arXiv:2305.04410v1 [cs.IR] 8 May 2023

WSFE：用于协同过滤中有效用户细分的 Wasserstein 子图特征编码器

Yankai Chen 香港中文大学
ykchen@cse.cuhk.edu.hk

Yifei Zhang 香港中文大学
yfzhang@cse.cuhk.edu.hk

Menglin Yang 香港中文大学
mlyang@cse.cuhk.edu.hk

Zixing Song
中国香港中文大学（香港）
zxsong@cse.cuhk.edu.hk

Chen Ma 香港城市大学
chenma@cityu.edu.hk

Irwin King 香港中文大学
king@cse.cuhk.edu.hk

摘要

基于向量化嵌入最大化用户-物品交互，是当前推荐模型中的标准流程。尽管这些方法在物品推荐方面表现优异，但它们却隐式地降低了对嵌入空间中用户间相似性建模的优先级；因此，相似用户识别的表现不佳，通常还需要额外的处理方案。为避免全面的模型重新训练，我们提出了 WSFE，这是一种与模型无关且无需训练的表示编码器，可灵活地即时应用于有效的用户细分。在最优传输理论的支撑下，WSFE 生成的编码表示，在真实空间与嵌入空间之间呈现出匹配的用户间相似性/距离度量。我们将 WSFE 融入六种最先进的推荐模型，并在六个真实世界数据集上开展了广泛实验。实证分析充分证明了 WSFE 的优越性与通用性，能够推动推荐领域中具有不同底层目标的多种下游任务。

ACM 参考格式：
Yankai Chen、Yifei Zhang、Menglin Yang、Zixing Song、Chen Ma 和 Irwin King。2023年。WSFE：用于协同过滤中有效用户细分的 Wasserstein 子图特征编码器。载于第*46*届国际 *ACM* 信息检索研究与发展会议论文集（*SIGIR '23*），*2023*年*7*月*23*日至*27*日，台北，台湾。ACM，纽约，纽约州，美国，5页。https://doi.org/10.1145/3539618.3592089

1 引言

协同过滤（CF）作为一种执行个性化建模和预测的有效策略，已被广泛部署用于推荐。协同过滤模型 [4,13, 25, 45, 47, 50] 的一种普遍学习范式是将用户和商品参数化为向量化嵌入，并学习重构用户的历史交互。

因此，学习到的嵌入有助于理解目标用户的多样化偏好，并预测他们未来的行为。

允许为个人或课堂使用而免费制作本作品全部或部分内容的数字或纸质副本，但不得为营利或商业利益制作或分发副本，且副本必须保留本声明和首页上的完整引文。必须尊重本作品中由作者以外的其他方拥有版权的组件。注明出处后可进行摘要引用。若要以其他方式复制、再版、发布到服务器或分发至列表，须事先获得明确许可和/或支付费用。请通过 permissions@acm.org 申请许可。*SIGIR '23*，*2023*年*7*月*23*日至*27*日，台北，台湾 © 2023年 版权所有归所有者/作者所有。出版权已授权给 ACM。ACM ISBN 978-1-4503-9408-6/23/07...$15.00 https://doi.org/10.1145/3539618.3592089

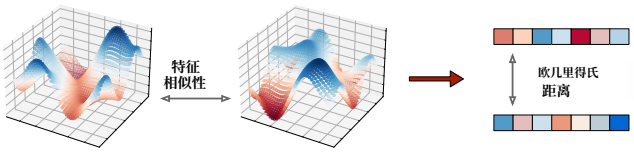

图 1：WSFE 对经验特征分布（左侧）的相似性进行编码，并与相应嵌入距离（右侧）相对应的示意图。

除了反映用户对商品的偏好外，学习到的用户嵌入还应具备另一项理想特性，即明确捕捉用户间相似性；这有助于直观识别具有相似用户兴趣和偏好关系的用户，为以用户为中心的分析和应用奠定基础，尤其有利于群组推荐和广告 [37, 39]。然而，近期模型通常将这一特性置于次要地位，甚至忽略不计 [13, 22, 29, 44, 46, 48]。为了弥补相似用户识别效果不佳的问题，可能需要彻底重新训练模型。为解决这一问题，我们致力于在协同过滤中编码高质量的嵌入，使其能够高效、无缝地支持用户细分任务。

在本文中，我们提出了 Wasserstein 子图特征编码器（WSFE），以用户-物品交互图的形式显式建模用户行为，并利用其高阶子图模式来度量用户间相似性。我们注意到，具有相似交互行为的用户自然会共享重叠的子图模式。基于这一观察，一种直接的解决方案是穷举计算底层子图中所有节点之间的相似性；然而，这在实践中可能难以处理，主要是因为图探索中的节点规模呈指数级增长。相反，我们提出的 WSFE 通过直接编码用户的子图潜在特征来捕获用户相似性，使其与模型无关，并能够灵活适用于各种基于图的推荐模型。具体而言，如图 1 所示，我们假设用户偏好遵循一个未知的高维概率分布；这种独特的偏好分布由在物品推荐任务中得到良好学习的潜在特征进行经验观测和表示。随后，WSFE 利用最优传输理论 [19, 32, 35, 42]中的 Wasserstein 度量，显式捕获分布距离。因此，编码后的用户表示能够有效反映其真实的物品交互相似性，从而产生匹配的欧几里得距离度量，便于在嵌入空间中进行用户细分。

总而言之，我们的贡献概括如下：

• 据我们所知，我们首次致力于提升协同过滤中用于有效用户细分的嵌入质量，同时不损害物品推荐的模型评估。

• 我们提出了 WSFE，通过捕捉高阶用户-物品交互图模式中的特征相似性，实现有效的表示编码。WSFE 适用于任意基于图的模型，且无需训练；因此，只要骨干模型训练完毕，就可以随时调用。

• 我们在六个真实世界数据集上，将 WSFE 融合到六种最先进的模型中，开展了广泛的实验。我们不仅在实证评估中展示了其性能优势，还为未来的研究提供了技术讨论。

2 WSFE 方法

2.1 预备知识

基于图的协同过滤。鉴于用户-物品交互图，基于图方法的一般思路是捕捉高跳邻居中的协同过滤信号。在本文中，我们研究 图卷积网络（*GCN*），通过利用拓扑结构 [3, 18, 40, 41] 平滑潜在特征来学习节点表示。该方法迭代传播邻域信息至目标节点，例如用户 u，其可以抽象为：

$$\boldsymbol{v}_{ngh\to u}^{(l)} = Prop\Big(\{\boldsymbol{v}_i^{(l-1)} : i \in \mathcal{N}(u)\}\Big), \tag{1}$$

其中，$\boldsymbol{v}_{ngh\to u}^{(l)}$是从 u的邻居集合$\mathcal{N}(u)$中的交互商品经过 l层传播后的表示。借助传播的信息，通过聚合中心节点和邻居节点 [11, 12, 53]的特征，迭代更新节点嵌入。**最优传输与 Wasserstein 度量**。最优传输是一个一般性问题，即尽可能高效地将一个质量分布（例如 P）转移到另一个质量分布（例如 Q）。由此得到的最小 L_2 代价可称为它们之间的分布距离：

$$W_2(P,Q) = \Big(\inf_{f\in\mathcal{F}(P,Q)} \int \|\boldsymbol{x} - f(\boldsymbol{x})\|^2 dP(\boldsymbol{x})\Big)^{\frac{1}{2}}, \tag{2}$$

其中，下确界取遍 P与 Q之间$\mathcal{F}$中的所有传输方案。对于一维分布，存在闭式解来计算这样的最优传输映射 f^*，即 $f^*(x)$:$= F_P^{-1}(F_Q(x))$；F是与 P关联的累积分布函数（CDF）。

对于高维情形，切片 *Wasserstein* 距离 [2, 8, 35] 的度量形式化定义如下：

$$SW_2(P,Q) = \Big(\int_{\mathbb{S}^{d-1}} W_2^2\big(P^{\boldsymbol{\theta}}, Q^{\boldsymbol{\theta}}\big) d\boldsymbol{\theta}\Big)^{\frac{1}{2}}, \tag{3}$$

其中，$P^{\boldsymbol{\theta}}$通过函数$g^{\boldsymbol{\theta}}$: $\mathbb{R}^d \to \mathbb{R}$投影为 $P^{\boldsymbol{\theta}}$:$= g^{\boldsymbol{\theta}}(P)$，且$g^{\boldsymbol{\theta}}(x) = \boldsymbol{\theta}^\top x$。$\boldsymbol{\theta}$是$\mathbb{R}^d$中的单位向量，而$\mathbb{S}^{d-1}$是单位 d维超球面。由于具有正定性、对称性和三角不等式 [20, 21, 32, 36, 55], 我们将其用作高维子图特征分布的距离度量。

2.2 子图特征编码

2.2.1 构建子图特征分布。如图2(A)所示，如果两个用户在历史偏好方面被视为相似，那么他们应该拥有相似的行为，并且交互图模式有所重叠。基于这一直觉，假设每个用户的偏好都遵循一种未知的，

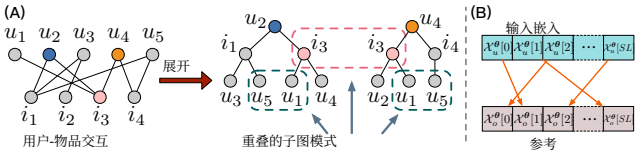

图2：（A）相似子图模式示意图。（B）输入与参考之间的映射过程。

相互独立且为 d维的概率测度，例如 P_u。我们假设，迄今观察到的 u 交互模式是从潜在分布 P_u 中采样得到的。因此，经验（离散）分布 $\widehat{P}_u$ 及其经验累积分布函数可表示为：

$$F_{\widehat{P}_u}(\boldsymbol{x}) = \frac{1}{L+1}\sum_{l=0}^{L} \delta(\boldsymbol{x} - \boldsymbol{v}_{ngh\to u}^{(l)}). \tag{4}$$

注意，我们初始化 $\boldsymbol{v}_{ngh\to u}^{0}(\cdot) = \boldsymbol{v}_u^0$此处 $\delta(\cdot)$ 在输入为零时返回 1，否则返回 0（从连续情形 $\int\delta(x)dx = 1$离散化）。不失一般性，这些经验分布具有代表性，即 $\widehat{P}_u \approx P_u$；因此，后文将 P_u 称为 $\widehat{P}_u$，以避免符号滥用。

2.2.2 实现 f^*。我们首先设置一个 d 维的参考分布 P_o，将其作为嵌入空间中的“原点”，以度量到任意输入的距离。P_o与随机特征嵌入相关联，例如 $\boldsymbol{v}_o^{(l)}$，如下：$F_{P_o}(\boldsymbol{x}) = \frac{1}{L+1}\sum_{n=0}^{L}\delta(\boldsymbol{x} - \boldsymbol{v}_o^{(l)})$。为在此类离散且 d 维的情形下实现最优传输映射 f^*，我们采用以下步骤。

我们首先通过投影函数$g^{\boldsymbol{\theta}}$对 P_o和 P_u进行分布切片。对于每一对分布切片 $P_u^{\boldsymbol{\theta}}$和 $P_o^{\boldsymbol{\theta}}$，令$\mathcal{X}_u^{\boldsymbol{\theta}}$收集其投影后的子图特征，记为$\mathcal{X}_u^{\boldsymbol{\theta}} = \{\boldsymbol{\theta}^\top \boldsymbol{v}_{ngh\to u}^{(l)}\}_{l=0}^{L}$（$\mathcal{X}_o^{\boldsymbol{\theta}} = \{\boldsymbol{\theta}^\top \boldsymbol{v}_o^{(l)}\}_{l=0}^{L}$也同理）。那么，相应的最优传输映射 $f^*(x) := F^{-1}_{P_u^{\boldsymbol{\theta}}}(F_{P_o^{\boldsymbol{\theta}}}(x))$可以定量地解释为：

$$f^*(x|\mathcal{X}_u^{\boldsymbol{\theta}}) = \text{argmin}_{x'\in\mathcal{X}_u^{\boldsymbol{\theta}}}\big(F_{P_u^{\boldsymbol{\theta}}}(x') = r\big), \text{where } r = F_{P_o^{\boldsymbol{\theta}}}(x). \tag{5}$$

此外，令 $\tau(x'|\mathcal{X}_u^{\boldsymbol{\theta}})$表示每个输入 x'按照$\mathcal{X}_u^{\boldsymbol{\theta}}$升序排列后的排名。我们可以替换术语$F_{P_u^{\boldsymbol{\theta}}}$，得到：

$$f^*(x|\mathcal{X}_u^{\boldsymbol{\theta}}) = \text{argmin}_{x'\in\mathcal{X}_u^{\boldsymbol{\theta}}}\big(\tau(x'|\mathcal{X}_u^{\boldsymbol{\theta}}) = \tau(x|\mathcal{X}_o^{\boldsymbol{\theta}})\big). \tag{6}$$

如图2(B)所示，公式（6）.实质上会在编码过程中对$\mathcal{X}_u^{\boldsymbol{\theta}}$的子图嵌入的不同层进行置换，从而能够进一步捕获并嵌入$\mathcal{X}_o^{\boldsymbol{\theta}}$到参考的距离。请注意，该距离采用公式（2）所示的$L_2$-范数形式，即又称欧几里得距离，这适用于召回向量化对象的场景，因为这类场景需要在嵌入空间中进行合理的距离度量。

2.2.3 实现 ***WSFE***。对于每一对分布切片，我们基于公式 (6) 的算法实现，按如下方式对其表示进行编码：

$$E_u^{\boldsymbol{\theta}} := \frac{1}{L+1}\Big\|_{l=0}^{L} f^*\Big(\boldsymbol{\theta}^\top \boldsymbol{v}_o^{(l)}|\mathcal{X}_u^{\boldsymbol{\theta}}\Big) - E_o^{\boldsymbol{\theta}} \text{ and } E_o^{\boldsymbol{\theta}} := \frac{1}{L+1}\Big\|_{l=0}^{L} \boldsymbol{\theta}^\top \boldsymbol{v}_o^{(l)}, \tag{7}$$

其中，$\|$ 表示拼接操作。根据公式（3）中的理论，下一步是为距离积分抽取无限多个投影，但这在实践中可能计算开销过大且难以实现。在本研究中，我们通过从$\mathbb{S}^{d-1}$进行 S次均匀采样，以蒙特卡罗近似实现该过程。因此，得到累积切片 Wasserstein

距离（即对公式（3）的近似），用于衡量参考 P_o 与原始输入特征分布 P_u 之间的距离：

$$SW_2(P_o, P_u) \approx \left(\frac{1}{S}\sum_{s=1}^{S} W_2^2(P_o^{\theta_s}, P_u^{\theta_s})\right)^{\frac{1}{2}}. \tag{8}$$

通过式（8）中的距离累积进行正则化后，我们的 *Wasserstein* 子图特征编码器（*WSFE*）最终定义如下：

$$\boldsymbol{E}_u := \frac{1}{S}\Big\|_{s=1}^{S} \boldsymbol{E}_u^{\boldsymbol{\theta}_s} \quad \text{and} \quad \boldsymbol{E}_o := \frac{1}{S}\Big\|_{s=1}^{S} \boldsymbol{E}_o^{\boldsymbol{\theta}_s}, \tag{9}$$

其中 $\boldsymbol{E}_u, \boldsymbol{E}_o \in \mathbb{R}^{SL+1}$ 请注意，在实践中，图卷积的数量 $L \le 4$ [11, 13, 18] 是一种常见设置，主要用于避免过平滑问题 [23]。此外，我们在 § 3.2.2 中的实证观察表明，设置 $S = 64$ 即可在可接受的计算开销下取得令人满意的模型性能。

2.3 理论分析

编码表示的一项重要预期是，它们能够反映子图特征分布的相似性/距离。我们在图3中通过以下定理对此进行说明：

定理1. 对于分布为 P_{u_i} 和 P_{u_j} 的任意用户子图特征 u_i 和 u_j，其编码表示满足：

① $\|\boldsymbol{E}_{u_i}\|_2 \approx SW_2(P_{u_i}, P_0)$.
② $\|\boldsymbol{E}_{u_i} - \boldsymbol{E}_{u_j}\|_2 \approx SW_2(P_{u_i}, P_{u_j})$.

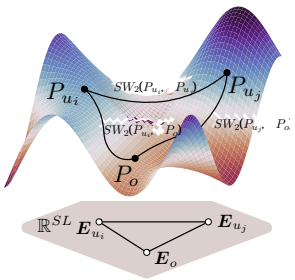

图3：瓦瑟斯坦到欧氏测地线。

证明。证明分为两部分。对于性质 ②，我们有：

$$\begin{aligned}
&\|\boldsymbol{E}_{u_i} - \boldsymbol{E}_{u_j}\|_2 \\
&= \left\|\Big\|_{s=1}^{S}(\boldsymbol{E}_{u_i}^{\boldsymbol{\theta}_s} - \boldsymbol{E}_{u_j}^{\boldsymbol{\theta}_s})\right\|_2 \\
&= \left\|\sum_{s=1}^{S}(\boldsymbol{E}_{u_i}^{\boldsymbol{\theta}_s} - \boldsymbol{E}_{u_j}^{\boldsymbol{\theta}_s})^2\right\|_2 \\
&= \left\|\frac{1}{S}\sum_{s=1}^{S}\frac{1}{L+1}\sum_{l=0}^{L}\left(f^*(\boldsymbol{\theta}_{\mathcal{T}_s} \boldsymbol{v}(l)_o, \mathbb{X}_{u_i}^{\boldsymbol{\theta}_s}) - f^*(\boldsymbol{\theta}_{\mathcal{T}_s} \boldsymbol{v}(l)_o, \mathbb{X}_{u_j}^{\boldsymbol{\theta}_s})\right)^2\right\|_2 \\
&\approx \left\|\int_{\mathbb{S}^{d-1}}\int_{\mathbb{R}}\left(F^{-1}_{P^{\boldsymbol{\theta}}_{u_i}}(F_{P^{\boldsymbol{\theta}}_o}(t)) - F^{-1}_{P^{\boldsymbol{\theta}}_{u_j}}(F_{P^{\boldsymbol{\theta}}_o}(t))\right)^2 dP_o^{\boldsymbol{\theta}}(t)d\boldsymbol{\theta}\right\|_2 \\
&= \left\|\int_{\mathbb{S}^{d-1}}\int_0^1 \left(F^{-1}_{P^{\boldsymbol{\theta}}_{u_i}}(r) - F^{-1}_{P^{\boldsymbol{\theta}}_{u_j}}(r)\right)^2 dr d\boldsymbol{\theta}\right\|_2.
\end{aligned} \tag{10}$$

设 $F^{-1}_{P^{\boldsymbol{\theta}}_{u_i}}(r) = x$，这意味着 $r = F_{P^{\boldsymbol{\theta}}_{u_i}}$。我们有：

$$\begin{aligned}
\|\boldsymbol{E}_{u_i} - \boldsymbol{E}_{u_j}\|_2 &= \left\|\int_{\mathbb{S}^{d-1}}\int_0^1 \left(x - F^{-1}_{P^{\boldsymbol{\theta}}_{u_j}}(F_{P^{\boldsymbol{\theta}}_{u_i}}(x))\right)^2 dP^{\boldsymbol{\theta}}_{u_j}(x)d\boldsymbol{\theta}\right\|_2 \\
&= \left\|\int_{\mathbb{S}^{d-1}} W_2^2(P^{\boldsymbol{\theta}}_{u_j}, P^{\boldsymbol{\theta}}_{u_i})d\boldsymbol{\theta}\right\|_2 = SW_2(P_{u_j}, P_{u_i}).
\end{aligned} \tag{11}$$

利用对称性，我们有 $SW_2(P_{u_i}, P_{u_j}) = SW_2(P_{u_j}, P_{u_i})$。然后对于参考 P_o，其编码表示可直接得到 $\boldsymbol{E}_o = \mathbf{0}$。因此，我们的证明如下：

$$\|\boldsymbol{E}_{u_i}\|_2 = \|\boldsymbol{E}_{u_i} - \boldsymbol{E}_o\|_2 \approx SW_2(P_{u_i}, P_o), \tag{12}$$

■

复杂度分析。 WSFE无需训练，可在骨干模型训练完成后立即在线使用。因此，WSFE 的复杂度为 $O(S(L+1)Md\log d)$，其中，计算 $\tau(\cdot)$ 的开销为 $O(d\log d)$。值得庆幸的是，该复杂度与输入数据规模呈线性关系，这表明编码可以在输入规模上完成。

表1：数据集的统计信息。

	MovieLens	Gowalla	Pinterest	Yelp2018	Kindle	AMZ-Book
#用户	6,040	29,858	55,186	31,668	115,652	52,643
#商品	3,952	40,981	9,916	38,048	98,729	91,599
#平均交互次数	165.60	34.31	26.52	49.31	15.85	56.69
#总交互次数	1,000,209	1,027,370	1,463,556	1,561,406	1,833,068	2,984,108

3 实验结果

我们评估 WSFE，旨在回答以下研究问题：**RQ1.** WSFE 如何提升最先进的推荐模型的用户细分性能？**RQ2.**不同的模型设置如何影响 WSFE 的性能？

3.1 实验设置

数据集。 我们从以下来源收集了六个经过广泛评估的公开数据集（包括其原始训练集/测试集划分）：MovieLens [5, 6,Gowalla [7, 10], Pinterest [9, 33], Yelp [49], Kindle [17, 52], 以及 Amazon-Book [1, 13]。数据集统计信息见表1。

评估协议。 用户细分所需的基本属性是用户间相似度测量。因此，给定一个查询用户，我们基于编码后的用户表示，将该任务视为对相似用户候选集进行排序。在本文中，我们根据用户交互过的重叠物品数量对相似用户进行排序；随后，将这些排序列表与学习模型推断出的 Top-K 结果进行比较。Recall@K 和 NDCG@K 是评估指标。

基线模型。 为展示 WSFE 的有效性，我们将其融入以下最先进的模型中。

(1) **LightGCN** [13] 是一种先进的基于图卷积网络的推荐模型，其结构更加简洁且功能强大。(2) **SGL** [46] 是一种具有代表性的基于图的模型，通过对比学习解决数据稀疏性问题。(3) **SimGCL** [51]是最先进的基于对比学习的推荐模型，直接在特征空间中进行简化的数据增强。(4) **NCL** [24] 是最先进的基于图的模型之一，通过对比邻域信息增强来提升性能。(5) **BUIR** [22] 是一种先进的模型，通过自举用户和物品表示实现协同过滤。(6) **DirectAU** [43] 是最新提出的模型，从对齐和均匀性的角度提升表示质量。

3.2 实证分析与讨论

3.2.1 整体性能（***RQ1***）。从表 2 中可以看出，

- 集成 WSFE 后，近期的推荐模型在所有数据集上的分群能力均有所提升。这不仅表明了我们方法的有效性，更重要的是，也验证了该方法对多种基于图的模型及不同数据集的普适性和灵活性。
- 我们注意到，模型在 MovieLens 数据集上的提升幅度大于在其他数据集上的提升幅度。一个主要原因是，MovieLens 用户的平均交互次数更多，如表1所示，达到 165.60 次，因此用户偏好分布更加复杂，而我们的 WSFE 能够充分利用这些丰富的信息来编码用户间相似性。

表 2：将 WSFE 应用于底层模型前后的实验结果（彩色效果最佳）。

数据集	模型	召回率@5	NDCG@5	召回率@20	NDCG@20	召回率@50	NDCG@50	召回率@100	NDCG@100
电影	LightGCN	0.10→0.18 (+80.00%)	0.36→0.61 (+69.44%)	0.37→0.69 (+86.49%)	0.36→0.65 (+80.56%)	0.85→1.68 (+97.65%)	0.62→1.20 (+93.55%)	1.63→3.04 (+86.50%)	0.97→1.83 (+88.66%)
	SGL	0.08→0.13 (+62.50%)	0.32→0.47 (+46.88%)	0.24→0.37 (+54.17%)	<escape>0.25→0.39 (+56.00%)</escape>	0.42→0.73 (+73.81%)	0.36→0.59 (+63.89%)	0.60→1.22 (+103.33%)	0.47→0.81 (+72.34%)
	SimGCL	0。90→1。02（+13。33%）	3。40→3。04（+13。82%）	2.13→3.77 (+77.00%)	2.30→3.60 (+56.52%)	3.10→8.56 (+176.13%)	2.92→5.99 (+105.14%)	4.45→13.71 (+208.09%)	3.35→8.58 (+156.12%)
	NCL	<escape>0.25→0.38(+52.00%)	0.84→1.30(+54.76%)</escape>	0.88→1.47 (+67.05%)	0.84→1.38 (+64.29%)	2.01→3.42 (+70.15%)	1.47→2.47 (+68.03%)	3。60→6.19 (+71.94%)	2.20→3.74 (+70.00%)
	BUIR	0.22→0.24 (+9.09%)	0.74→0.82 (+10.82%)	0.83→0.96 (+15.66%)	0.79→0.92 (+16.46%)	1.81→2.19 (+20.99%)	1.34→1.60 (+19.40%)	3.21→3.95 (+23.05%)	<escape>1.</escape>98→2</escape>.01 (</escape>+21 </escape>.21%)</escape>
	DirectAU	0.09→0.10 (+11.11%)	0.32→0.36 (+12.50%)	<escape>0.</escape>25→0</escape>.31 (</escape>+24</escape>.00%)</escape>	0.26→0.32 (+23.08%)	0.53→0.66 (+24.53%)	0.41→0.51 (+24.39%)	0.98→1.23 (+25.51%)	0.62→0.77 (+24.19%)
Gowalla	LightGCN	4.41→4.56（+3.40%）	6。59→6.52 (-1.06%)	8.75→9.26 (+5.83%)	6.64→6.80 (+2.41%)	13。30→14.05（+5.64%）	8.46→8.71 (+2.96%)	17.78→18.86 (+6.07%)	9.94→10.29 (+3.52%)
	SGL	4.96→5.30 (+6.85%)	7.58→7.62 (+0.53%)	9.85→10.66 (+8.22%)	<escape>7.</escape>53→7</escape>.86 (</escape>+4</escape>.38%)</escape>	15.13→16.68 (+10.24%)	9.66→10.23 (+5.90%)	20.70→22.72 (+9.76%)	11.49→12.19 (+6.09%)
	SimGCL	5.25→6.94 (+32.19%)	8.55→10.32 (+20.70%)	10.14→13.99 (+37.97%)	8.11→10.52 (+29.72%)	14.73→20.23 (+37.34%)	9.99→13.02 (+30.33%)	17.88→24.15 (+35.07%)	11.12→14.34 (+28.96%)
	NCL	4.65→5.01 (+7.53%)	8.31→8.71 (+4.81%)	9.41→10.20 (+8.40%)	<escape>7.</escape>81→8</escape>.34 (</escape>+6</escape>.79%)</escape>	13.78→15.11 (+9.65%)	9.71→10.43 (+7.42%)	18.16→20.02 (+10.24%)	11.22→12.11 (+7.93%)
	BUIR	2.94→3.07（+4.42%）	5.64→5.74 (+1.77%)	6.65→6.92 (+4.06%)	5.52→5.66 (+2.54%)	10.74→11.00 (+2.42%)	7.30→7.42 (+1.64%)	15.09→15.36 (+1.79%)	8.78→8.91 (+1.48%)
	DirectAU	5.03→5.41 (+7.55%)	7.99→8.22 (+2.88%)	10.24→10.98（+7.23%）	7.90→8.31 (+5.19%)	15.71→16.94 (+7.83%)	10.16→10.74 (+5.71%)	21.20→22.86 (+7.83%)	12.00→12.73 (+6.08%)
Pinterest	LightGCN	2.24→2.38 (+6.25%)	4.64→4.94 (+6.47%)	7.24→7.68 (+6.08%)	5.51→5.87 (+6.53%)	13.67→14.67 (+7.32%)	8.37→8.94 (+6.81%)	21.11→22.65 (+7.30%)	11.04→11.81 (+6.97%)
	SGL	3.93→3.92 (-0.25%)	7.58→7.58（0%）	11.21→11.35 (+1.25%)	8.57→8.63 (+0.70%)	19.62→19.86（+1.22%）	12.23→12.35（+0.98%）	28.16→28.44（+0.99%）	15.31→15.44（+0.85%）
	SimGCL	5.04→8.49（+68.45%）	9.28→14.86（+60.13%）	13.56→21.31（+57.15%）	10.33→16.20（+56.82%）	22.70→32.82（+44.58%）	14.31→21.21（+48.22%）	31.68→41.79（+31.91%）	17.54→24.37（+38.94%）
	NCL	4.10→4.63（+12.93%）	8.21→9.11（+10.84%）	11.48→13.02（+13.41%）	8.98→10.08（+12.25%）	19.55→22.32（+14.17%）	12.59→14.18（+12.63%）	27.85→31.76（+14.04%）	15.60→17.58（+12.69%）
	BUIR	1.16→1.22 (+5.17%)	2.55→2.64 (+3.53%)	4.09→4.31 (+5.38%)	3.16→3.28 (+3.80%)	8.29→8.77 (+5.79%)	5.05→5.26 (+4.16%)	13.34→14.16 (+6.15%)	6.92→7.23 (+4.48%)
	DirectAU	8.03→9.12 (+13.57%)	13.34→14.84 (+11.24%)	21.32→24.11 (+13.09%)	15.40→17.26 (+12.08%)	34.53→38.47 (+11.47%)	20.93→23.30 (+11.32%)	46.73→51.57 (+10.36%)	25.21→27.88 (+10.59%)
Yelp	LightGCN	2.13→2.37 (+11.27%)	1.31→1.64 (+25.19%)	2.16→2.72 (+25.93%)	<escape>1.94→2.39 (+23.20%)</escape>	4.07→5.23 (+28.50%)	2.88→3.62 (+25.69%)	6.47→8.37 (+29.37%)	3.87→4.88 (+26.10%)
	SGL	0.92→0.97 (+5.43%)	2.43→2.50 (+2.88%)	2.28→2.57（+12.72%）	2.14→2.33 (+8.88%)	4.06→4.67 (+15.02%)	3.02→3.36 (+11.26%)	6.21→7.18 (+15.62%)	3.91→4.38 (+12.02%)
	<escape>SimGCL	1.01→2.00(+98.02%)	2.72→5.24(+92.65%)</escape>	2.30→5.31 (+130.87%)	2.22→4.83 (+117.57%)	3.67→8.68 (+136.51%)	2.92→6.52 (+123.29%)	4.96→11.71 (+136.09%)	3.46→7.78 (+124.86%)
	NCL	1.16→1.41 (+21.55%)	3.43→4.06 (+18.37%)	2.91→3.65 (+25.43%)	2.87→3.52 (+22.65%)	4.76→6.13 (+28.78%)	3.83→4.79 (+25.07%)	6.74→8.88 (+31.75%)	4.67→5.91 (+26.55%)
	BUIR	0.53→0.53（+0%）	1.61→1.62 (+0.62%)	1.55→1.58（+1.94%）	<escape>1.</escape>49→</escape>.81 (</escape>+1 </escape>.31%)</escape>	2.76→2.84 (+2.90%)	2.13→2.17 (+1.88%)	4.19→4.33 (+3.34%)	<escape>2.74→2.79 (+1.82%)</escape>
	DirectAU	1.39→1.66 (+19.42%)	3.54→4.12 (+16.38%)	3.36→4.15 (+23.51%)	3.11→3.74 (+20.26%)	5.66→7.05 (+24.56%)	4.24→5.15 (+21.46%)	8.28→10.27 (+24.03%)	5.31→6.46（+21.66%）
Kindle	LightGCN	7.21→7.26 (+0.69%)	8.81→8.56 (-2.84%)	14.88→15.06 (+1.21%)	10.26→10.21 (-0.49%)	19.76→20.23 (+2.38%)	12.18→12.19 (+0.08%)	23.20→24.01 (+3.49%)	13.28→13.37 (+0.68%)
	SGL	7.90→7.94 (+0.51%)	9。63→9.33 (-3.12%)	16.93→17.58 (+3.84%)	11.53→11.61 (+0.69%)	23.32→24.37 (+4.50%)	13.99→14.19 (+1.43%)	27.87→29.43 (+5.60%)	15.41→15.74 (+2.14%)
	SimGCL	8.54→8.80 (+3.04%)	11.55→11.25 (-2.60%)	17.27→18.57 (+7.53%)	12.62→12.98 (+2.85%)	22.79→24.71 (+8.42%)	14.82→15.38 (+3.78%)	25.95→28.20 (+8.67%)	16.87→17.50 (+3.73%)
	NCL	9.26→9.66(+4.32%)	12.55→12.85(+2.39%)	18.25→19.42(+6.41%)	13.50→14.10 (+4.44%)	23.89→25.43 (+6.45%)	15.83→16.58 (+4.74%)	27.62→29.60 (+7.17%)	17.08→17.96 (+5.15%)
	BUIR	7.30→7.43 (+1.78%)	10.01→10.01 (0%)	15.50→15.67 (+1.10%)	11.28→11.34 (+0.53%)	20.64→20.98 (+1.65%)	13.46→13.58 (+0.89%)	24.20→24.72 (+2.15%)	14.68→14.85 (+1.16%)
	DirectAU	8.00→8.26 (+3.25%)	10.24→10.23 (-0.10%)	17.69→18.66 (+5.48%)	12.23→12.60 (+3.03%)	24.89→26.57 (+6.75%)	15.07→15.67 (+3.98%)	30.18→32.61 (+8.05%)	16.79→17.60 (+4.82%)
AMZ-Book	LightGCN	2.02→2.14 (+5.94%)	4.62→4.77 (+3.25%)	5.15→5.65 (+9.71%)	4.48→4.82 (+7.59%)	8.13→9.26 (+13.90%)	5.92→6.49 (+9.63%)	11.05→12.78（+15.66%）	7.06→7.85（+11.19%）
	SGL	2.47→2.59 (+4.86%)	<escape>4.</escape>57→4</escape>.82 (</escape>+2 </escape>.00%)</escape>	6.06→6.64 (+9.57%)	5.30→5.65 (+6.60%)	9.46→10.66（+12.68%）	6.94→7.52 (+8.36%)	12.64→14.39 (+13.84%)	8.18→8.95 (+9.41%)
	SimGCL	2.72→3.09 (+13.60%)	6.40→7.02 (+9.69%)	6.13→7.14 (+16.48%)	5.64→6.42 (+13.83%)	8.69→10.35 (+19.10%)	6.19→7.99 (+29.08%)	10.55→12.76 (+20.95%)	7.68→8.95 (+16.54%)
	NCL	2.63→2.97 (+7.53%)	6.49→7.18 (+4.81%)	6.30→7.32 (+8.40%)	5.81→6.61 (+6.79%)	9.53→11.35 (+9.65%)	7.44→8.58 (+7.42%)	12.44→15.10 (+10.24%)	8.58→10.07 (+7.93%)
	BUIR	1.43→1.46 (+2.10%)	<escape>5.</escape>78→5</escape>.82 (</escape>+1 </escape>.04%)</escape>	3.95→4.00（+1.27%）	<escape>3.60→3.63 (+0.83%)</escape>	6.30→6.40 (+1.59%)	4.79→4.83 (+0.84%)	8.45→8.55 (+1.18%)	5.66→5.70 (+0.71%)
	DirectAU	2.88→3.13 (+8.68%)	6.55→7.00 (+6.87%)	6。88→7.63 (+10.90%)	6.12→6.68 (+9.15%)	10.48→11.80 (+12.60%)	7.89→8.70 (+10.27%)	13.68→15.54 (+13.60%)	9.16→10.17 (+11.03%)

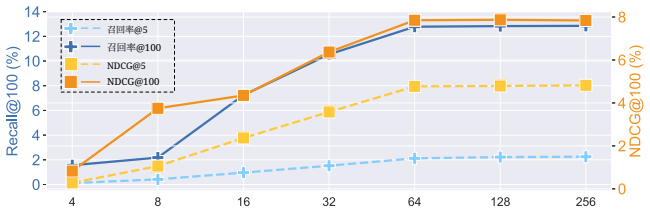

图4：改变 S 值的实验结果。

• 此外，采用 WSFE 后，基于对比学习的模型（例如 SGL [46], SimGCL [51], 和 NCL [24], ）通常能取得更大的提升。这是因为数据增强技术（无论是针对原始数据还是潜在特征）随后为 WSFE 的发挥提供了嵌入丰富化。

3.2.2 切片数量的影响 S（***RQ2A***）。鉴于 LightGCN 性能出色且稳定，我们在 AMZ-Book 数据集上将其用作骨干模型，以展示模型分析。我们依次改变 S 的值，并将结果绘制于图4中。我们发现，将 S 从 4 增至 64 对模型性能的影响更大，这很直观，因为此举能够得到更准确、更细粒度的累积近似。然而另一方面，持续增加 S 也会带来更多计算开销和内存压力。因此，将 S 设为 64 是一个折中选择，既能带来正向提升，也在模型性能与资源消耗之间实现了实用的权衡。

3.2.3 降维（***RQ2B***）。在评估过程中，我们注意到一些模型会遇到“内存不足”问题。为解决这一问题，我们对式（7）中的逐层嵌入进行聚合，将总维度从 S（L+1）降至 S。

表3：聚合选择。

聚合器指标	拼接 K=5	拼接 K=100	*Sum* K=5	*Sum* K=100	*Max* K=5	*Max* K=100
Recall@K	**2.14**	12.78	2.12	**12.81**	1.77	11.27
NDCG@K	4.77	**7.85**	**4.78**	7.77	4.31	7.01

从表3中我们注意到，求和令人惊讶地展现出与原始拼接操作相当的性能。这表明尽管拼接在理论支持下具有更完整的表示编码，但求和适用于计算资源有限的场景，可用于降维。

4 结论与未来拓展

在这项工作中，我们提出了 WSFE，用于对表示进行编码，以实现协同过滤中的有效用户细分。大量实验表明，所提方法有效，并且适用于各种模型部署。对于未来工作，我们指出以下三个主要方向：

(1) 鉴于 § 3.2.1 中的实证发现，值得探索在子图特征领域运用对比学习技术 [28, 30, 54, 56]，以进一步改进模型。(2) 值得研究如何将我们的免训练模型应用于信息检索和自治数据库管理等其他场景 [14–16, 26, 27, 34, 38]。(3) 我们计划设计无监督正则化机制，使 WSFE 与骨干模型能够联合优化，甚至相互增强，以用于多任务学习。

致谢

Yankai Chen、Yifei Zhang、Menglin Yang、Zixing Song 和 Irwin King 的部分研究经费由中国国家重点研发计划（编号：2018AAA0100204）以及中国香港特别行政区研究资助局（RGC GRF 2151185；CUHK 14222922）提供。Chen Ma 获得了香港城市大学启动基金（编号：9610564）和战略研究基金（编号：7005847）的资助。

参考

[1] 亚马逊。2015年。数据集。https://github.com/gusye1234/LightGCN-PyTorch/tree/master/data/amazon-book.[2] Nicolas Bonneel、Julien Rabin、Gabriel Peyré 和 Hanspeter Pfister。2015年。测度的切片 Wasserstein 重心与 Radon Wasserstein 重心。JMIV 51, 1（2015），22–45页。[3] Yankai Chen、Yixiang Fang、Yifei Zhang 和 Irwin King。2023年。用于汉明空间中高效且有效的 Top-N 搜索的二部图卷积哈希。WebConf。[4] Yankai Chen、Huifeng Guo、Yingxue Zhang、Chen Ma、Ruiming Tang、Jingjie Li 和 Irwin King。2022年。通过多维量化强化学习学习二值化图表示，以实现 Top-k 推荐。载于 SIGKDD。[5] Yankai Chen、Menglin Yang、Yingxue Zhang、Mengchen Zhao、Ziqiao Meng、Jianye Hao 和 Irwin King。2022年。利用双曲几何对无标度图进行建模，以实现知识感知推荐。载于 WSDM。94–102页。[6] Yankai Chen、Yaming Yang、Yujing Wang、Jing Bai、Xiangchen Song 和 Irwin King。2022年。结合协同指导的注意力知识感知图卷积网络，用于个性化推荐。载于 ICDE。[7] Eunjoon Cho、Seth A Myers 和 Jure Leskovec。2011年。友谊与移动性：基于位置的社交网络中的用户移动。载于 SIGKDD。1082–1090页。[8] Ishan Deshpande、Yuan-Ting Hu、Ruoyu Sun、Ayis Pyrros、Nasir Siddiqui、Sanmi Koyejo、Zhizhen Zhao、David Forsyth 和 Alexander G Schwing。2019年。最大切片 Wasserstein 距离及其在生成对抗网络中的应用。载于 CVPR。10648–10656页。[9] Xue Geng、Hanwang Zhang、Jingwen Bian 和 Tat-Seng Chua。2015年。学习图像和用户特征，以实现社交网络中的推荐。载于 ICCV。4274–4282页。[10] Gowalla。2011年。数据集。https://github.com/gusye1234/LightGCN-PyTorch/tree/ master/data/gowalla。[11] William L Hamilton、Rex Ying 和 Jure Leskovec。2017年。大规模图上的归纳式表示学习。载于 NeurIPS。1025–1035页。[12] Bowei He、Xu He、Yingxue Zhang、Ruiming Tang 和 Chen Ma。2023年。用于流式推荐的动态可扩展图卷积。WebConf。[13] Xiangnan He、Kuan Deng、Xiang Wang、Yan Li、Yongdong Zhang 和 Meng Wang。2020年。LightGCN：简化并增强用于推荐的图卷积网络。载于 SIGIR。639–648页。[14] Xuming Hu、Fukun Ma、Chenyao Liu、Chenwei Zhang、Lijie Wen 和 Philip S Yu。2021年。通过增量元自训练实现半监督关系抽取。载于 EMNLP：Findings。[15] Xuming Hu、Lijie Wen、Yusong Xu、Chenwei Zhang 和 Philip S. Yu。2020年。Self-ORE：用于开放关系抽取的自监督关系特征学习。载于 EMNLP。3673–3682页。[16] Xuming Hu、Chenwei Zhang、Yawen Yang、Xiaohe Li、Li Lin、Lijie Wen 和 Philip S. Yu。2021年。用于低资源关系抽取的梯度模仿强化学习。载于 EMNLP。2737–2746页。[17] 亚马逊Kindle。2015年。数据集。https://github.com/Coder-Yu/SELFRec/tree/main/ dataset/amazon-kindle。[18] Thomas N Kipf 和 Max Welling。2017年。基于图卷积网络的半监督分类。ICLR。[19] Soheil Kolouri、Navid Naderializadeh、Gustavo K Rohde 和 Heiko Hoffmann。2020年。用于图学习的 Wasserstein 嵌入。ICLR。[20] Soheil Kolouri、Kimia Nadjahi、Umut Simsekli、Roland Badeau 和 Gustavo Rohde。2019年。广义切片Wasserstein距离。NeurIPS 32。[21] Soheil Kolouri、Yang Zou 和 Gustavo K Rohde。2016年。概率分布的切片Wasserstein核。载于 CVPR。5258–5267页。[22] Dongha Lee、SeongKu Kang、Hyunjun Ju、Chanyoung Park 和 Hwanjo Yu。2021年。通过自举学习用户和物品表示，以实现单类协同过滤。载于 SIGIR。317–326页。[23] Guohao Li、Matthias Muller、Ali Thabet 和 Bernard Ghanem。2019年。DeepGCNs：图卷积网络能否像卷积神经网络一样深入？载于 CVPR。9267–9276页。[24] Zihan Lin、Changxin Tian、Yupeng Hou 和 Wayne Xin Zhao。2022年。通过邻域增强对比学习改进图协同过滤。载于 WebConf。2320–2329页。[25] Guang Ling、Haiqin Yang、Irwin King 和 Michael R Lyu。2012年。用于协同过滤的在线学习。载于 IJCNN。IEEE，1–8页。[26] Aiwei Liu、Xuming Hu、Li Lin 和 Lijie Wen。2022。通过迭代学习模式链接图增强语义的文本到 SQL 解析。载于 SIGKDD。1021–1030。[27] Aiwei Liu、Xuming Hu、Lijie Wen 和 Philip S Yu。2023。对 ChatGPT 零样本文本到 SQL 能力的全面评估。arXiv 预印本 arXiv:2303.13547（2023）。[28] Shuliang Liu、Xuming Hu、Chenwei Zhang、Shu'ang Li、Lijie Wen 和 Philip S. Yu。2022。HiURE：用于无监督关系抽取的层次化示例对比学习。载于 NAACL-HLT。5970–5980。[29] Fangyuan Luo、Jun Wu 和 Tao Wang。2022。基于隐式反馈的快速 Top-N推荐离散列表式个性化排序。（2022）。[30] Yueen Ma、Zixing Song、Xuming Hu、Jingjing Li、Yifei Zhang 和 Irwin King。2023。用于概念相关性估计的图组件对比学习。载于 AAAI。[31] MovieLens。2003。数据集。https://grouplens.org/datasets/movielens/1m/。[32] Navid Naderializadeh、Joseph F Comer、Reed Andrews、Heiko Hoffmann 和 Soheil Kolouri。2021 34, 3389 3400 。通过切片Wasserstein嵌入进行池化。NeurIPS – 。[33] Pinterest。2015。数据集。https://sites.google.com/site/xueatalphabeta/dataset- 1/pinterest_iccv。[34] Zexuan Qiu、Qinliang Su、Jianxing Yu 和 Shijing Si。2022。利用对比乘积量化端到端地优化和量化 BERT嵌入，实现高效文档检索。EMNLP（2022）。[35] Julien Rabin、Gabriel Peyré、Julie Delon 和 Marc Bernot。2011。Wasserstein重心及其在纹理混合中的应用。载于 SSVM。Springer，435–446。[36] Xin Shen、Kyungdon Joo 和 Jean Oh。2023。FishRecGAN：一种用于鱼眼校正和标定的端到端生成对抗网络。arXiv（2023）。[37] Xin Shen、Jiaying Shi、Sungro Yoon、Jon Katzur、Hanbo Wang、Jim Chan 和 Jin Li。2023。基于客户购物意图学习个性化推荐。arXiv（2023）。[38] Xin Shen、Xiaonan Zhao 和 Rui Luo。2023。语义嵌入深度神经网络：提升多标签图像分类性能的通用方法。arXiv（2023）。[39] Xin Shen、Yan Zhao、Sujan Perera、Yujia Liu、Jinyun Yan 和 Mitchell Goodman。2023。利用客户表示学习个性化页面内容排序。arXiv（2023）。[40] Zixing Song、Ziqiao Meng、Yifei Zhang 和 Irwin King。2021。图结构数据的半监督多标签学习。载于 CIKM。[41] Zixing Song、Yifei Zhang 和 Irwin King。2022。面向鲁棒半监督节点分类的最优非对称图结构。载于 SIGKDD。1656–1665。[42] Cédric Villani。2009。最优传输：旧问题与新进展，第338卷。Springer。[43] Chenyang Wang、Yuanqing Yu、Weizhi Ma、Min Zhang、Chong Chen、Yiqun Liu 和 Shaoping Ma。2022。协同过滤中的表示对齐与均匀性。载于 SIGKDD。1816–1825。[44] 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。[45] Jun Wu、Fangyuan Luo、Yujia Zhang 和 Haishuai Wang。2020。半离散矩阵分解。IEEE Intelligent Systems 35，5（2020），73–83。[46] Jiancan Wu、Xiang Wang、Fuli Feng、Xiangnan He、Liang Chen、Jianxun Lian 和 Xing Xie。2021。用于推荐的自监督图学习。载于 SIGIR。726–735。[47] Menglin Yang、Zhihao Li、Min Zhou、Jiahong Liu 和 Irwin King。2022。HICF：双曲信息协同过滤。载于 SIGKDD。2212–2221。[48] Menglin Yang、Min Zhou、Jiahong Liu、Defu Lian 和 Irwin King。2022。HRCF：通过双曲几何正则化增强协同过滤。载于 WebConf。2462–2471。[49] Yelp2018。2018。数据集。https://github.com/gusye1234/LightGCN-PyTorch/tree/master/data/yelp2018。[50] Rex Ying、Ruining He、Kaifeng Chen、Pong Eksombatchai、William L Hamilton 和 Jure Leskovec。2018。面向网络规模推荐系统的图卷积神经网络。载于 SIGKDD。974–983。[51] Junliang Yu、Hongzhi Yin、Xin Xia、Tong Chen、Lizhen Cui 和 Quoc Viet Hung Nguyen。2022。图增强是否必要？用于推荐的简单图对比学习。载于 SIGIR。1294–1303。[52] Junliang Yu、Hongzhi Yin、Xin Xia、Tong Chen、Jundong Li 和 Zi Huang。2022。推荐系统的自监督学习：综述。arXiv（2022）。[53] Xinni Zhang、Yankai Chen、Cuiyun Gao、Qing Liao、Shenglin Zhao 和 Irwin King。2022。采用个性化特征引用的知识感知神经网络，用于冷启动推荐。arXiv（2022）。[54] Yifei Zhang、Yankai Chen、Zixing Song 和 Irwin King。2023。对比式跨尺度图知识协同。arXiv（2023）。[55] Yifei Zhang 和 Hao Zhu。2020。用于文档检索的离散Wasserstein自编码器。载于 ICASSP。IEEE，8159–8163。[56] Yifei Zhang、Hao Zhu、Zixing Song、Piotr Koniusz 和 Irwin King。2022。COSTA：用于图对比学习的协方差保持特征增强。I