和数据集可在 https://github.com/MC-CV/CIGF 获取。

CCS 概念

• **信息系统** →**推荐系统**。

关键词

多行为推荐、交互图、多任务

ACM 参考文献格式:

WeiGuo、孟畅、袁恩铭、何志成、Huifeng Guo、Yingxue Zhang、Bo Chen、Yaochen Hu、Ruiming Tang、Xiu Li 和 Rui Zhang。2023。基于压缩交互图的多行为推荐框架。载于*2023*年 *ACM Web* 会议论文集（*WWW '23*），*2023*年*5*月*1*日至*5*日，美国得克萨斯州奥斯汀。ACM，美国纽约州纽约市，12页。https://doi.org/10.1145/3543507.3583312

*两位作者对本研究的贡献相同。
†此项工作完成于他们在华为诺亚方舟实验室担任研究实习生期间。
‡通讯作者。

本作品采用知识共享署名国际 4.0 许可协议授权。

*WWW '23, 2023*年*5*月*1*日至*5*日, 美国得克萨斯州奥斯汀©2023 版权所有归所有者/作者所有。
ACM 国际标准书号 978-1-4503-9416-1/23/04.
https://doi.org/10.1145/3543507.3583312

∗ 表示共同第一作者，贡献相同。
† 他们在华为诺亚方舟实验室担任研究实习生期间完成的工作。
‡ 表示共同通讯作者。

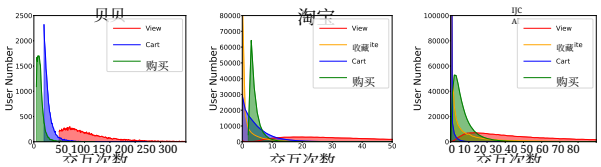

图 2：不同交互数量下各行为的用户数量直方图。

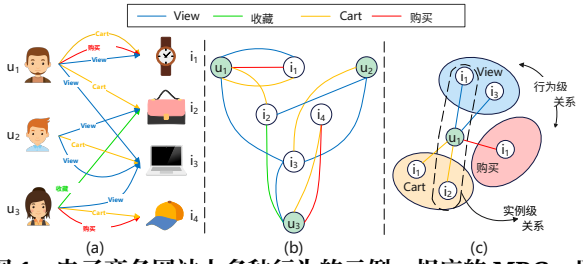

图 1：电子商务网站上多种行为的示例、相应的 MBG，以及行为级关系与实例级关系的比较。

1 引言

推荐系统（RS）是满足个性化信息需求的重要工具。为了预测用户对物品的偏好，众多方法致力于协同过滤（CF） [29] 技术，从用户和物品的历史交互中学习用户表示和物品表示，并基于这些表示进行预测。大多数 CF 方法 [14, 15, 26, 34, 38] 针对单一类型的行为进行设计，很少考虑用户的多方面偏好，而这类偏好广泛存在于真实世界的网络应用中。以图1(a)所示的电子商务网站为例，用户通过不同的行为与物品进行交互，例如浏览、加入购物车、标记为收藏和购买。由于不同类型的行为体现了用户多样兴趣所产生的不同交互模式，因此，显式利用多行为数据进行推荐至关重要。

NMTR [10],、DIPN [12], 和MATN [36] 将多种行为视为不同类型，并采用神经协同过滤单元、注意力算子和Transformer架构来建模它们之间的依赖关系，其效果远优于将它们视为同一类型。多行为数据可以表示为多路二部图（MBG），如图 1(b) 所示。近年来，得益于图神经网络（GNN）模型 [7, 14, 25, 34] 在表示关系信息以及建模用户与物品之间携带协同信号的高阶关系方面的能力，基于图神经网络的模型已成为推荐领域的热门方法。例如，MBGCN [19],、GHCF [6],和MB-GMN [37]进一步结合多图、非采样和元网络，增强图神经网络在多路二部图上捕获高阶协同信号的能力。

多行为数据既可以被视为多行为关系学习的“特征”，也可以被视为多任务监督学习的“标签”。尽管经过多年的研究，仍然存在两个挑战：

• **数据分布不均衡**。 如图 2 所示，不同用户和不同行为的观测交互高度不均衡，少数用户和行为涵盖了大部分交互。

• **稀疏目标行为** （待预测的行为，例如电子商务中的购买）。我们还发现，大多数用户的购买记录少于 10 条；与包含数千至数百万个物品的整个物品空间相比，这些记录极为稀疏。

我们深入探究了这些挑战，并观察到以下局限性：

• **将多行为数据视为“特征”时，对高阶关系的建模不足**。 特征”。用户 - 物品关系对于揭示促使用户偏好物品的潜在原因具有重要意义。例如，如图1(a)所示， u_1 与 i_4 之间存在若干三阶关系（例如，$u_1 \xrightarrow{cart} i_2 \xrightarrow{be\ favored\ by} u_3 \xrightarrow{purchase} i_4$）。借助协同效应，我们预测 u_1 可能会购买 i_4，因为作为与 u_1相似的 u_3，该用户此前购买过 i_4。现有方法，如MBGCN、GHCF和MB-GMN，已尝试利用图神经网络来融合高阶关系。然而，它们采用两阶段范式：首先，考虑属于某一行为的所有历史记录，为每种行为学习表示；随后，利用学习到的表示对不同行为之间的高阶关系进行建模。我们认为，如图1(c)所示，这种关系建模方式是行为级的。由于不平衡数据分布，学习到的关系很容易偏向高阶用户和行为，从而使学习到的表示无法有效捕捉高阶关系。

• **将多行为数据视为“标签”时可能出现梯度冲突**。早期工作，如 MBGCN [19] 和 MATN [36]，仅使用目标行为作为标签来训练模型，因此容易受到目标行为稀疏所导致的稀疏性问题影响。为缓解这一问题，采用多任务学习（MTL）技术，将辅助行为用作标签是一种可行的方法。然而，由于负迁移[1] [32] 现象，使用多个目标进行训练并不容易。负迁移是指知识在不同任务间迁移时，模型性能出现下降。因此，将多行为数据视为“标签”存在风险。近期的一些工作，如 NMTR [10], 、GHCF[6], 和 MB-GMN [37]，研究了多行为推荐中的多任务学习。由于这些方法使用相同的输入，可能会因梯度耦合问题而遭遇梯度冲突。详细说明见第3.3节。

为解决上述局限性，我们提出一种新颖的基于压缩交互图的框架（CIGF），以更好地学习用户和物品的表示。针对将多行为数据“作为特征”处理时高阶关系建模不足的问题，我们设计了压缩交互图卷积网络（CIGCN），以显式建模高阶关系。CIGCN 首先利用矩阵乘法作为交互算子，生成显式编码实例级高阶关系（包括用户-用户、用户-物品和物品-物品关系）的高阶交互图；然后利用节点级注意力机制选取最有用的高阶交互图，并压缩图空间。最后，结合先进的图卷积网络模型和残差连接

[1]此处我们忽略跷跷板现象（即多任务学习模型提升某些任务的性能，却牺牲其他任务的性能） [31]，因为多行为推荐的目标是预测目标行为。

应用于这些图，以探索高阶图信息，并缓解表示学习中的过平滑问题。

为缓解将多行为数据“视为标签”时可能产生的梯度冲突，我们在 CIGCN 之上提出了一种用于多任务学习的分离输入的多专家（MESI）网络。MESI 网络是一种层次化神经架构，与 MMOE [24] 和 PLE [31]类似。不过，我们引入了独立输入，取代原始 MMOE 和 PLE 多任务学习模型中使用的相同输入。具体而言，我们使用始于不同类型行为的关系来学习独立输入。通过显式使用独立输入来学习任务感知信息，可以缓解相同输入在不同任务间传递知识时可能产生的梯度冲突，从而使学习过程更加稳定有效。关于 MESI 梯度解耦的解释，请参阅第3.3节。

综上所述，我们的工作做出了以下贡献：

• 我们从全新的视角审视多行为推荐问题，通过数据分析和理论支持，将多行为数据视为“特征”和“标签”。

• 我们提出了一种新颖的压缩交互图框架（CIGF），该框架由压缩交互图卷积网络（CIGCN）和分离输入的多专家（MESI）网络组成。在将多行为数据视为“特征”时，CIGCN 旨在通过显式图交互进行实例级高阶关系建模。在将多行为数据视为“标签”时，MESI 旨在通过分离输入缓解潜在的梯度冲突。

• 我们在三个真实世界数据集上开展了大量实验，以验证所提出 CIGF 框架的有效性。消融分析和深入分析进一步验证了 CIGCN 和 MESI 的有效性与合理性。此外，我们还进一步分析了该方法的复杂度，并在附录A.6中开展了详细的效率实验。

2 相关工作

用于推荐的图神经网络。 基于图神经网络的方法可以通过将多行为数据视为“特征”来处理。大多数现有图神经网络都是针对同质图提出的，例如 NGCF[34], LR-GCCF [7], 和 LightGCN [14], ，这些方法忽略了多种边类型。近年来，一些研究者开始关注异质图，并提出了 HGNN [39], R-GCN [27], 和 HGAT [23]等方法。然而，这些方法仅通过利用行为级表示来建模关系，从而考虑行为级关系。基于超图的方法[3, 9] 利用超图对复杂的高阶关系进行建模。然而，由于超图中的一条边连接两个或更多节点，因此不适用于多行为场景，因为在这种场景中，一对节点之间可能连接多条边。现有的基于元路径的方法，如 Metapath2vec [8], MCRec [16], 和 HAN [35] ，使用人工选择的元路径对高阶关系进行建模，但这类方法受限于对专家知识的需求，以及难以搜索涵盖任意长度和边类型的所有有用元路径。

用于推荐的多任务学习。多任务学习方法可以将多行为数据视为“标签”来处理。广泛使用的一种模型是图3(d)中的共享底层结构。尽管它有助于多个任务间共享知识，但由于任务差异，仍面临任务冲突的风险。为应对任务差异，一些研究采用注意力网络进行信息融合。图3(e)中的 MMOE [24]扩展了 MOE [18]，通过使用不同的门控网络，在多任务学习中获得不同的融合权重。图3(f)中的 PLE [31]进一步提出在底层利用共享或任务特定专家网络，然后采用门控网络自适应地组合这些专家网络，从而应对任务冲突并缓解负迁移问题。然而，这些方法仍对多任务学习使用相同的输入。我们认为，这种方式可能会因梯度耦合问题而产生梯度冲突。详细说明见第3.3节。

多行为推荐。现有的多行为推荐方法可分为两类：基于图的方法和基于多任务学习的方法 [17]。前一类将多行为数据视为“特征”。一些早期工作，如 DIPN [12] 和 MATN [36]，未能捕捉高阶关系，因此表现不佳。大多数近期工作（例如 GHCF [6] 和 MBGCN [19]）采用行为级建模方式，无法捕捉细粒度的实例级多行为关系。另一些方法，如 MBGCN[19] 和 MGNN [40]，直接从 MBG 中学习高阶关系；由于数据分布不平衡，它们难以广泛挖掘有用关系。与上述方法不同，我们提出的 CIGCN 通过显式图交互和图压缩对高阶关系进行建模，因此能够在实例级学习关系。后一类将多行为数据视为“标签”。图 3(a) 中的 NMTR [10] 假设用户的多种类型行为按固定顺序发生，这一假设可能过于严格，无法适用于所有用户。图 3(b) 中的 GHCF 采用了与多任务学习共享底层类似的架构。二者唯一的区别在于，GHCF 使用双线性运算（请参阅第3.3节）作为预测头，而共享底层使用神经网络。图 3(c) 中的 MB-GMN [37]进一步提出使用元预测网络来捕捉多任务学习中的复杂跨类型行为依赖。这些现有方法对所有样本使用相同的静态权重来优化多个任务。其最明显的缺点是容易因样本差异而产生冲突风险，因为不同样本对不同任务可能有不同偏好。相比之下，我们提出的 MESI 网络会根据不同样本的特性学习自适应权重。此外，我们利用分离输入来学习任务感知信息，从而缓解潜在的梯度冲突。

3 预备知识

3.1 问题定义

在本节中，我们给出多行为推荐任务的形式化定义。我们分别将用户集合和物品集合记为 $\mathrm{U} = \{u_1, u_2, \ldots\ldots, u_M\}$ 和 $\mathrm{I} = \{i_1, i_2, \ldots\ldots, i_N\}$。行为对应的用户-物品交互矩阵记为 $\mathrm{Y} = \{\mathrm{Y}_1、\mathrm{Y}^2、\ldots\ldots、\mathrm{Y}^K\}$。其中，$M$、 N 和 K 分别表示用户数、物品数和行为类型数； $y_{ui}^k = 1$ 表示用户 u 在行为 k 下与物品 i 发生交互，否则为 $y_{ui}^k = 0$。通常，我们需要优化一种目标行为（例如购买），记为 Y^K；其他行为 $\{\mathrm{Y}_1、\mathrm{Y}^2、\ldots\ldots、\mathrm{Y}^{K-1}\}$（例如浏览和标记为收藏）则作为辅助行为，用于辅助预测

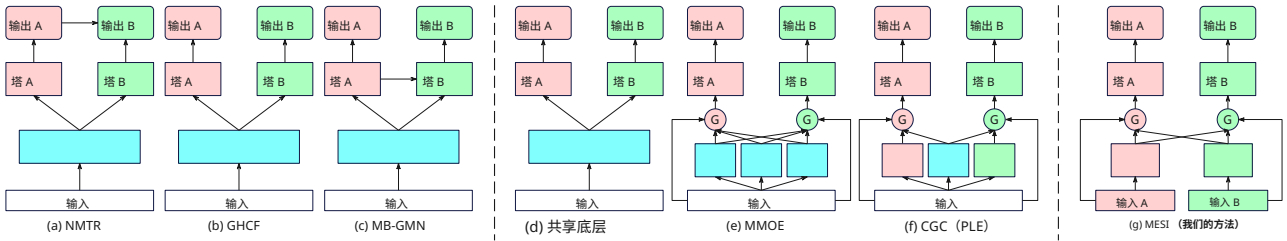

图 3：现有模型与我们提出的 MESI模型的网络结构。蓝色矩形表示共享层，粉色和绿色矩形表示任务专属层，粉色和绿色圆形表示任务专属门。

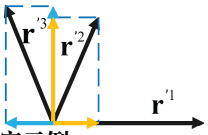

图 4：梯度冲突示例。

目标行为。我们的目标是预测用户 u 在目标行为 K 下与物品 i 发生交互的概率。

3.2 图与关系定义

如图 1(b) 所示，我们将多路二部图（MBG）记为 $G = (V, E, A)$，其中 $V = U \cup I$ 是包含所有用户和物品的节点集，$E = \cup_{r \in \mathcal{R}} E_r$ 是包含用户与物品之间所有行为记录的边集。这里，r 表示一种特定类型的行为，R 是所有可能行为类型的集合。$A = \cup_{r \in \mathcal{R}} A_r$ 是邻接矩阵集，其中 A_r 表示特定行为图 $G_r = (V, E_r, A_r)$ 的邻接矩阵。

关系 P 定义为 MBG G 中形式为 $v_1 \xrightarrow{r_1} v_2 \xrightarrow{r_2} \cdots \xrightarrow{r_l} v_{l+1}$ 的一条路径。我们将 $r_{\mathcal{P}} = \{r_1, r_2, \cdots r_l\}$ 记为该路径中所有边类型的集合。如果 $l \geq 2$ 且 $|r_{\mathcal{P}}| = 1$，则将这条路径定义为高阶单行为关系。如果 $l \geq 2$ 且 $|r_{\mathcal{P}}| \geq 2$，则将这条路径定义为高阶多行为关系。如果存在一条连接节点 v_s 与节点 v_t、长度为 l 的路径，则节点 v_s 是节点 v_t 的第 l 阶可达邻居。在新生成的图 G_l 中，如果任意两个相连节点 v_s 和 v_t 在原始 MBG G 中均为第 l 阶可达，则将 G_l 定义为第 l 阶图。我们将在第 4.2 节通过高阶图交互和卷积，说明如何显式建模高阶多行为关系。

3.3 多任务学习中的耦合梯度问题

如第2节所总结，现有的大多数方法在多任务学习中使用相同的输入。这可能导致多任务学习中的梯度耦合问题，从而限制各任务的学习能力。这里以 GHCF [6] 中的双线性模块为例进行说明。双线性模块可表示为：

$$\hat{o}_{u,i}^k = \mathbf{x}_u^{*T} \cdot diag\left(\mathbf{r}^k\right) \cdot \mathbf{y}_i^* = \sum_j^d (\mathbf{x}_u^* \circ \mathbf{y}_i^* \circ \mathbf{r}^{kT})_j \tag{1}$$

其中（$\circ$）表示哈达玛积运算，$\hat{o}_{u,i}^k$ 表示第 k 种行为的预测值，$\mathbf{x}_u^*$ 和 $\mathbf{y}_i^*$ 分别表示用户 u 和物品 i 的学习到的表示。$\mathbf{r}^k \in \mathbb{R}^{1 \times d}$ 是一个行为感知变换向量，用于将用户和物品的表示投影到多任务学习（MTL）的独立预测头上，而 d 表示嵌入维度。这里以平方损失为例进行优化：

$$\mathcal{L}_{u,i} = \sum_{k=1}^{K} (\hat{o}_{u,i}^k - o_{u,i}^k)^2 \tag{2}$$

其中 $o_{u,i}^k$ 是真实标签。那么有：

$$\frac{\partial \mathcal{L}_{u,i}}{\partial (\mathbf{x}_u^* \circ \mathbf{y}_i^*)} = \sum_{k=1}^{K} a_{u,i}^k \mathbf{r}^k = \sum_{k=1}^{K} \mathbf{r}'^k \tag{3}$$

其中 $a_{u,i}^k$ 是一个标量，$\mathbf{r}'^k$ 是来自第 k 个行为的合成梯度，它决定了相同的输入向量 $\mathbf{x}_u^* \circ \mathbf{y}_i^*$ 的更新幅度和方向。可以看出，所有行为的梯度彼此耦合。图 4 展示了 $K = 3$ 的一个示例。假设 $\mathbf{r}'^1$ 是一个参考向量，我们对所有其他向量进行正交分解。可以发现，其他向量的分量与参考向量的方向并不相同。这表明耦合梯度可能引发梯度冲突。其他方法、损失函数以及 MESI 的解耦梯度的证明可参见附录 A.2 和 A.3。

4 我们提出的方法

下面介绍所提出的 CIGF框架，该框架以端到端的方式将多行为数据同时视为“特征”和“标签”。其架构如图5所示，由三个主要部分组成：i）输入层，将用户和物品参数化为嵌入向量；ii）压缩交互图卷积网络（CIGCN），通过将多行为数据显式视为“特征”，从中提取实例级高阶关系；iii）分离输入的多专家（MESI）网络，通过将多行为数据视为“标签”，利用分离输入从中挖掘多任务监督信号。

4.1 输入

我们首先应用共享嵌入层，将用户和物品的独热编码ID转换为低维稠密嵌入。形式化地，给定一个用户-物品对（u，i），用户 u 和物品 i 的嵌入查找操作可表示如下：

$$\mathbf{x}_u = \mathbf{P}^T \cdot \mathbf{p}_u, \ \mathbf{y}_i = \mathbf{Q}^T \cdot \mathbf{q}_i \tag{4}$$

其中，$\mathbf{p}_u \in \mathbb{R}^{M \times 1}$ 和 $\mathbf{q}_i \in \mathbb{R}$ 分别表示用户 u 和物品 i 的独热编码 ID，$\mathbf{P} \in \mathbb{R}^{M \times d}$ 和 $\mathbf{Q} \in \mathbb{R}^{N \times d}$ 分别表示用户和物品嵌入矩阵，d 表示嵌入维度。

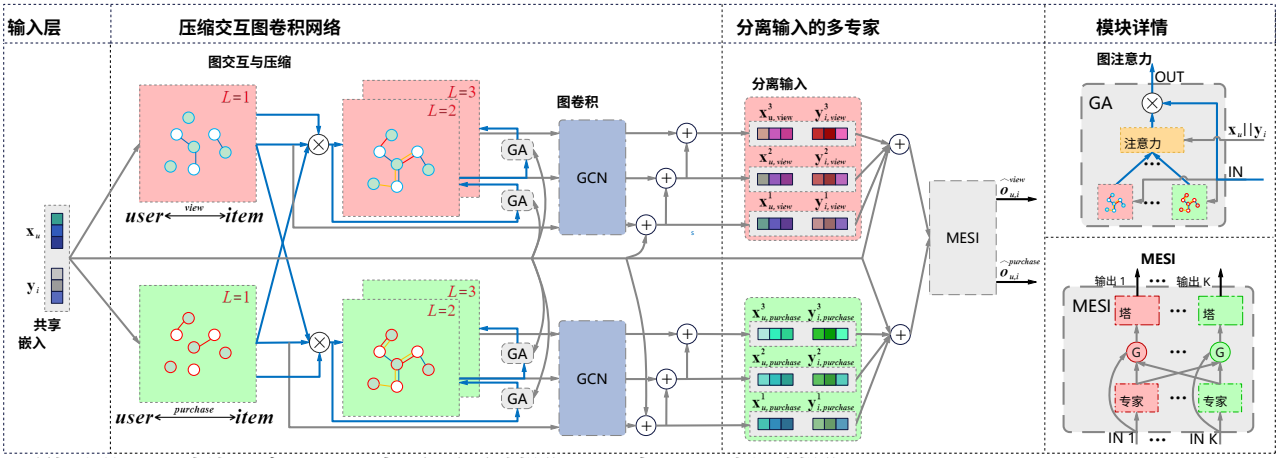

图5：所提出的 CIGF框架示意图。(⊗) 表示矩阵乘法操作，(⊕) 表示逐元素加法操作。

4.2 压缩交互图卷积

4.2.1 图交互。受 DCN [33] 和 xDeepFM [22]成功的启发，它们通过显式特征交叉对高阶特征交互进行建模，我们使用邻接矩阵乘法作为图交互算子，以显式进行实例级高阶关系建模。如图5所示，我们首先将 MBG 划分为若干行为指定图 G_1，$\mathrm{G}^2 \cdots \mathrm{G}^K$。对应的邻接矩阵为$\mathbf{A}_1$，$\mathbf{A}^2 \cdots \mathbf{A}^K$，其可表示为:

$$\mathbf{A}^k = \begin{pmatrix} 0 & \mathbf{Y}^k \\ \left(\mathbf{Y}^k\right)^T & 0 \end{pmatrix} \tag{5}$$

其中，$\mathbf{Y}^k$ 是行为 k的用户-物品交互矩阵。接着，我们利用这些邻接矩阵进行显式的高阶图交互，以编码任意两个节点之间的实例级 关系。将从行为 k 开始的所有可能的 l阶（$1 \le l \le L$）交互图集合记为 B_k^l。之所以在此仅使用从行为 k 开始的交互图，是为了生成具有不同高阶关系的图集，并将其作为独立输入 提供给 MESI，以缓解潜在的梯度冲突。B_k^l 的生成过程可表示为:

$$\mathcal{B}_k^l = \mathcal{B}_k^{l-1} \otimes \{\mathbf{A}^1, \mathbf{A}^2, \cdots, \mathbf{A}^K\} \tag{6}$$

其中，$\mathrm{B}_k^1 = \{\mathbf{A}^k\}$。(⊗) 表示分别对两个集合中任意一对矩阵进行矩阵乘法。注意，共有 K 个高阶图 $\mathrm{B}_k^L (1 \le k \le K)$ 集合，每个集合均以一个行为特定的邻接矩阵$\mathbf{A}^k$为起点。通过在每一步选择不同的行为特定图$\mathbf{A}^k$，我们可以构建一个高阶图集，其中包含多个语义不同的 l阶交互图。具体而言，所有可能的 l 阶图的数量可计算如下:

$$card(\mathcal{B}_k^l) = pow(K, l-1) \tag{7}$$

其中，$card(\cdot)$ 表示集合中元素数量的度量，$pow(K, l-1)$ 是计算给定数字 K 的 $l-1$ 次幂的函数，K 是行为类型的数量。然而，由于所有可能的 l 阶图的数量是一个指数函数关于 $l-1$，使用如此庞大的空间来生成 l 阶交互图并不切实际。

4.2.2 图压缩。为了在有限的时间复杂度和空间复杂度下找到适用的解决方案，我们采用图压缩层，通过节点级多头注意力机制迭代地构建高阶图集。目标节点 v 的图压缩层（节点 v 可以是用户节点 u 或物品节点 i）可表示为:

$$\mathcal{B}_{v,k}^l = \mathcal{B}_{v,k}^{l-1} \otimes \{\alpha_{v,k}^{l,1} \cdot [\mathbf{A}^1, \cdots, \mathbf{A}^K], \cdots, \alpha_{v,k}^{l,H} \cdot [\mathbf{A}^1, \cdots, \mathbf{A}^K]\} \tag{8}$$

其中 $\mathrm{B}_{v,k}^1 = \{\mathbf{A}^k\}$。$(\cdot)$ 表示向量乘法运算，H 表示头的数量，$\alpha_{v,k}^{l,h} \in \mathbb{R}^{1\times K}$ 表示节点 v 在第 l 阶和第 h 个头上的可学习注意力向量。通过采用节点级多头注意力机制，生成的第 l 阶图的数量从 $pow(K, l-1)$ 减少至 $pow(H, l-1)$。由于 H 通常远小于 K，且 $l-1$ 通常是一个很小的值，因此 $pow(H, l-1)$ 的规模是可接受的。注意力机制不仅可用于降低复杂度，还用于寻找对高阶图生成最有用的行为。为了自适应地选择与用户和物品最相关的行为进行表示学习，我们使用节点级注意力机制来获取不同行为的软权重，其定义如下:

$$\alpha_{u,k}^{l,h} = \sigma(\mathbf{W}_k^{l,h}\mathbf{x}_u + \mathbf{b}_k^{l,h}), \alpha_{i,k}^{l,h} = \sigma(\mathbf{W}_k^{l,h}\mathbf{y}_i + \mathbf{b}_k^{l,h}) \tag{9}$$

其中 σ 是激活函数，此处设为 LeakyReLU 以获得更好的性能。$\mathbf{W}_k^{l,h} \in \mathbb{R}^{K\times d}$，而 $\mathbf{b}^{l,h} \in \mathbb{R}^{K\times 1}$ 分别是特征变换矩阵和偏置矩阵。需要注意的是，我们还在此处采用了按行为和层划分的注意力机制（即对不同的层和行为使用不同的变换矩阵），并在第 5.3.1 节中通过实验验证了其有效性。这样，我们就能为用户和物品生成个性化高阶图集合，用于后续的信息传播与整合。

4.2.3 图卷积。通过图交互层和图压缩层生成图集后，我们利用图卷积丰富用户和物品的表示。邻居

信息在每个图中的传播可表示为：

$$\mathbf{x}_{N_u,k}^{l,s} = Agg(\mathbf{x}_u, \mathbf{B}_{u,k}^{l,s}), \mathbf{y}_{N_i,k}^{l,t} = Agg(\mathbf{y}_i, \mathbf{B}_{i,k}^{l,t}) \quad (10)$$

其中，$\mathbf{B}_{u,k}^{l,s}$和$\mathbf{B}_{i,k}^{l,t}$分别表示图集$\mathbf{B}_{u,k}$中第s个和第t个图的邻接矩阵，N_i和u分别表示u和i的邻居，而$\mathbf{x}_{N_u,k}^{l,s}$和$\mathbf{y}_{N_i}^{l,t}$表示从第s个和第t个图聚合邻居信息得到的输出。$Agg(\cdot)$是一个可用于信息聚合的任意图卷积算子。我们使用以下四种最先进的图卷积网络模型来实现$Agg(\cdot)$：GCN聚合器[21]、NGCF聚合器[34]、LR-GCCF聚合器[7]和LightGCN聚合器[14]。请注意，矩阵乘法会产生非常稠密的高阶图，计算上难以接受。因此，我们利用矩阵结合律加速聚合过程，以提高效率。例如，$(\mathbf{A}^k \otimes \mathbf{A}^k \otimes \mathbf{A}^k) \times \mathbf{x}_u$可通过$\mathbf{A}^k \times (\mathbf{A}^k \times (\mathbf{A}^k \times \mathbf{x}_u))$加速计算，其中$(\times)$表示稀疏矩阵与向量之间的乘法。由于$(\times)$将稀疏矩阵和向量合并为一个向量，因此可以有效降低后续乘法的计算复杂度。经过邻居信息传播后，对于每一层中的每个节点u和i，我们都得到$pow(H, l-1)$个邻居表示。为简便起见，我们对这些表示执行求和操作，得到最终的用户和物品表示：

$$\mathbf{x}_{N_u,k}^{l} = \sum_{s=1}^{pow(H,l-1)} \mathbf{x}_{N_u,k}^{l,s}, \mathbf{y}_{N_i,k}^{l} = \sum_{t=1}^{pow(H,l-1)} \mathbf{y}_{N_i,k}^{l,t}. \quad (11)$$

为了更好地探索高阶邻居信息并缓解过平滑问题，我们在图卷积层中引入残差操作，以进行最终的节点信息更新，定义如下：

$$\mathbf{x}_{u,k}^{l} = \mathbf{x}_{N_u,k}^{l} + \mathbf{x}_{u,k}^{l-1}, \mathbf{y}_{i,k}^{l} = \mathbf{y}_{N_i,k}^{l} + \mathbf{y}_{i,k}^{l-1} \quad (12)$$

由于不同层的输出反映了不同阶的关系，我们最终通过如下求和操作将这些输出聚合为一个向量：

$$\mathbf{x}_{u,k}^{*} = \sum_{l=0}^{L} \mathbf{x}_{u,k}^{l}, \mathbf{y}_{i,k}^{*} = \sum_{l=0}^{L} \mathbf{y}_{i,k}^{l} \quad (13)$$

其中$\mathbf{x}_{u,k}^{0} = \mathbf{x}_u$和$\mathbf{y}_{i,k}^{0} = \mathbf{y}_i$分别是用户$u$和物品$i$的初始嵌入。需要注意的是，中心节点直接聚合不同层的邻居信息。与在l层利用$l-1$层的输出递归更新节点嵌入相比，这种方式已被证明有助于解决用户—物品交互图[30]的异质性问题。

4.3 分离输入的多专家

通过CIGCN的设计，我们为每个用户u和每个物品i获得了K表示$\mathbf{x}_{u,k}^{*}$和$\mathbf{y}_{i,k}^{*}$（$1 \le k \le K$），如图5所示。每个表示描述了用户u或物品i对从行为k开始的关系的个性化偏好。为缓解将多行为数据“作为标签”处理时可能产生的梯度冲突，我们在本节提出了一种具有新颖“分离输入”设计的分离输入的多专家（MESI）网络。

现有的多行为方法，如NMTR[10]、GHCF[6]和MB-GMN[37]，对所有样本使用相同的静态权重来优化多个任务，因此会受到样本差异的限制，正如第2节所分析的那样。为了解决这个问题，我们采用了一种层次化神经架构，类似于用于多任务学习的MMOE[24]和PLE[31]。具体而言，我们使用专家网络替代NMTR、GHCF和MB-GMN中采用的共享底层，以学习行为感知信息。在本文中，每个专家定义为$\mathbf{x}_{u,k}^{*}$和$\mathbf{y}_{i,k}^{*}$的组合，可表示为：

$$\mathbf{f}_{u,i}^{k} = \mathbf{x}_{u,k}^{*} \circ \mathbf{y}_{i,k}^{*} \quad (14)$$

其中，$(\circ)$表示哈达玛积运算，$\mathbf{x}_{u,k}$和$\mathbf{y}_{i,k}$是与行为k相关的输入。由于此处使用分离输入来生成专家网络，因此总共可以得到K个专家网络。

由于不同的专家网络可能包含用户的不同偏好或物品的不同属性，因此有必要组合这些专家网络，以便对每个任务进行最终预测。接着，我们使用分离输入为每个任务生成任务感知门控，以自动选择对该任务预测有用的专家子集。任务k的门控可定义为：

$$\mathbf{g}_{u,i}^{k} = Softmax(\mathbf{W}_g(\mathbf{x}_{u,k}^{*} || \mathbf{y}_{i,k}^{*}) + \mathbf{b}_g) \quad (15)$$

其中，$(||)$表示向量拼接运算，$\mathbf{W}_g \in \mathbb{R}^{K \times 2d}$和$\mathbf{b}_g \in \mathbb{R}^{K \times 1}$分别是特征变换矩阵和偏置矩阵，而$\mathbf{g}_{u,i}^{k} \in \mathbb{R}^{K \times 1}$是注意力向量，用作选择器来计算所有专家网络的加权和。任务k的最终预测分数计算如下：

$$\hat{o}_{u,i}^{k} = h^k(\sum_{j=1}^{K} \mathbf{g}_{u,i}^{k}(j) \cdot \mathbf{f}_{u,i}^{j}) \quad (16)$$

其中，$\mathbf{g}_{u,i}^{k}(j)$表示向量$\mathbf{g}_{u,i}^{k}$的第j个元素，$h^k(\cdot)$是塔函数。遵循[18]，为简单起见，我们在此使用平均操作作为塔函数。

4.4 多任务学习的联合优化

由于我们已经获得了每种行为k对应的预测值$\hat{o}_{u,i}^{k}$，因此我们使用<style id='6'>贝叶斯个性化排序（BPR）[26]损失进行多任务学习，其形式化表示如下：

$$\mathcal{L} = -\sum_{k=1}^{K} \sum_{(u,s,t) \in O_k} \ln \sigma(\hat{o}_{u,s}^{k} - \hat{o}_{u,t}^{k}) + \lambda ||\Theta||_2^2 \quad (17)$$

其中，$O_k = \left\{(u, s, t) | (u, s) \in O_k^{+}, (u, t) \in O_k^{-}\right\}$表示训练数据集。$O_k^{+}$表示行为$k$下观测到的正用户-物品交互，$O_k^{-}$表示行为$k$下未观测到的用户-物品交互。$\Theta$表示所有模型参数的集合，$\sigma$表示Sigmoid函数，$\lambda$表示$\Theta$的正则化系数。

5 实验

5.1 实验设置

5.1.1 数据集。为减少偏差，我们采用与MB-GMN[37]相同的公开数据集（即**贝贝**、**淘宝**和**IJCAI**）[2]及预处理方法，统计信息如表1所示。

表1：数据集统计。

数据集	User	Item	交互	行为
贝贝	21,716	7,977	3,338,068	浏览,购物车,购买
淘宝	147,894	99,037	7,658,926	浏览,收藏,购物车,购买
IJCAI	423,423	874,328	36,203,512	浏览,收藏,购物车,购买

[2]https://github.com/akaxlh/MB-GMN

5.1.2 对比基线方法。为了进行全面比较，我们将 CIGF 与四类具有代表性的基线方法进行比较：i）基于神经网络的单行为模型，即 DMF [38] 和 AutoRec [28]；ii）基于神经网络的多行为模型，即 NMTR [10],、DIPN [12], 和 MATN [36]；iii）基于图神经网络的单行为模型，即 NGCF[34] 和 LightGCN [14]；iv）基于图神经网络的多行为模型，即 NGCF$_M$ [34]、LightGCN$_M$（以多行为图增强的 LightGCN [14]）、GHCF [6],和 MBGCN [19]。GHCF[3] 和 LightGCN[4] 使用公开代码，其他模型（DMF、AutoRec、NGCF、NMTR、DIPN、MATN、MBGCN 和 MB-GMN）的最佳结果取自 [37]。

5.1.3 评估指标。我们采用命中率（HR@N）和归一化折损累计增益（NDCG@N）来评估性能。默认情况下，我们在所有实验中设置 $N = 10$。在三个数据集上也可以得到其他指标（即 $N = 1$，5，20）的相似结果，但受篇幅限制，此处不予展示。实现细节见附录A.1。

5.2 总体性能比较

从表2中，我们可以从模型有效性的角度得出以下观察结果（复杂度分析见附录A.6）：

表2：总体比较。⋆表示与最佳基线（以下划线标示的数值）相比，CIGF的结果具有统计显著性，显著性水平为 p-值<0.05。

数据集	贝贝		淘宝		IJCAI	
模型	HR	NDCG	HR	NDCG	HR	NDCG
DMF	0.597	0.336	0.305	0.189	0.392	0.250
AutoRec	0.607	0.341	0.313	0.190	0.448	0.287
NGCF	0.611	0.375	0.302	0.185	0.461	0.292
LightGCN	0.643	0.378	0.373	0.235	0.443	0.283
NMTR	0.613	0.349	0.332	0.179	0.481	0.304
DIPN	0.631	0.394	0.317	0.178	0.475	0.296
MATN	0.626	0.385	0.354	0.209	0.489	0.309
NGCF$_M$	0.634	0.372	0.374	0.221	0.481	0.307
LightGCN$_M$	0.651	0.391	0.391	0.243	0.486	0.317
GHCF	0.608	0.378	0.415	0.241	-	-
MBGCN	0.642	0.376	0.369	0.222	0.463	0.277
MB-GMN	0.691	0.410	0.491	0.300	0.532	0.345
CIGF	**0.700⋆**	**0.443⋆**	**0.592⋆**	**0.383⋆**	**0.601⋆**	**0.400⋆**
%Improv	1.30%	8.05%	20.57%	27.67%	12.97%	15.94%

• CIGF 在全部三个数据集上始终表现出色。具体而言，在贝贝、淘宝和 IJCAI 数据集上，CIGF 的表现分别比最强的基线方法高出 **1.30%**、**20.57%** 和 **12.97%**（以命中率（*8.05%*、*27.67%* 和 *15.94%* 计），而以归一化折损累计增益计）。此外，在淘宝和 IJCAI 数据集上的性能提升远高于贝贝数据集。一个可能的原因是，贝贝数据集中不同类型行为的交互信息彼此覆盖（如附录 A.5 所示，购买过某物品的用户必定也浏览过该物品并将其加入购物车），这降低了高阶关系建模的重要性。

[3]https://github.com/chenchongthu/GHCF；由于非采样学习带来的内存使用量难以承受，GHCF 不适用于 IJCAI 数据集。[4]https://github.com/kuandeng/LightGCN

• 在大多数数据集上，NGCF 和 LightGCN 的表现优于 DMF 和 AutoRec，这证明了图神经网络在提取高阶协同信号方面的优势。通过区分不同的行为，NMTR、DIPN 和 MATN 的表现远胜于 DMF 和 AutoRec。这验证了提取并建模不同类型行为之间关系信息的必要性。

• NGCF、LightGCN、NMTR、DIPN 和 MATN 在大多数数据集上的表现都不如 NGCF$_M$、LightGCN$_M$、MBGCN 和 MB-GMN，这表明神经网络模型和单行为图神经网络模型在建模高阶多行为关系方面能力不足。这证明了同时考虑多行为和高阶关系的必要性。

5.3 CIGF 消融研究

表 3：不同 CIGF 变体的性能。

数据集	贝贝		淘宝		IJCAI	
模型	命中率 归一化折损累计增益		命中率 归一化折损累计增益		命中率 归一化折损累计增益	
基础模型	0.649	0.392	0.444	0.275	0.457	0.297
去除 CIGCN	0.660	0.410	0.460	0.286	0.495	0.322
去除 MESI	0.662	0.401	0.528	0.340	0.573	0.382
CIGF	**0.700**	**0.443**	**0.592**	**0.383**	**0.601**	**0.400**

5.3.1 关键组件的有效性。为评估 CIGF框架中各子模块的有效性，我们考察了三种模型变体：（1）基础模型：移除 CIGCN 部分（即直接使用行为特定图进行卷积），并用双线性模块替换 MESI 网络。该变体无法建模实例级高阶关系，并在多任务学习中使用相同输入。（2）去除 ***CIGCN***：移除 CIGCN 部分。（3）去除 ***MESI***：用双线性模块替换 MESI 部分。如表 3 所示，与基础模型相比，CIGCN 和 MESI 均带来了性能提升，完整的 CIGF框架取得了最佳结果。因此，我们认为实例级高阶多行为关系和分离输入均有效且相辅相成。将多行为数据同时视为“特征”和“标签”是必要的。

表 4：不同注意力变体的性能。

数据集	贝贝		淘宝		IJCAI	
方法	HR	NDCG	命中率 归一化折损累计增益		HR	NDCG
全局级	0.694	0.438	0.565	0.361	0.572	0.373
节点级	0.698	0.442	0.561	0.356	0.587	0.390
节点级+层	0.698	0.442	0.564	0.361	0.590	0.392
节点级+行为	0.698	0.441	0.588	0.379	0.599	0.398
节点级+行为+层	**0.700**	**0.443**	**0.592**	**0.383**	**0.601**	**0.400**

5.3.2 注意力模块的影响。为展示我们注意力模块的有效性，我们考虑了四种变体：（1）全局级：所有用户/物品共享全局级的注意力权重。（2）节点级：所有层和行为共享注意力权重，但不同用户/物品的权重各不相同。（3）节点级+层：所有行为共享注意力权重，但不同层或用户/物品的权重各不相同。（4）节点级+行为：所有层共享注意力权重，但不同的行为或用户/物品对应不同的权重。

表 5：多任务学习模块的影响。

数据集	贝贝		淘宝		IJCAI	
方法	HR	NDCG	HR	NDCG	HR	NDCG
CIGCN-SB	0.605	0.341	0.389	0.231	0.485	0.302
CIGCN-Bilinear	0.662	0.401	0.528	0.340	0.573	0.382
CIGCN-MMOE	0.663	0.391	0.541	0.338	0.546	0.339
CIGCN-PLE	0.653	0.381	0.521	0.320	0.526	0.325
CIGF	**0.700**	**0.443**	**0.592**	**0.383**	**0.601**	**0.400**

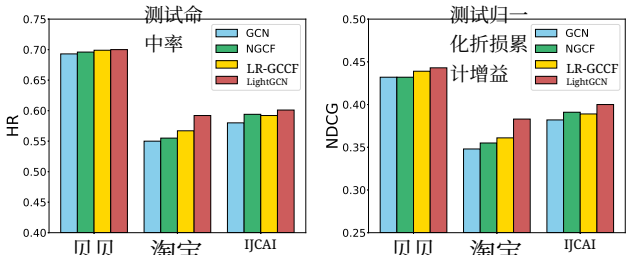

图6：GCN 聚合器的影响。

从表 4 所示的结果可以看出，在大多数情况下，全局级注意力在所有变体中表现最差，这表明为每个节点学习定制信息的重要性。此外，所有增强型节点级变体的表现均优于纯节点级注意力机制，而我们提出的注意力机制在三个数据集上均取得了最佳表现。这些结果表明，我们提出的行为级和层级节点级注意力机制在高阶多行为关系选择中的有效性与合理性。

5.3.3 多任务学习模块的影响。为进一步展示我们提出的 MESI 在多任务学习中的优越性，我们用四种最先进的多任务学习模块替换它，即共享底层 [5],双线性 [6], MMOE [24],和 PLE [31],，并将它们应用于 CIGCN 顶层，以实现多行为推荐。请注意，CIGCN 生成的表示会被用作分离输入。为使其适用于这四种使用相同输入的模块，我们对这些表示取平均，得到一个统一输入。由此得到的变体分别命名为 CIGCN-SB、CIGCN-Bilinear、CIGCN-MMOE、CIGCN-PLE 和 CIGF。结果汇总于表 5。如我们所见，CIGCN-SB 在所有数据集上都是所有多任务学习模型中表现最差的。CIGCN-Bilinear 用轻量级矩阵变换替换了 CIGCN-SB 中的神经网络预测头，表现更好。可能的原因是，轻量级操作能够降低过拟合风险。此外，CIGCN-MMOE 和 CIGCN-PLE 都采用了带有自适应注意力权重的门控网络进行信息融合，因此表现优于静态且权重相同的 CIGCN-SB。最后，我们的 MESI 在所有数据集上始终表现最佳。这验证了分离输入在多任务学习中的有效性。

5.3.4 GCN 聚合器的影响。为探究不同 GCN 聚合器的影响，我们比较了所提方法的不同变体，
提出的模型采用了不同的 GCN 聚合器，包括 *GCN* 聚合器 [21], *NGCF* 聚合器 [34], *LR-GCCF*聚合器 [7], 和 *LightGCN*聚合器 [14]。实验结果如图6所示。可以看出，NGCF 聚合器在所有数据集上的表现都优于 GCN 聚合器。一个可能的原因是

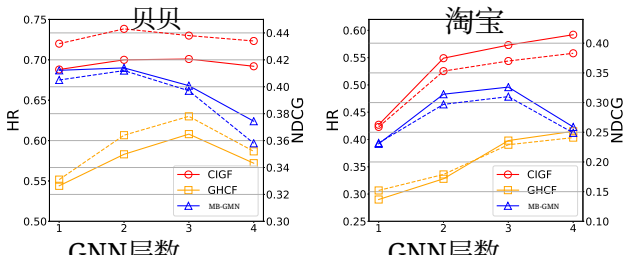

图7：层数的影响。实线和虚线分别表示命中率和归一化折损累计增益。

表6：选出的排名前三和排名后三的关系。

数据集	关系	二阶	三阶	四阶
贝贝	前 3 名	CC、CV、PP	-	-
	底部-3	CP, PV, VP	-	-
淘宝	顶部-3	FP、CP、PC	FPP、CPP、PCP	PFFP、CFFP、FPPV
	倒数第三	PP、CC、FF	PPP、PPV、PPF	PPPF、PPPC、PPPP
IJCAI	前三名	CV、FV、PV	PVV、CVV、FFF	PVPV、CPPV、CVPV
	后三名	FC , CC, PC	FCF, CCF, FCP	FCFC , CCFC, PCFC

NGCF 聚合器引入的额外特征交互提供了更多信息。我们还发现，LR-GCCF聚合器在几乎所有数据集上的表现都略优于 NGCF 聚合器。原因在于，移除变换矩阵可以缓解过拟合。此外，LightGCN聚合器通过同时移除变换矩阵和激活函数，在所有数据集上都取得了最佳表现。

5.4 模型设计的深入分析

在本部分中，我们通过实验深入分析将多行为数据视为“特征”时的实例级高阶关系建模，以及将多行为数据视为“标签”时可能出现的梯度冲突。

5.4.1 实例级高阶关系建模。我们改变 CIGF 的深度，以探究模型能否从实例级高阶关系中受益，并将结果与对行为级高阶关系进行建模的 GHCF 和 MB-GMN 进行比较。由于篇幅所限，我们仅在图7中展示贝贝和淘宝数据集上的结果，另一个数据集上的结果与之相符。可以看到，随着层数增加，CIGF 的表现始终优于其他方法。此外，我们还发现，CIGF 在贝贝数据集上的表现保持稳定，在淘宝数据集上则持续提升；而随着层数增加，MBGCN 在两个数据集上的表现都迅速下降，GHCF 在贝贝数据集上的表现也迅速下降。这一观察验证了我们提出的方法在实例级高阶关系建模方面的有效性。

实例级 高阶关系有助于最终推荐。此外，它还可以揭示促使用户产生物品偏好的潜在原因。为此，我们根据表6中所有用户的平均注意力权重，选择并展示每个阶数所有可能关系中的排名前三和排名后三的关系。请注意，由于我们的模型在该数据集上使用两层时效果最佳，贝贝上只有二阶关系。正如我们所看到的，三阶关系
$user \xrightarrow{favor} item \xrightarrow{be\ purchased\ by} user \xrightarrow{purchase} item$ 具有

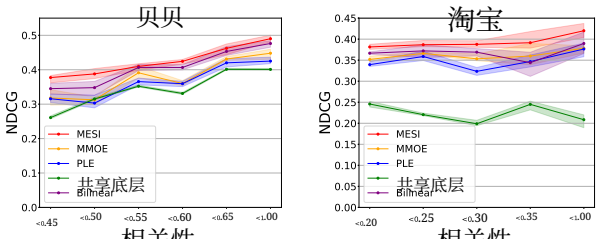

图8：具有不同的行为相关性的用户群体的平均性能。

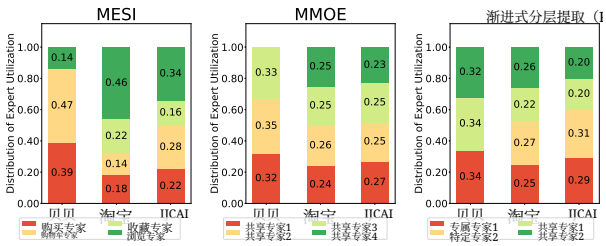

图 9：基于门控的模型中的专家利用情况

最高的平均注意力权重。一个可能的解释是，用户倾向于购买相似用户购买过的物品。此外，二阶关系 $user \xrightarrow{favor} item \xrightarrow{be\ carted\ by} user$ 在 IJCAI 上的平均注意力权重最低。这一合理性可在附录 A.5 中得到验证：在 IJCAI 上，用户仅对相同物品具有收藏和购物车行为的概率为零。

5.4.2 梯度冲突分析。 为验证我们的模型能够缓解潜在的梯度冲突，我们针对行为相关性水平不同的用户群体开展实验。具体而言，我们根据所有行为之间的平均皮尔逊相关系数 [4] 将测试集划分为六个用户群体。平均皮尔逊相关系数的计算方法参见附录A.4。为确保公平比较，我们从每个用户群体中选取一个子集，使每位用户的交互次数保持固定，从而避免节点度对结果产生潜在影响 [34]。图 8 展示了实验结果。由于篇幅所限，我们省略了 IJCAI 数据集上的结果，其趋势一致。为获得更严谨的结果，我们将每项实验运行 5 次，并在图中绘制均值和波动范围。我们发现，MESI 在所有用户群体中始终优于所有基线方法，这进一步证明了 MESI 在多任务学习中的优越性。此外，随着行为相关性的增强，MESI 的性能有所提升，而其他基线方法的性能则出现波动甚至下降。一个可能的原因是，在不同任务之间迁移知识时，潜在的梯度冲突会导致负迁移。

为了理解我们提出的 MESI 能够缓解潜在梯度冲突的原因，我们开展实验，对比 MESI 与其他基于门控的模型（MMOE 和 PLE）对专家网络的利用情况。接下来 [31], 我们在图 9 中展示目标行为预测所使用的专家网络的平均权重分布。请注意，我们省略了其他行为所使用的门控，因为我们的目标是预测目标行为的交互概率。此外，为便于比较，我们在贝贝数据集上将 MMOE 和 PLE 的专家网络数量固定为 3，在淘宝和 IJCAI 数据集上固定为 4，对于 MMOE 和 PLE 均如此。结果表明，与 MMOE 和 PLE 对所有专家网络近乎均匀的权重分布相比，我们的 MESI 能够更好地区分不同的专家网络。因此，我们的 MESI 可以有选择地利用不同的行为信息来更新梯度，从而避免潜在的冲突。

6 结论

本文提出了用于多行为推荐的 CIGF 框架。为显式建模实例级高阶关系，我们引入了 CIGCN模块，该模块利用矩阵乘法作为交互算子来生成高阶交互图，并在这些图上执行图卷积，以探索关系整合。为缓解潜在的梯度冲突，我们提出了 MESI网络，显式采用特定于行为的独立输入 。这样可以降低负迁移的风险。我们在三个真实世界数据集上进行了全面实验，结果表明，所提出的 CIGF 在所有三个数据集上均优于所有最先进的方法。进一步分析表明，CIGF 能够充分捕获高阶关系，并有效缓解负迁移。

致谢

本研究部分得到科技创新2030—重大项目（项目编号：2021ZD0201404）和Aminer·深圳·科学超脑的资助。我们还感谢 MindSpore [1] 对本研究提供部分支持；MindSpore 是一种新型深度学习计算框架。

参考文献

[1] 2020。MindSpore。https://www.mindspore.cn[2] Martín Abadi、Paul Barham、Jianmin Chen、Zhifeng Chen、Andy Davis、Jeffrey Dean、Matthieu Devin、Sanjay Ghemawat、Geoffrey Irving、Michael Isard 等。2016。{TensorFlow}：一个用于{大规模} 机器学习的系统。载于第 *12* 届 *USENIX* 操作系统设计与实现研讨会（*OSDI 16*）。265–283。[3] Austin R Benson、David F Gleich 和 Jure Leskovec。2016。复杂网络的高阶组织。*Science* 353，6295（2016），163–166。[4] Michael R Berthold 和 Frank Höppner。2016。使用欧几里得距离和皮尔逊相关系数对时间序列进行聚类。*arXiv* 预印本：*1601.02213*（2016）。[5] Rich Caruana。1997。多任务学习。*Machine learning* 28，1（1997），41–75。[6] Chong Chen、Weizhi Ma、Min Zhang、Zhaowei Wang、Xiuqiang He、Chenyang Wang、Yiqun Liu 和 Shaoping Ma。2021。图异构多关系推荐。载于*AAAI*，第 35 卷。3958–3966。[7] Lei Chen、Le Wu、Richang Hong、Kun Zhang 和 Meng Wang。2020。重新审视基于图的协同过滤：一种线性残差图卷积网络方法。载于*AAAI*，第 34 卷。27–34。[8] Yuxiao Dong、Nitesh V Chawla 和 Ananthram Swami。2017。metapath2vec：面向异构网络的可扩展表示学习。载于*SIGKDD*。[9] Yifan Feng、Haoxuan You、Zizhao Zhang、Rongrong Ji 和 Yue Gao。2019。超图神经网络。载于*AAAI*，第 33 卷。3558–3565。[10] Chen Gao、Xiangnan He、Dahua Gan、Xiangning Chen、Fuli Feng、Yong Li、Tat-Seng Chua 和 Depeng Jin。2019。基于多行为数据的神经多任务推荐。载于*ICDE*。IEEE，1554–1557。[11] Xavier Glorot 和 Yoshua Bengio。2010。理解深度前馈神经网络训练的难点。载于*AISTATS*。249–256。[12] Long Guo、Lifeng Hua、Rongfei Jia、Binqiang Zhao、Xiaobo Wang 和 Bin Cui。2019。购买还是浏览？：利用结合多种行为的基于注意力的深度网络预测实时购买意图。载于*SIGKDD*。1984–1992。[13] Kaiming He、Xiangyu Zhang、Shaoqing Ren 和 Jian Sun。2016。用于图像识别的深度残差学习。载于*CVPR*。770–778。[14] Xiangnan He、Kuan Deng、Xiang Wang、Yan Li、Yongdong Zhang 和 Meng Wang。2020。LightGCN：简化并强化用于推荐的图卷积网络。*arXiv* 预印本 *arXiv:2002.02126*（2020）。[15] Xiangnan He、Lizi Liao、Hanwang Zhang、Liqiang Nie、Xia Hu 和 Tat-Seng Chua。2017。神经协同过滤。载于*WWW*。173–182。

[16] Binbin Hu、Chuan Shi、Wayne Xin Zhao 和 Philip S Yu。2018。利用基于元路径的上下文，通过神经协同注意力模型进行 Top-N 推荐。载于*SIGKDD*。1531–1540。[17] Chao Huang。2021。推荐领域异构关系学习的最新进展。*arXiv* 预印本 *arXiv:2110.03455*（2021）。[18] Robert A Jacobs、Michael I Jordan、Steven J Nowlan 和 Geoffrey E Hinton。1991。局部专家自适应混合。神经计算 3，1（1991），79–87。[19] Bowen Jin、Chen Gao、Xiangnan He、Depeng Jin 和 Yong Li。2020。基于图卷积网络的多行为推荐。载于*SIGIR*。[20] Diederik P Kingma 和 Jimmy Ba。2014。Adam：一种随机优化方法。*arXiv* 预印本 *arXiv:1412.6980*（2014）。[21] Thomas N Kipf 和 Max Welling。2016。基于图卷积网络的半监督分类。*arXiv* 预印本 *arXiv:1609.02907*（2016）。[22] Jianxun Lian、Xiaohuan Zhou、Fuzheng Zhang、Zhongxia Chen、Xing Xie 和 Guangzhong Sun。2018。xdeepfm：结合显式和隐式特征交互的推荐系统。载于*SIGKDD*。1754–1763。[23] Hu Linmei、Tianchi Yang、Chuan Shi、Houye Ji 和 Xiaoli Li。2019。用于半监督短文本分类的异构图注意力网络。载于*EMNLP-IJCNLP*。4821–4830。[24] Jiaqi Ma、Zhe Zhao、Xinyang Yi、Jilin Chen、Lichan Hong 和 Ed H Chi。2018。利用多门控专家混合对多任务学习中的任务关系进行建模。载于*SIGKDD*。1930–1939。[25] 孟畅、Ziqi Zhao、Wei Guo、Yingxue Zhang、Haolun Wu、Chen Gao、Dong Li、Xiu Li 和 Ruiming Tang。2022。面向多行为推荐的由粗到细的知识增强多兴趣学习框架。*arXiv* 预印本 *arXiv:2208.01849*（2022）。[26] 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）。[27] Michael Schlichtkrull、Thomas N Kipf、Peter Bloem、Rianne Van Den Berg、Ivan Titov 和 Max Welling。2018。利用图卷积网络对关系数据建模。载于欧洲语义网会议。Springer，593–607。

[28] Suvash Sedhain、Aditya Krishna Menon、Scott Sanner 和 Lexing Xie。2015。AutoRec：自编码器与协同过滤的结合。载于 *WWW*。111–112。[29] Xiaoyuan Su 和 Taghi M Khoshgoftaar。2009。协同过滤技术综述。人工智能进展 2009（2009）。[30] Jianing Sun、Yingxue Zhang、Wei Guo、Huifeng Guo、Ruiming Tang、Xiuqiang He、Chen Ma 和 Mark Coates。2020。用于推荐的邻居交互感知图卷积网络。载于 *SIGIR*。1289–1298。[31] 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 *RecSys*。269–278。[32] Lisa Torrey 和 Jude Shavlik。2010。迁移学习。载于 机器学习应用与趋势研究手册：算法、方法与技术。IGI Global，242–264。[33] Ruoxi Wang、Bin Fu、Gang Fu 和 Mingliang Wang。2017。用于广告点击预测的深度与交叉网络。载于 *ADKDD'17* 论文集。1–7。[34] Xiang Wang 、 Xiangnan He 、 Meng Wang 、 Fuli Feng 和 Tat-Seng Chua。2019。神经图协同过滤。载于 *SIGIR*。165–174。[35] Xiao Wang、Houye Ji、Chuan Shi、Bai Wang、Yanfang Ye、Peng Cui 和 Philip S Yu。2019。异构图注意力网络。载于 *WWW*。2022–2032。[36] Lianghao Xia、Chao Huang、Yong Xu、Peng Dai、Bo Zhang 和 Liefeng Bo。2020。通过记忆增强型Transformer网络进行推荐的多路行为关系学习。载于 *SIGIR*。2397–2406。[37] Lianghao Xia、Yong Xu、Chao Huang、Peng Dai 和 Liefeng Bo。2021。用于多行为推荐的图元网络。载于 *SIGIR*。757–766。[38] Hong-Jian Xue、Xinyu Dai、Jianbing Zhang、Shujian Huang 和 Jiajun Chen。2017。推荐系统的深度矩阵分解模型。载于 *IJCAI*，第17卷。澳大利亚墨尔本，3203–3209。[39] Chuxu Zhang、Dongjin Song、Chao Huang、Ananthram Swami 和 Nitesh V Chawla。2019。异构图神经网络。载于 *SIGKDD*。793–803。[40] Weifeng Zhang、Jingwen Mao、Yi Cao 和 Congfu Xu。2020。用于多行为推荐的多路图神经网络。载于 *CIKM*。2313–2316。

A 附录

附录A.1 参数设置

我们提出的 CIGF 使用 TensorFlow [2] 实现。为确保公平比较，所有模型的用户和物品嵌入维度均设为 16，并采用 Xavier 方法初始化模型参数 [11]。我们使用 Adam [20] 优化模型，学习率和批量大小分别设为 0.001 和 256。此外，图模型的图卷积网络层数在 {1,2,3,4,5} 中搜索。为简化起见，图压缩层中仅使用一个注意力头，因为这已取得足够的性能提升。其他参数设置与 MB-GMN 保持一致 [37]。所有实验均运行 5 次，并报告平均结果。

A.2 多任务学习中的耦合梯度问题

为简化起见，我们假设现有多任务学习模型中学习到的用户/物品表示可以表示为：

$$\mathbf{x}_u^* = g^u(\mathbf{x}_u, \mathbf{A}), \mathbf{y}_i^* = g^i(\mathbf{y}_i, \mathbf{A}) \tag{18}$$

其中，$g^u(\cdot)$ 和 $g^i(\cdot)$ 表示表示学习函数，$\mathbf{x}_u$ 和 $\mathbf{y}_i$ 是用户 u 和物品 i 的初始嵌入，$\mathbf{A}$ 是 MBG G 对应的邻接矩阵。注意，对于 $g^u(\cdot)$ 和 $g^i(\cdot)$，$\mathbf{A}$ 是可选的，因此可以将它们推广为非图函数。

将 $(\mathbf{x}_u^*, \mathbf{y}_i^*)$ 作为多任务学习的相同输入，损失函数可表示为：

$$\begin{aligned}\mathcal{L}_{u,i} &= \sum_{k=1}^{K} L(\hat{o}_{u,i}^k - o_{u,i}^k) \\ &= \sum_{k=1}^{K} L(f_k(\mathbf{x}_u^*, \mathbf{y}_i^*) - o_{u,i}^k)\end{aligned} \tag{19}$$

其中，$\hat{o}_{u,i}^k$ 表示用户 u 在第 k 种行为下与物品 i 交互的预测概率，$o_{u,i}^k$ 是真实标签，$L(\cdot)$ 是损失函数，$f_k(\cdot)$ 是多任务学习模型中的预测函数。于是我们有：

$$\begin{aligned}\frac{\partial \mathcal{L}_{u,i}}{\partial(\mathbf{x}_u^* \circ \mathbf{y}_i^*)} &= \sum_{k=1}^{K} \frac{\partial L(f_k(\mathbf{x}_u^*, \mathbf{y}_i^*) - o_{u,i}^k)}{\partial(\mathbf{x}_u^* \circ \mathbf{y}_i^*)} \\ &= \sum_{k=1}^{K} \frac{\partial f_k(\mathbf{x}_u^*, \mathbf{y}_i^*)}{\partial(\mathbf{x}_u^* \circ \mathbf{y}_i^*)} * L'(f_k(\mathbf{x}_u^*, \mathbf{y}_i^*) - o_{u,i}^k) \\ &= \sum_{k=1}^{K} a_{u,i}^k \mathbf{r}^k \\ &= \sum_{k=1}^{K} \mathbf{r}'^k\end{aligned} \tag{20}$$

其中，$(\circ)$ 表示哈达玛积运算，$a_{u,i}^k = L'(f_k(\mathbf{x}_u^*, \mathbf{y}_i^*) - o_{u,i}^k)$ 是一个标量。$\mathbf{r}^k = \frac{\partial f_k}{\partial(\mathbf{x}_u^* \circ \mathbf{y}_i^*)}$。由于 $\mathbf{r}^k$ 表示标量对向量的导数，因此它也是一个向量。$\forall k \in \{1, 2, \ldots, K\}$ 中，$\mathbf{r}'^k$ 决定向量 $\mathbf{x}_u^* \circ \mathbf{y}_i^*$ 的更新幅度和方向。

我们可以看到，所有行为的梯度都是耦合的。与第3.3节类似，我们可以发现，如果在多任务学习中使用相同的输入，就会因梯度耦合问题而产生梯度冲突。

A.3 MESI 在多任务学习中的解耦梯度

相比之下，我们提出的 MESI 在多任务学习中采用独立输入 $\mathbf{x}_{u,k}^*$ 和 $\mathbf{y}_{i,k}^*$ $(k \in \{1, 2, \ldots, K\})$。MESI 的损失函数可表示为：

$$\begin{aligned}\mathcal{L}^*_{u,i} &= \sum_{k=1}^{K} L^*(\hat{o}_{u,i}^k - o_{u,i}^k) \\ &= \sum_{k=1}^{K} L^*(h^k(\sum_{j=1}^{K} \mathbf{g}_{u,i}^k(j) \cdot \mathbf{f}_{u,i}^j) - o_{u,i}^k)\end{aligned} \tag{21}$$

其中

$$\mathbf{g}_{u,i}^k = Softmax(\mathbf{W}_g(\mathbf{x}_{u,k}^* || \mathbf{y}_{i,k}^*) + \mathbf{b}_g) \tag{22}$$

$$\mathbf{f}_{u,i}^k = \mathbf{x}_{u,k}^* \circ \mathbf{y}_{i,k}^* \tag{23}$$

$\hat{o}_{u,i}^k$ 表示用户 u 在第 k 种行为下与物品 i 发生交互的预测概率，$o_{u,i}^k$ 是真实标签，$L^*(\cdot)$ 是用于优化的损失函数。此外，$\mathbf{g}_{u,i}^k$ 表示任务 k 的门控，$\mathbf{f}_{u,i}^k$ 表示由输入 $\mathbf{x}_{u,k}^*$ 和 $\mathbf{y}_{i,k}^*$ 生成的专家，详见第4.3节。

对于任意待优化的参考输入向量 $\mathbf{x}_{u,t}^*$ 和 $\mathbf{y}_{i,t}^*$（$t \in \{1, 2, \ldots, K\}$），我们有：

$$\begin{aligned}\frac{\partial \mathcal{L}^*_{u,i}}{\partial(\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*)} &= \sum_{k=1}^{K} \frac{\partial L^*(h^k(\sum_{j=1}^{K} \mathbf{g}_{u,i}^k(j) \cdot \mathbf{f}_{u,i}^j) - o_{u,i}^k)}{\partial(\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*)} \\ &= \sum_{k=1}^{K} \frac{\partial(\sum_{j=1}^{K} \mathbf{g}_{u,i}^k(j) \cdot \mathbf{f}_{u,i}^j)}{\partial(\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*)} * a_{u,i}^k \\ &= \sum_{k=1}^{K} (\frac{\partial(\mathbf{g}_{u,i}^k(t) \cdot (\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*))}{\partial(\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*)} + \frac{\partial(\sum_{\substack{j=1 \\ j \neq t}}^{K} \mathbf{g}_{u,i}^k(j) \cdot \mathbf{f}_{u,i}^j)}{\partial(\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*)}) * a_{u,i}^k \\ &= \sum_{k=1}^{K} \mathbf{g}_{u,i}^k(t) * a_{u,i}^k + \frac{\partial(\sum_{\substack{j=1 \\ j \neq t}}^{K} \mathbf{g}_{u,i}^t(j) \cdot \mathbf{f}_{u,i}^j)}{\partial(\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*)} * a_{u,i}^t + \\ &\sum_{\substack{k=1 \\ k \neq t}}^{K} \frac{\partial(\sum_{j=1}^{K} \mathbf{g}_{u,i}^k(j) \cdot \mathbf{f}_{u,i}^j)}{\partial(\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*)} * a_{u,i}^k \\ &= \sum_{k=1}^{K} \mathbf{g}_{u,i}^k(t) * a_{u,i}^k + \frac{\partial(\sum_{\substack{j=1 \\ j \neq t}}^{K} \mathbf{g}_{u,i}^t(j) \cdot \mathbf{f}_{u,i}^j)}{\partial(\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*)} * a_{u,i}^t + 0 \\ &= \sum_{k=1}^{K} \mathbf{g}_{u,i}^k(t) * a_{u,i}^k + \sum_{\substack{j=1 \\ j \neq t}}^{K} \frac{\partial(\mathbf{g}_{u,i}^t(j) \cdot \mathbf{f}_{u,i}^j)}{\partial(\mathbf{x}_{u,t}^* \circ \mathbf{y}_{i,t}^*)} * a_{u,i}^t\end{aligned} \tag{24}$$

其中

$$a_{u,i}^{k} = (h^{k'}(\sum_{j=1}^{K} \mathbf{g}_{u,i}^{k}(j) \cdot \mathbf{f}_{u,i}^{j}) * L^{*'}(h^{k}(\sum_{j=1}^{K} \mathbf{g}_{u,i}^{k}(j) \cdot \mathbf{f}_{u,i}^{j}) - o_{u,i}^{k})$$

是一个标量。

在上述推导过程中可以清楚地看到，我们提出的 MESI 将不同行为的梯度解耦，并有选择地利用不同行为的信息来更新梯度，从而缓解了梯度冲突问题。

A.4 皮尔逊相关系数的计算

我们选择皮尔逊相关系数将用户划分为不同的测试组。用户 u 的行为 i 和 j 之间的皮尔逊相关系数可按如下方式计算：

$$r_u^{s,t} = \frac{\sum_{j=1}^{N} \left(\mathrm{Y}_{u,j}^{s} - \bar{\mathrm{Y}}_{u}^{s}\right)\left(\mathrm{Y}_{u,j}^{t} - \bar{\mathrm{Y}}_{u}^{t}\right)}{\sqrt{\sum_{j=1}^{N} \left(\mathrm{Y}_{u,j}^{s} - \bar{\mathrm{Y}}_{u}^{s}\right)^2} \sqrt{\sum_{j=1}^{N} \left(\mathrm{Y}_{u,j}^{t} - \bar{\mathrm{Y}}_{u}^{t}\right)^2}} \tag{25}$$

其中，$\mathrm{Y}_{u,j}^{s}$ 和 $\mathrm{Y}_{u,j}^{t}$ 分别表示用户-物品交互矩阵 $\mathbf{Y}^s$ 和 $\mathbf{Y}^t$ 中第 u 行、第 j列的元素；$\bar{\mathrm{Y}}_{u}^{s}$ 和 $\bar{\mathrm{Y}}_{u}^{t}$ 表示输入向量 $\mathbf{Y}_u^s$ 和$\mathbf{Y}_u^t$的均值。N是输入向量的长度，也就是物品的数量。得到每对行为之间的皮尔逊相关系数后，我们可以计算每位用户 u 的所有行为之间的最终平均皮尔逊相关系数：

$$r_u = \frac{2}{K(K-1)} \sum_{s=1}^{K-1} \sum_{t=s+1}^{K} r_u^{s,t} \tag{26}$$

A.5 标签相关性分析

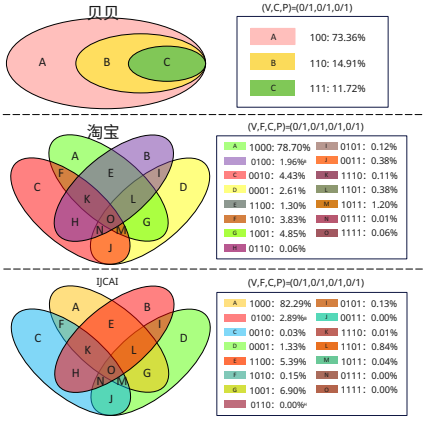

图 10：三个数据集上的标签相关性维恩图。1/0 表示是否具有此类行为。例如，0110 表示这些用户只有与物品相关的收藏和购物车行为。

多行为数据可以视为多任务监督学习的“标签”。图 10 展示了将多行为数据视为标签时的标签相关性维恩图，其中不同的重叠区域代表不同的标签相关性。

A.6 复杂度与效率分析

A.6.1 复杂度分析。**时间复杂度**。我们分析了 CIGF 的时间复杂度，其中 CIGCN模块是主要开销。CIGCN 的计算复杂度为 $\sum_{k=1}^{K}\sum_{l=1}^{L} O\left(\left|\mathrm{B}_{v,\,k}^{l}\right| \cdot d\right)$，其中 $\left|\mathrm{B}_{v,\,k}^{l}\right|$ 表示集合 $\mathrm{B}_{v,\,k}^{l}$中所有图的边数，K表示行为数，L表示层数，d表示嵌入维度。在 CIGCN 中，集合 $\mathrm{B}_{v,\,k}^{l}$中的稠密图 $\mathrm{B}_{l,\,v,\,k}^{s}$会转换为 l稀疏图进行计算。由于 l通常很小，其时间复杂度与现有图神经网络相当，这一点也通过 A.6.2 节中的实验得到了进一步验证。

空间复杂度。我们提出的 CIGF 中可学习参数主要来自用户和物品嵌入$\mathbf{x}_u$ 和$\mathbf{y}_i$，这与现有的图神经网络类似。此外，由于将集合 B_v^l中的稠密图$\mathrm{B}_{l,\,v,\,k}^{s}$转换为行为指定稀疏图 G_1，G_2，$\cdots$，G^K 进行计算，因此无需额外的内存空间来存储这些图，使得中间过程的内存占用可以接受。

表7：所有三个数据集上不同方法的训练时间比较（每轮次秒数）。

训练时间（秒） 模型 \ 数据集	贝贝	淘宝	IJCAI
GHCF	**8.31**	20.02	-
MB-GMN	14.95	27.03	79.25
CIGF	10.37	**16.78**	**61.60**

表 8：三种数据集上不同方法的测试时间比较（每轮次耗时，秒）。

测试时间（秒） 模型 \ 数据集	贝贝	淘宝	IJCAI
GHCF	9.88	34.48	-
MB-GMN	2.96	19.83	68.77
CIGF	**2.46**	**18.79**	**59.56**

A.6.2 效率分析。除了模型效果之外，训练效率也很重要。表7展示了我们的 CIGF 与两个具有代表性的基线方法在全部三个数据集上的训练时间（一个轮次）对比。最佳基线方法 MB-GMN 的训练时间最长，而我们的 CIGF 在三个数据集上分别缩短了 **30.64%**、**37.92%** 和 **22.27%** 的时间。此外，尽管 GHCF 在贝贝数据集上的速度略快于我们的 CIGF，但由于非采样学习导致内存使用量过高，它无法应用于 IJCAI 数据集。另外，如表8所示，在三个数据集上，我们提出的 CIGF 比其他模型中速度最快的模型分别快 **16.89%**、**5.24%** 和 **13.39%**。GHCF 的训练效率表现良好，但测试表现最差。可能的原因是，GHCF 的非采样学习损失大幅提高了损失计算效率，从而显著提升了训练效率。而 GHCF 中构成其主要复杂度的图神经网络部分更为复杂，因此测试耗时更长。总而言之，我们认为 CIGF 的整体效率最佳。