

# 融合动态稀疏性与异构知识蒸馏的 Top-k 推荐算法

付诗棋 朱金侠 许琪晨 杜泽宇

辽宁工程技术大学软件学院 辽宁 葫芦岛 125105

(815074008@qq.com)

**摘要** 当前的推荐算法主要侧重于利用深度学习技术提高推荐精度,进而为用户推荐其感兴趣的内容集合。然而,这类方法得到的推荐结果往往计算开销大,模型冗余度高,不适合资源受限场景。针对上述问题,提出一种融合动态稀疏性与异构知识蒸馏的协同优化框架(DySparseHKD),构建轻量化的推荐模型,在减少参数数量的同时保留关键特征。提出基于交互冗余的动态稀疏率分配方法,捕获更高效的参数配置,利用教师模型训练轨迹实现渐进式知识传递,缓解异构模型间的知识差异,根据学生模型当前的学习状态动态调整知识迁移粒度,提高迁移效率;最后,通过联合优化目标实现模型复杂度与推荐性能的深度解耦。在3个真实的数据集上进行相关的实验,结果表明所提模型在具有更低复杂度的同时实现了模型效率与推荐效果的有机融合。

**关键词:** 动态稀疏技术;异构知识蒸馏;知识迁移;协同过滤;轻量化推荐模型

**中图分类号** TP301

## Dynamic Sparsity and Heterogeneous Knowledge Distillation for Top-k Recommendation

FU Shiqi, ZHU Jinxia, XU Qichen and DU Zeyu

School of Software, Liaoning Technical University, Huludao, Liaoning 125105, China

**Abstract** Current recommendation algorithms mainly focus on using deep learning techniques to enhance recommendation accuracy, thereby providing users with a collection of content they are interested in. However, the recommendation results obtained by such methods often have high computational costs and model redundancy, making them unsuitable for resource-constrained scenarios. To address these issues, a collaborative optimization framework (DySparseHKD) that integrates dynamic sparsity and heterogeneous knowledge distillation is proposed. This framework builds a lightweight recommendation model that reduces the number of parameters while retaining key features. A dynamic sparsity rate allocation method based on interaction redundancy is proposed to capture more efficient parameter configurations. The training trajectory of the teacher model is utilized to achieve progressive knowledge transfer, alleviating the knowledge gap between heterogeneous models. The knowledge transfer granularity is dynamically adjusted according to the current learning state of the student model to improve transfer efficiency. Finally, the deep decoupling of model complexity and recommendation performance is achieved through joint optimization objectives. Experiments on three real datasets show that the proposed model achieves an organic integration of model efficiency and recommendation effect while maintaining lower complexity.

**Keywords** Dynamic sparse technology, Heterogeneous knowledge distillation, Knowledge transfer, Collaborative filtering, Lightweight recommendation model

深度学习与大规模计算资源的快速发展,推动了推荐系统与大型语言模型(Large Language Model, LLM)的广泛应用,但二者在模型效率与性能优化方面仍面临严峻挑战。现有推荐算法<sup>[1-3]</sup>主要侧重于通过集成异构模型(如矩阵分解<sup>[4]</sup>、图神经网络<sup>[5]</sup>)提升推荐准确性,但计算开销大,模型冗余度高,不适合资源受限场景。动态稀疏化技术通过冗余分析实现高效压缩,在保持模型性能的同时显著降低计算和存储成本,成为 LLM 轻量化的关键技术<sup>[6]</sup>,但在个性化推荐领

域尚未得到充分探索。推荐领域中用户-项目交互数据的高维稀疏性与冗余噪声导致模型推理成本激增,部署成本高。此外,知识蒸馏(Knowledge Distillation, KD)通过将复杂教师模型的知识迁移至轻量学生模型实现压缩,然而现有方法<sup>[7-8]</sup>局限于同构模型间的知识传递,而异构教师<sup>[9]</sup>因损失函数差异导致蒸馏效率显著下降。文献[10]发现教师模型早期训练阶段的预测知识更易被学生学习,而随着训练推进,教师的项目排序复杂度增加,导致跨模型知识迁移困难。

基金项目:国家重点研发计划项目(2018YFB1402901);国家自然科学基金项目(61772249);辽宁省教育厅一般项目(LJ2019QL017)

This work was supported by the National Key Research and Development Program of China(2018YFB1402901), National Natural Science Foundation of China(61772249) and General Project of the Education Department of Liaoning Province(LJ2019QL017).

通信作者:朱金侠(zjx15804291750@163.com)

本文通过交互冗余分析量化模型,识别非关键连接并进行结构化稀疏,实现高效压缩,突破数据稀疏性与模型准确性的权衡边界;同时,设计动态异构知识蒸馏策略提升排序能力,利用教师模型的训练轨迹实现从易到难的知识传递,缓解异构模型间的知识差异;其次,利用表示互信息(Representation Mutual Information, RMI)量化学习特征的有效性,衡量在知识蒸馏过程中师生模型的排名一致性;此外,根据学生模型当前的学习状态动态调整知识迁移粒度,构建从粗粒度全局结构特征到细粒度局部排序模式的渐进式学习路径,提高迁移效率,降低模型偏差;最后,通过联合优化目标平衡模型效率与推荐精度间的深度解耦。

## 1 相关工作

随着深度学习和大规模计算资源的迅速发展,与模型压缩和性能优化紧密相关的稀疏化技术和知识蒸馏方法研究逐渐增多。稀疏化技术通过减少或调整网络连接直接降低参数量,重点在于识别并保留关键连接。知识蒸馏则通过知识迁移将复杂模型的能力转移到轻量级模型,重点在于提取并迁移有效知识。

### 1.1 稀疏化技术

稀疏化技术是模型轻量化的重要方法,但现有技术使用权重剪枝、结构化剪枝、稀疏训练、彩票假说等方法压缩模型,缺乏动态调整机制。文献[11]提出“深度压缩”框架,通过剪枝、量化和编码三步压缩深度神经网络,减少模型参数量,但静态剪枝需预训练模型,影响模型容量。文献[12]通过结构化剪枝(如通道剪枝),直接移除整个神经元或卷积核,对硬件友好,可加速推理,但是灵活性低,可能损失重要特征。文献[13]从随机稀疏掩码初始化到动态调整稀疏模式,提出动态稀疏训练(Dynamic Sparse Training, SET),允许网络在训练过程中动态调整稀疏连接模式,但训练不稳定,需复杂调度策略。文献[14]提出彩票假说,发现子网络在随机初始化时即可通过剪枝达到与原网络相当的性能,可为高效模型训练提供理论依据,但是须重复训练多次,计算成本高。

### 1.2 知识蒸馏

知识蒸馏<sup>[15]</sup>的核心是通过“教师-学生”架构将大型模型(教师)的知识迁移到小型模型(学生),优化学生模型性能,以解决模型的压缩与部署问题。文献[16]将知识蒸馏引入推荐系统去偏领域,在均匀数据上训练教师模型,在原有训练数据上训练学生模型,使学生模型汲取来自于教师模型的无偏信息,建立一个用于反事实去偏的通用知识蒸馏框架。此后,知识蒸馏开始逐步应用于推荐系统领域,并由经典单教师蒸馏发展到多教师蒸馏、进阶蒸馏等。

单教师蒸馏<sup>[17]</sup>实现简单,训练成本低,能有效压缩模型规模,适用于资源受限场景,但学生模型的性能受限于单一教师的知识,若教师与学生模型架构差异大(如 RNN 与 MLP),导致中间特征难以对齐,知识迁移效率将降低。多教师蒸馏<sup>[18]</sup>通过集成多个教师模型的优势,解决单教师知识单一的问题,其关键方法为投票策略、动态权重分配、知识合并。文献[19]提出多教师和多学生蒸馏的集成蒸馏方法,以弥补学习能力的不足。文献[20]通过输出层平均与中间层相异性约束整合多教师网络,提高模型预测速度。多教师蒸馏减少了

单教师模型的偏差,提升了鲁棒性,支持多模态或多任务知识融合,但集成策略设计复杂,计算成本高,若教师模型质量参差不齐,可能引入噪声。进阶蒸馏<sup>[21]</sup>应运而生,突破了传统蒸馏的局限性。自蒸馏<sup>[22]</sup>无需额外的教师模型,降低了依赖,但可能陷入局部最优。跨模态蒸馏<sup>[23]</sup>可挖掘多源数据关联性,提升推荐多样性,但需复杂特征映射,增加了训练难度。

### 1.3 Top-k 推荐

Top- $k$  推荐任务通过矩阵分解、协同过滤、图神经网络等方法生成高质量列表。文献[24]通过优化嵌入学习、增强关系捕捉、精准权重分配和合理分数生成,解决了传统方法在 Top- $k$  推荐中对邻居重要性建模不足的问题,但对稀疏交互的适应性不足。文献[25]通过结构化约束条件平衡推荐准确性与多样性,有效提升了用户兴趣覆盖广度与推荐新颖性,适用于资源分配不均的复杂推荐场景,但灵活性受限。文献[26]提出一种高效且负责任的框架,通过混合任务分配策略结合传统推荐系统与大型语言模型,针对弱用户群体设计上下文学习机制,在控制计算成本的同时增强推荐的鲁棒性与公平性,但依赖模型资源。

综上所述,推荐系统研究已从稀疏化、知识蒸馏和 Top- $k$  推荐等多个维度展开。文献[40]提出的 DSL (Dynamic Sparse Learning) 模型实现了稀疏度动态调整与全生命周期效率优化,但未涉及知识迁移对稀疏模型精度的补偿机制。进一步,文献[41]提出的模型探索了图结构稀疏化与多层次的知识蒸馏的结合,实现了结构压缩和性能保持,然而稀疏评分机制仅针对图结构而非模型参数本身,导致模型仍存在参数冗余问题。针对上述问题,本文提出一种新的 Top- $k$  推荐算法,该框架创新性地将动态稀疏技术与异构知识蒸馏相结合,在实现模型轻量化的同时保障了精度的稳定性。通过注意力机制筛选用户-项目交互图中的关键边,将动态稀疏技术延伸至模型核心参数层,自适应调整稀疏模式。

需要指出的是,虽然文献[10]也探讨了知识蒸馏在推荐系统中的应用,但本文方法与之有以下两点不同。(1)稀疏策略不同:本文采用动态稀疏训练机制,在训练过程中自适应调整稀疏模式,保留关键特征并减少冗余连接;而文献[10]主要关注知识迁移过程,未充分考虑模型结构的动态优化。(2)知识迁移机制不同:本文设计了基于教师训练轨迹的课程学习机制和动态调整的知识粒度转移策略,实现了异构模型间的高效知识迁移;文献[10]虽也利用教师模型训练轨迹,但未考虑知识粒度的动态调整,限制了知识迁移的适应性和灵活性。

## 2 算法

### 2.1 问题定义

**定义 1**(Top- $k$  推荐) 给定用户集  $U$ 、项目集  $I$  以及用户-项目的隐式交互集  $\mathcal{R}$ , 推荐模型  $f: U \times I \rightarrow \mathcal{R}$  用于预测用户-项目对之间的交互分数,基于此为每个用户  $u$  生成  $k$  个未观测项目的有序推荐列表,实现 Top- $k$  推荐。

**定义 2**(动态稀疏性) 旨在自适应地识别和保留关键的用户-项目之间的交互信息,通过可学习的稀疏率  $s(u, i) \in [0, 1]$  动态调整每个交互的重要程度。根据用户行为模式和项目特征的动态变化,自适应地确定稀疏化程度,在保证模型

性能的同时实现计算资源的高效利用。

**定义 3(异构知识蒸馏)** 给定教师模型集合  $F = \{f^1, f^2, \dots, f^M\}$ , 通过知识蒸馏技术将其压缩为轻量级学生模型  $f$ , 以降低推理阶段的计算成本。学生模型在保持推荐性能的同时, 显著减少了推理的计算开销, 适用于实时服务和资源受限环境。本文所设计的异构知识蒸馏不依赖特定教师模型的结构, 可灵活适配任意教师模型。

## 2.2 解决方案

本文设计了一种融合动态稀疏性与异构知识蒸馏的协同优化框架, 动态稀疏性与异构知识蒸馏的协同设计能实现参数压缩与特征保留、跨阶段知识迁移、模型效率与推荐性能的协同。通过动态稀疏剔除无效连接并压缩参数, 结合异构知识蒸馏迁移关键特征。利用训练轨迹实现渐进式知识传递, 联合优化目标平衡多损失, 实现模型效率与推荐性能的深度

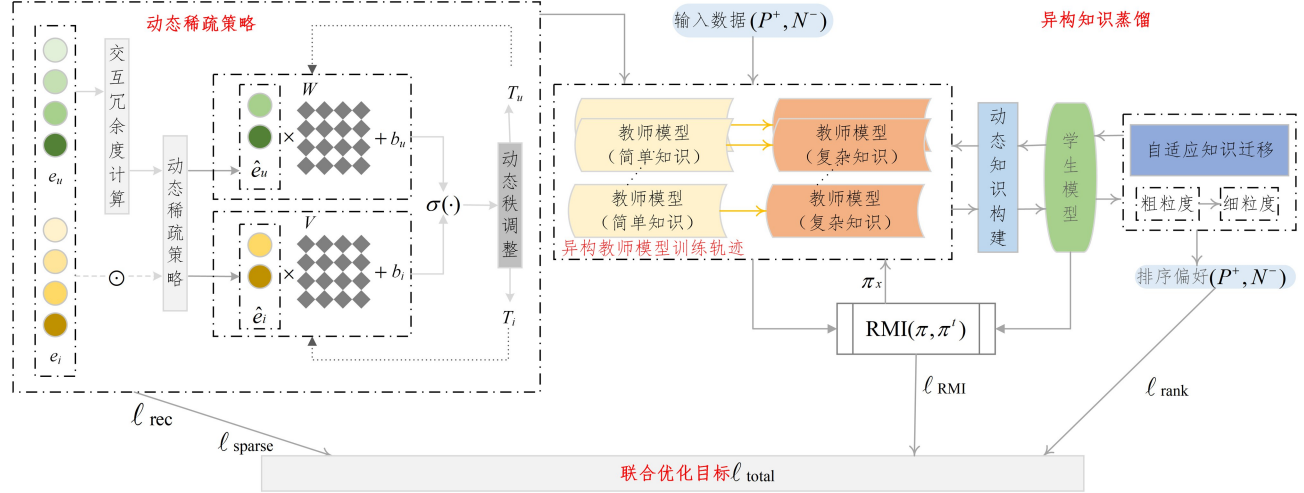


图 1 融合动态稀疏性与异构知识蒸馏的协同优化框架

Fig. 1 Collaborative optimization framework integrating dynamic sparsity and heterogeneous knowledge distillation

### 2.2.1 动态稀疏策略

推荐算法中, 高冗余度会导致模型过拟合及推理成本激增等问题<sup>[27]</sup>。本文通过余弦相似度衡量用户-项目交互的冗余程度, 识别可稀疏化的不重要交互, 在保证推荐精度的同时实现模型轻量化。具体如下:

$$D(u, i) = \sum_{u \in N(u)} \frac{e_u^T e_i}{\|e_u\| \|e_i\|} \quad (1)$$

其中,  $e_u$  为用户  $u$  的嵌入向量,  $e_i$  为项目  $i$  的嵌入项目,  $N(u)$  为用户  $u$  的邻居集合。

本文通过结合冗余度与项目流行度动态调整稀疏率, 对冗余度高/流行度低的项目提高压缩概率, 在保留关键交互的同时提升计算效率。计算如下:

$$s_{u,i} = \sigma(\xi D(u, i) + \zeta Pop(i)), \sum_i s_{u,i} = S \quad (2)$$

其中,  $\sigma(\cdot)$  为 Sigmoid 激活函数;  $Pop(i)$  为项目  $i$  的流行度, 本文假设项目被购买/被点击的次数越多就代表越流行; 超参数  $\xi, \zeta \in [0, 1]$ , 用于平衡冗余两者间的权重;  $S$  为全局稀疏预算。

结合式(1)和式(2), 根据信息论可知冗余度高且流行度低的交互携带的有效信息量较少, 压缩此类连接可在保持信息完整性的前提下降低模型复杂度。

通过式(2)对用户、项目的原始嵌入进行稀疏处理, 过滤冗余交互, 生成轻量化的稀疏嵌入, 平衡推荐模型的存储成本与特征表达能力。计算如下:

解耦。协同优化框架如图 1 所示, 共分 3 步。第 1 步, 动态稀疏策略基于用户与项目的嵌入向量计算交互冗余度, 精准识别可稀疏化的非关键交互, 再结合冗余度与项目流行度动态调整稀疏率, 平衡信息保留与计算效率, 最终通过低秩矩阵实现模型压缩, 在减少参数数量的同时维持模型性能。第 2 步, 异构知识蒸馏。在训练的初始阶段, 教师模型先从简单的知识开始蒸馏, 随着训练轨迹的推进, 逐步递增知识难度, 减小小学生模型因直接学习异构教师模型知识而产生的巨大差异, 再利用排名互信息(RMI)衡量在知识蒸馏过程中师生模型的排名一致性, 确保知识传递的准确性与有效性。第 3 步, 联合优化目标。通过整合稀疏损失、重构损失、互信息损失、自适应蒸馏损失, 构建全面优化目标函数, 平衡推荐准确性、模型压缩效率与知识蒸馏效果, 最终实现 Top-k 推荐。

$$\hat{e}_u = s_u \odot e_u \quad (3)$$

$$\hat{e}_i = s_i \odot e_i \quad (4)$$

其中, 元素级乘法  $\odot$  用于保留关键特征, 抑制冗余。

通过低秩矩阵分解减少冗余参数数量, 降低模型复杂度, 捕获用户的潜在兴趣和项目的潜在特征, 进而预测用户-项目交互得分, 为模型后续优化构建完整的推荐链路。计算如下:

$$\phi_{u,i} = \sigma(\hat{e}_u^T W + b_u) \cdot \sigma(\hat{e}_i^T V + b_i) \quad (5)$$

其中,  $\sigma(\cdot)$  为激活函数, 用于增强模型的非线性表达能力;  $W \in \mathbb{R}^{d \times T}$ ,  $V \in \mathbb{R}^{d \times T}$  为投影矩阵, 通过低秩映射压缩嵌入维度, 降低模型复杂度。

为优化用户-项目交互的低秩矩阵分解, 通过最小化重构误差与正则项动态确定最优秩, 平衡模型拟合能力与参数稀疏性。

$$T_{u,i} = \operatorname{argmin} T = (\chi \| \phi_{u,i} - y_{u,i} \|_2^2 + \eta \| W_T \|_{1,2}) \quad (6)$$

其中,  $y_{u,i}$  为用户与项目之间的真实交互;  $\chi, \eta \in [0, 1]$  为损失系数, 用于平衡重构误差;  $\| W_T \|_{1,2}$  为混合范数约束投影矩阵稀疏性。

### 2.2.2 异构知识蒸馏

本文提出的异构知识蒸馏机制主要包含 3 个关键组成部分: 排序知识的表征, 异构教师知识的自适应融合, 以及知识

迁移的动态调节。

首先,排名匹配蒸馏<sup>[28]</sup>训练学生模型来模拟教师的排列,通过交叉熵损失优化学生模型对教师模型排序的细粒度匹配,学习教师在正样本  $P$  上的局部排序模式,忽略负样本  $N$  的低排名关系。具体如下:

$$\ell_p(P, N) = -\log \prod_{k=1}^{|P|} \frac{\exp \psi_{u, P_k}}{\exp \psi_{u, P_k} + \exp \psi_{u, N_l}} \quad (7)$$

其中,  $P$  是教师模型排序中靠前的正样本,  $N$  是从剩余项目中随机抽取的负样本。通过最小化该损失,提升推荐系统对用户真实偏好的捕捉精度。

$\text{RMI}^{[29]}$  量化教师排名与学生排名的依赖程度,评估排序知识的传递效。具体如下:

$$\text{RMI}(\pi, \pi') = \sum_{k=1}^N \sum_{l=1}^N P(\pi_k = \pi'_l) \log \frac{P(\pi_k = \pi'_l)}{P(\pi_k) P(\pi'_l)} \quad (8)$$

其中,  $\pi$  为学生模型生成的项目排名;  $\pi'$  为教师模型生成的项目排名;  $\pi_k$  为项目排列中的第  $k$  个项目;  $P(\cdot)$  为概率质量函数,用于计算排名结果的概率分布。

其次,异构教师知识的自适应融合根据排名互信息动态分配各教师模型的权重,让学生模型优先学习与自身排名一致性高的教师知识,平衡异构教师之间的知识转移。具体如下:

$$w_x = \frac{\text{RMI}(\pi, \pi'_x)}{\sum_{x=1}^X \text{RMI}(\pi, \pi'_x)} \quad (9)$$

其中,  $X$  为异构教师模型的数量,  $\pi'_x$  为第  $x$  个教师模型生成的项目排名。

将多个教师模型的排序知识按权重融合,生成动态目标排名,为学生模型提供更全面的学习信号,优化学生模型的排序能力。具体如下:

$$\pi^d = \sum_{x=1}^X w_x \pi'_x \quad (10)$$

随后,通过最小化排名互信息迫使学生排名趋近动态目标排名,优化学生模型与动态目标排名的一致性,强化多教师集成知识的传递效果。

$$\ell_{\text{RMI}} = 1 - \text{RMI}(\pi, \pi') \quad (11)$$

知识迁移的动态调节,即在训练过程中,通过动态调整目标排名,也就是从局部  $\ell_p$  向整体  $\ell_{\text{rank}}$  的转移,实现知识迁移难度的逐步递增,缓解异构模型间的知识差异。计算方式如下:

$$\ell_{\text{rank}} = -\log \prod_{k=1}^{|P|} \frac{\exp \psi_{u, P_k}}{\exp \psi_{u, P_k} + \sum_{l=1}^{|N|} \exp \psi_{u, N_l}} \quad (12)$$

### 2.2.3 联合优化目标

通过融合重构损失、稀疏损失、互信息损失、排序损失,实现对模型性能的全面优化,平衡推荐准确性、模型压缩效率和知识蒸馏效果等多目标。

$$\ell_{\text{total}} = \ell_{\text{rec}} + \alpha \ell_{\text{sparse}} + \beta \ell_{\text{RMI}} + \gamma \ell_{\text{rank}} \quad (13)$$

其中,  $\ell_{\text{rec}} = \sum_{u,i}^{\Lambda} (y_{u,i} - \hat{y}_{u,i})$  为重构损失,约束模型对用户-项目交互的还原能力,确保基础预测精度;  $\ell_{\text{sparse}} = \sum_{u,i} s_{u,i}^2$  为稀疏损失,其中  $s_{u,i}$  为动态调整的稀疏率,该损失用于压缩模型参数,降低冗余,提升模型的计算效率;  $\ell_{\text{RMI}}$  为互信息损失,优化学生与教师排名的一致性,提高异构知识蒸馏效果;  $\ell_{\text{rank}}$  为自适应蒸馏损失,强化模型对用户兴趣的排序能力,使推荐结果

更贴合用户真实需求;  $\alpha, \beta, \gamma \in [0, 1]$  为损失权重系数,调节各项损失对总损失的贡献比例,最终实现推荐性能、模型效率的综合优化。

算法 1 展示了动态稀疏在不同训练阶段的适应性变化。

### 算法 1 稀疏率的计算及更新过程

输入: 用户嵌入  $e_u$ ; 项目嵌入  $e_i$ ; 项目流行度  $\text{Pop}_{(i)}$ ; 稠稀疏预算  $S$ ; 训练轮次  $T$

输出: 动态稀疏率  $S_{u,i}$ ;

1. 初始化稀疏矩阵  $S_{\text{all}}$ ;
2. For each epoch in  $T$ ;
3.   For each user  $u$  in  $U$ ;
4.     For each item  $i$  in  $I$ ;
5.       计算交互冗余度  $R(u, i) = \frac{e_u^T e_i}{\|e_u\| \|e_i\|}$ ;
6.       计算流行度权重  $\text{Pop}_{(i)} = \sigma(\lambda \cdot (\text{Pop}_{(i)} - u_{\text{pop}}))$ ;
7.       动态稀疏率  $S_{u,i} = S * (R_{(u,i)} * (1 - \text{Pop}_{(i)}))$ ;
8.       更新  $S_{\text{all}}(u, i) = S_{u,i}$ ;
9.   End for
10.   通过稀疏率矩阵  $S_{\text{all}}$  对嵌入向量进行元素级过滤(式(3)一式(4));
11.   执行低秩矩阵分解与联合优化(式(5)一式(6),及式(13));
12.   根据损失函数反馈,调整超参表;
13.   End for
14. End for
15. Return  $S_{u,i}$ ;

根据算法 1,在早期训练阶段,模型对用户兴趣捕捉尚不精准,此时稀疏率主要依赖项目流行度过滤噪声,通过保留高流行度项目的连接稳定训练,例如剔除长尾项目的低可靠交互;中期阶段,随着嵌入向量的更新,交互冗余度成为主导因素,模型开始识别并压缩用户与项目间的冗余连接(如同类项目重复交互),聚焦关键特征以优化参数效率;后期阶段,稀疏率趋于稳定,通过超参数微调平衡模型压缩与性能,避免过度稀疏导致信息丢失。值得注意的是,动态稀疏率在训练初期可能因嵌入向量不稳定而波动,尤其在 CiteULike(稀疏度 0.000956)这类稀疏数据集中,冗余度计算易受噪声干扰,需通过预热训练、平滑正则化等策略缓解风险。

## 3 实验

### 3.1 数据集

实验采用 Amazon-music, CiteULike 和 Foursquare 这 3 个公开的数据集来评估提出的模型。表 1 列出了数据集的统计信息。

表 1 实验数据信息  
Table 1 Statistics of datasets

| 数据集          | 用户/个   | 项目/个   | 交互数据/条    | 稀疏度       |
|--------------|--------|--------|-----------|-----------|
| Amazon-music | 5 729  | 9 267  | 65 344    | 0.001 231 |
| CiteULike    | 5 219  | 25 181 | 125 580   | 0.000 956 |
| Foursquare   | 19 465 | 28 593 | 1 115 108 | 0.002 004 |

### 3.2 基线算法

本文提出的融合动态稀疏性与异构知识蒸馏的协同优化框架,分别与以下 5 种先进的推荐模型比较。

1)RRD<sup>[30]</sup>:该模型通过多专家动态选择机制迁移教师模型隐层知识,结合放松排名蒸馏动态采样关键物品优化排序,

在保持推荐性能的同时显著提升模型推理效率。

2) CL-DRD<sup>[31]</sup>:该模型通过课程学习动态调整蒸馏数据难度,从粗粒度偏好对逐步过渡到细粒度文档排序,优化稠密检索模型的排序能力,在多数数据集上验证其有效性。

3) DCD<sup>[28]</sup>:该模型提出双修正策略优化排序蒸馏,通过动态校准排序偏差与特征分布,将精排知识迁移至轻量级模型,提升 Top- $k$  推荐的排序质量与效率。

4) MKDRec<sup>[18]</sup>:该模型融合多源异构数据(用户行为、物品语义、社交关系等),构建动态知识蒸馏框架,实现内容信息与协同信号的自适应对齐,有效缓解冷启动与数据稀疏问题,提升推荐精度与泛化能力。

5) HetComp<sup>[10]</sup>:该模型设计动态知识蒸馏框架,将异构模型知识按训练轨迹从易到难迁移至轻量级学生模型,优化推理效率与推荐精度。

### 3.3 实验设置

本文模型和算法使用 Python 语言基于 TensorFlow 框架实现,机器配置为 GPU i7-8700K 3.7 GHz,操作系统为 Windows 10。在实验中,对于每一个数据集,随机选择每个用户的 80% 的交互历史构成训练集,剩余的 20% 作为测试集。随机选取训练集中的 20% 作为验证集,保证每个用户在训练集中至少有 10 条交互记录。基于以往模型<sup>[10]</sup>的经验,学生模型使用 MF<sup>[30]</sup>, ML<sup>[32]</sup> 和 DNN<sup>[33]</sup>;教师模型选取代表性的 MF<sup>[34]</sup>, ML<sup>[35]</sup>, DNN<sup>[36]</sup>, GNN<sup>[37]</sup>, AE<sup>[38]</sup> 和 IAE<sup>[39]</sup>。教师模型的用户/项目嵌入维度设置为 64,隐藏层维度为 [32, 64],学生模型设置为 6(约为教师模型的 1/10)。教师模型训练轨迹采样点为 4 个(对应 25%, 50%, 75% 100% 训练进度),每个教师模型保存 20 个训练轨迹点(每 10 个 epoch 保存一次),学生模型在训练时按顺序学习这些轨迹点的知识。批处理大小为 256,迭代次数为 200 次时模型收敛。

本文教师模型选用多种异构结构并采样 25%, 50%, 75% 和 100% 训练进度轨迹点,其在“课程学习”中的作用逻辑为:按训练进度从低到高将轨迹点划分为“简单到复杂”的知识阶段,早期轨迹点(25% 和 50%)对应教师模型粗粒度排序知识(如高流行度项目偏好)供学生模型初始化学习,后期轨迹点(75% 和 100%)包含细粒度排序细节(如长尾项目差异)用于进阶阶段;单个教师轨迹点权重根据学生的学习状态动态调整(早期侧重低进度点,后期侧重高进度点),多教师轨迹点通过排名互信息(RMI)计算动态权重并融合,以优先学习与学生排名一致性高的知识,避免单一模型偏差。同时,使用早期轨迹点热启动学生模型以加速嵌入向量收敛,并根据验证集性能触发轨迹点切换,过滤低互信息异常轨迹点,以保障知识迁移的稳定性。该机制通过分阶段释放知识、动态分配权重及风险控制,缓解异构模型间的知识差异,提升学生模型对不同难度排序模式的学习效率。

### 3.4 评估指标

在本文,每种模型均输出用户对所有项目的偏好分数。为了评估 Top- $k$  推荐和偏好排序的有效性,实验使用召回率(Recall@ $k$ )、归一化折现累积收益(NDCG@ $k$ )、负似然对数(NLL)、稀疏性(Sparsity)、推理延迟(Inference Latency)作为推荐效果方面的评价指标。

召回率是所有“被正确推荐的项目”占有“应该被推荐的正确的项目”的比例。

$$Recall@k = \frac{1}{|U|} \sum_{u \in U} \frac{|HitItems(u) \cap RecList_k(u)|}{|HitItems(u)|} \quad (14)$$

其中, $U$  是用户集合, $HitItems(u)$  是用户  $u$  真实交互的项目集合, $RecList_k(u)$  是模型为用户  $u$  生成的 Top- $k$  推荐列表。

NDCG 是一种基于排名的测试指标,排名靠前的项目得分更高。

$$NDCG@k = \frac{1}{|U|} \sum_{u=1}^{|U|} \frac{\sum_{i=1}^K \frac{2^{rel_i} - 1}{\log_2(i+1)}}{\sum_{i=1}^{|RecList(u)|} \frac{2^{rel_i} - 1}{\log_2(i+1)}} \quad (15)$$

其中, $rel_i$  是第  $i$  个推荐项目的相关性得分, $RecList(u)$  是用户  $u$  的完整推荐列表。

负似然对数(NLL)用于衡量模型预测概率与真实交互的拟合度,使模型预测概率尽可能贴近真实交互模式。

$$NLL(\theta) = -\frac{1}{|D|} \sum_{(u,i) \in D} (y_{u,i} * \ln \hat{y}_{u,i} + (1 - y_{u,i}) * \ln(1 - \hat{y}_{u,i})) \quad (16)$$

其中, $\theta$  为模型所有可学习参数, $D$  为交互样本集, $y_{u,i}$  为用户-项目的真实交互, $\ln(\cdot)$  为自然对数函数。 $\hat{y}_{u,i}$  为模型预测的交互概率,由动态稀疏嵌入(式(3)和式(4))和低秩投影(式(5))共同计算所得。

$$\hat{y}_{u,i} = \sigma(W_u * \hat{e}_u + W_i * \hat{e}_i + b) \quad (17)$$

其中, $W_u$  和  $W_i$  为投影矩阵, $b$  为偏置项。

稀疏性量化用户-项目交互矩阵中缺失值的比例,用于评估模型压缩技术的有效性。

$$Sparsity = \frac{\sum_{u,i} S_{u,i}}{|U| * |I|} \quad (18)$$

其中, $S_{u,i}$  为用户  $u$  与项目  $i$  之间的交互稀疏率, $I$  为项目集合。

推理延迟为模型生成推荐结果所需的时间,用于衡量用户体验。

$$Latency = \frac{1}{M} \sum_{m=1}^M Time_m \quad (19)$$

其中, $M$  为推理测试的总次数(需预先设定,如  $M=1000$  次), $Time_m$  为第  $m$  次推理的耗时(单位为 ms)。

### 3.5 实验结果

#### 3.5.1 模型推荐效果

将本文模型与对比模型分别在 3 个相同的数据集上实验,结果如表 2、表 3 所列。从模型推荐效果的总体比较来看,DySparseHKD 模型在 3 个数据集上的 Recall@ $k$  和 NDCG@ $k$  均优于其他方法,证明了模型的高效性与良好的泛化能力。

模型训练过程中, $k$  的取值分别为 10 和 50,这是因为  $k=10$  代表用户高度注意力区域的推荐效果,符合用户实际行为,用于提升用户即时体验; $k=50$  能够覆盖用户完整兴趣范围的长尾推荐,减少因推荐范围过窄导致的用户兴趣的遗漏,更全面地评估模型的推荐能力。

结果表明,本文模型通过动态稀疏技术,结合交互冗余度与项目流行度动态分配稀疏率,能更精准地筛选关键特征,减少噪声干扰,在稀疏交互数据中能更有效地挖掘潜在用户兴

趣,提升推荐的准确性。利用异构知识蒸馏从不同架构的教师模型中提取多样化知识,多源知识的融合,使模型能覆盖用户短期冲动偏好与长期稳定兴趣,捕捉用户-项目交互的多维度特征,增强对复杂偏好模式的学习能力。

表 2 Recall@k 的总体比较  
Table 2 Overall performance comparison of Recall@k

| 学生模型 | 方法          | Amusic        |               | CiteULike     |               | Foursquare    |               |
|------|-------------|---------------|---------------|---------------|---------------|---------------|---------------|
|      |             | NDCG@10       | NDCG@50       | NDCG@10       | NDCG@50       | NDCG@10       | NDCG@50       |
| MF   | RRD         | 0.0659        | 0.1077        | 0.0740        | 0.1113        | 0.0905        | 0.1446        |
|      | CL-DRD      | 0.0648        | 0.1071        | 0.0794        | 0.1175        | 0.0933        | 0.1464        |
|      | DCD         | 0.0675        | 0.1079        | 0.0851        | 0.1246        | 0.0965        | 0.1491        |
|      | MKDRec      | 0.0688        | 0.1101        | 0.0897        | 0.1288        | 0.1002        | 0.1504        |
|      | HetComp     | 0.0747        | 0.1157        | 0.1031        | 0.1396        | 0.1036        | 0.1594        |
|      | DySparseHKD | <b>0.0813</b> | <b>0.1247</b> | <b>0.1104</b> | <b>0.1488</b> | <b>0.1147</b> | <b>0.1667</b> |
|      | Improve/%   | <b>8.84</b>   | <b>7.78</b>   | <b>7.08</b>   | <b>6.59</b>   | <b>10.71</b>  | <b>4.58</b>   |
| ML   | RRD         | 0.0643        | 0.1074        | 0.0701        | 0.1116        | 0.0813        | 0.1366        |
|      | CL-DRD      | 0.0621        | 0.1061        | 0.0718        | 0.1131        | 0.0825        | 0.1387        |
|      | DCD         | 0.0653        | 0.1086        | 0.0724        | 0.1144        | 0.0845        | 0.1399        |
|      | MKDRec      | 0.0673        | 0.1110        | 0.0832        | 0.1185        | 0.0874        | 0.1410        |
|      | HetComp     | 0.0751        | 0.1156        | 0.0916        | 0.1287        | 0.0962        | 0.1521        |
|      | DySparseHKD | <b>0.0813</b> | <b>0.1243</b> | <b>0.1065</b> | <b>0.1347</b> | <b>0.1102</b> | <b>0.1642</b> |
|      | Improve/%   | <b>8.26</b>   | <b>7.53</b>   | <b>16.27</b>  | <b>4.66</b>   | <b>14.55</b>  | <b>7.96</b>   |
| DNN  | RRD         | 0.0613        | 0.1016        | 0.0792        | 0.1186        | 0.0939        | 0.1484        |
|      | CL-DRD      | 0.0623        | 0.1047        | 0.0816        | 0.1183        | 0.0983        | 0.1536        |
|      | DCD         | 0.0646        | 0.1071        | 0.0838        | 0.1240        | 0.1017        | 0.1576        |
|      | MKDRec      | 0.0675        | 0.1096        | 0.0877        | 0.1263        | 0.1023        | 0.1587        |
|      | HetComp     | 0.0768        | 0.1190        | 0.1050        | 0.1413        | 0.1079        | 0.1642        |
|      | DySparseHKD | <b>0.0822</b> | <b>0.1249</b> | <b>0.1163</b> | <b>0.1578</b> | <b>0.1182</b> | <b>0.1795</b> |
|      | Improve/%   | <b>7.03</b>   | <b>4.96</b>   | <b>10.76</b>  | <b>11.68</b>  | <b>9.55</b>   | <b>9.32</b>   |

表 3 NDCG@k 总体比较  
Table 3 Overall performance comparison of NDCG@k

| 学生模型 | 方法          | Amusic        |               | CiteULike     |               | Foursquare    |               |
|------|-------------|---------------|---------------|---------------|---------------|---------------|---------------|
|      |             | NDCG@10       | NDCG@50       | NDCG@10       | NDCG@50       | NDCG@10       | NDCG@50       |
| MF   | RRD         | 0.0659        | 0.1077        | 0.0740        | 0.1113        | 0.0905        | 0.1446        |
|      | CL-DRD      | 0.0648        | 0.1071        | 0.0794        | 0.1175        | 0.0933        | 0.1464        |
|      | DCD         | 0.0675        | 0.1079        | 0.0851        | 0.1246        | 0.0965        | 0.1491        |
|      | MKDRec      | 0.0688        | 0.1101        | 0.0897        | 0.1288        | 0.1002        | 0.1504        |
|      | HetComp     | 0.0747        | 0.1157        | 0.1031        | 0.1396        | 0.1036        | 0.1594        |
|      | DySparseHKD | <b>0.0813</b> | <b>0.1247</b> | <b>0.1104</b> | <b>0.1488</b> | <b>0.1147</b> | <b>0.1667</b> |
|      | Improve/%   | <b>8.84</b>   | <b>7.78</b>   | <b>7.08</b>   | <b>6.59</b>   | <b>10.71</b>  | <b>4.58</b>   |
| ML   | RRD         | 0.0643        | 0.1074        | 0.0701        | 0.1116        | 0.0813        | 0.1366        |
|      | CL-DRD      | 0.0621        | 0.1061        | 0.0718        | 0.1131        | 0.0825        | 0.1387        |
|      | DCD         | 0.0653        | 0.1086        | 0.0724        | 0.1144        | 0.0845        | 0.1399        |
|      | MKDRec      | 0.0673        | 0.1110        | 0.0832        | 0.1185        | 0.0874        | 0.1410        |
|      | HetComp     | 0.0751        | 0.1156        | 0.0916        | 0.1287        | 0.0962        | 0.1521        |
|      | DySparseHKD | <b>0.0813</b> | <b>0.1243</b> | <b>0.1065</b> | <b>0.1347</b> | <b>0.1102</b> | <b>0.1642</b> |
|      | Improve/%   | <b>8.26</b>   | <b>7.53</b>   | <b>16.27</b>  | <b>4.66</b>   | <b>14.55</b>  | <b>7.96</b>   |
| DNN  | RRD         | 0.0613        | 0.1016        | 0.0792        | 0.1186        | 0.0939        | 0.1484        |
|      | CL-DRD      | 0.0623        | 0.1047        | 0.0816        | 0.1183        | 0.0983        | 0.1536        |
|      | DCD         | 0.0646        | 0.1071        | 0.0838        | 0.1240        | 0.1017        | 0.1576        |
|      | MKDRec      | 0.0675        | 0.1096        | 0.0877        | 0.1263        | 0.1023        | 0.1587        |
|      | HetComp     | 0.0768        | 0.1190        | 0.1050        | 0.1413        | 0.1079        | 0.1642        |
|      | DySparseHKD | <b>0.0822</b> | <b>0.1249</b> | <b>0.1163</b> | <b>0.1578</b> | <b>0.1182</b> | <b>0.1795</b> |
|      | Improve/%   | <b>7.03</b>   | <b>4.96</b>   | <b>10.76</b>  | <b>11.68</b>  | <b>9.55</b>   | <b>9.32</b>   |

3.5.2 多维度性能评估

DySparseHKD模型在多数数据集上呈现显著多维度性能优势,实验结果如表4和表5所列。表4所列数据表明,其通过动态稀疏技术对用户-项目交互矩阵的冗余连接进行结构性剔除,稀疏度指标显著高于基线模型,验证了交互冗余分析与动态稀疏率分配策略的有效性;负对数似然(NLL)值低于基线模型,表明模型通过低秩矩阵分解与异构知识蒸馏的特征协同优化,显著提升了用户-项目交互概率预测与真实分布的拟合精度。结合表5所列模型参数量与推理延迟数据,动态稀疏技术使模型参数量大

幅压缩,推理延迟显著降低,而异构知识蒸馏通过多教师训练轨迹的渐进式知识迁移,在轻量化架构下保留了细粒度排序能力。

表 4 稀疏率与负对数似然评估  
Table 4 Evaluation of sparsity rate and negative log likelihood

| Method      | Amusic       |               | CiteULike    |               | Foursquare   |               |
|-------------|--------------|---------------|--------------|---------------|--------------|---------------|
|             | Sparsity     | NLL           | Sparsity     | NLL           | Sparsity     | NLL           |
| MKDRec      | 0.496        | 0.0714        | 0.524        | 0.0879        | 0.562        | 0.0731        |
| HetComp     | 0.529        | 0.0689        | 0.558        | 0.0812        | 0.594        | 0.0795        |
| DySparseHKD | <b>0.689</b> | <b>0.0523</b> | <b>0.654</b> | <b>0.0701</b> | <b>0.678</b> | <b>0.0658</b> |

表 5 参数量与推理延迟评估

Table 5 Evaluation of parameter quantity and inference latency

| Method      | Amusic        |             | CiteULike     |             | Foursquare    |              |
|-------------|---------------|-------------|---------------|-------------|---------------|--------------|
|             | Parameters/百万 | Latency/s   | Parameters/百万 | Latency/s   | Parameters/百万 | Latency/s    |
| MKDRec      | 2.14          | 9.03        | 2.89          | 11.21       | 3.62          | 14.06        |
| HetComp     | 1.87          | 8.85        | 2.55          | 10.87       | 3.28          | 13.89        |
| DySparseHKD | 0.92          | <b>7.12</b> | 1.24          | <b>9.03</b> | 1.56          | <b>12.01</b> |

实验结果表明,该模型通过动态稀疏的冗余压缩与异构蒸馏的知识增强协同机制,实现了模型复杂度与推荐性能的深度解耦,在资源受限场景下达成推理效率与预测可靠性的双重提升。

3.5.3 消融实验

为验证模型的有效性,对其进行消融实验,结果如图 2 所示。其中,DySparseHKD\_1 为消融动态稀疏化模型,DyS-

parseHKD\_2 为消融异构知识蒸馏模型。结果表明,DyS-parseHKD 模型的推荐效果明显更为优异。

实验结果表明,DySparseHKD\_1 难以从稀疏数据中高效提取有效信息,无法充分捕捉用户-项目交互的潜在模式;DySparseHKD\_2 模型的知识来源变得单一,难以捕捉复杂多变的用户偏好与物品特征,无法充分利用不同模型架构的优势,进而导致模型性能降低。

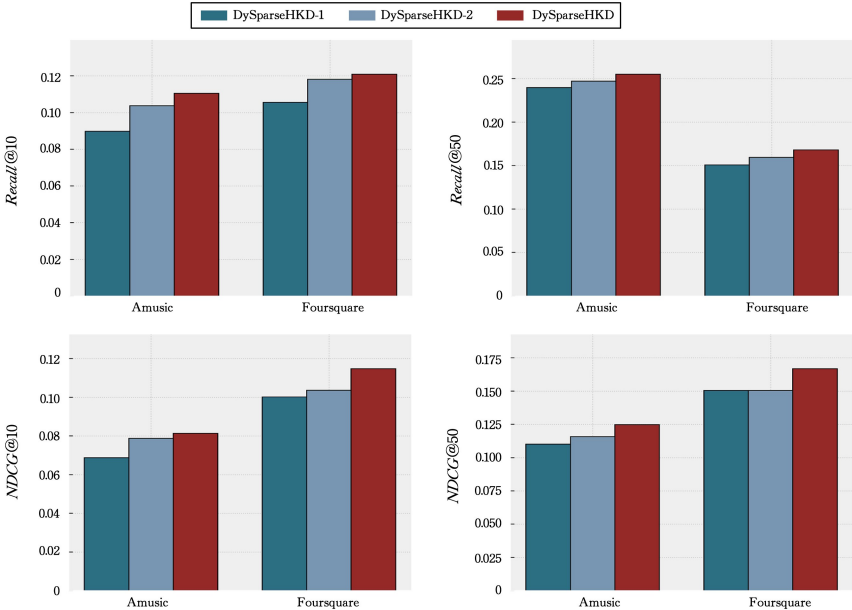


图 2 消融分析

Fig. 2 Ablation analyses

综上验证了动态稀疏化模块和异构知识蒸馏模块对模型性能均有贡献,动态稀疏化模块保障了模型对稀疏数据的高效处理,异构知识蒸馏模块则丰富了模型的知识输入与特征学习能力,两模块形成“压缩-增强”的协同闭环,而非独立模块的叠加,提升了模型的检索与排序能力。

**结束语** 本文提出了一种融合动态稀疏性与异构知识蒸馏的 Top-k 推荐算法 DySparseHKD,创新性地融合了动态稀疏技术和异构知识蒸馏架构,解决了现有推荐方法中知识利用不充分和模型泛化能力有限的问题。实验结果表明,所提方法在多个公共数据集上均优于现有的推荐算法。具体来说,动态稀疏技术有效缓解交互冗余问题,异构知识蒸馏将多源异构知识有效整合传递至轻量级学生模型。未来可进一步探索多模态数据融合(如兴趣点语义、社交关系)以增强特征表达,开发在线动态稀疏调整机制优化实时推荐性能,并结合图神经网络与深度表征学习挖掘用户-项目深层关联,解决冷启动问题并提升推荐准确性,同时探索跨领域知识迁移策略以扩展应用场景。

参考文献

[1] JIN Y,LI M K,LU Y,et al. Long-Tailed Visual Recognition via

Self-Heterogeneous Integration with Knowledge Excavation [C]//CVPR 2023: IEEE Conference on Computer Vision and Pattern Recognition. Vancouver,BC Canada:IEEE,2023:23695-23704.

[2] ZHANG B C,SUN C G,TAN J C,et al. SHARK: A Lightweight Model Compression Approach for Large-scale Recommender Systems[C]// Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. New York,NY,United States: Association for Computing Machinery,2023:4930-4937.

[3] ADNAN M,EBRAHIMZADEH M Y,MAHAIAN D,et al. Nair: Heterogeneous Acceleration Pipeline for Recommendation System Training[C]// International Symposium on Computer Architecture (ISCA 2024). Buenos Aires, Argentina: IEEE, 2024:1063-1079.

[4] ISINKAYE F O. Matrix Factorization in Recommender Systems: Algorithms, Applications, and Peculiar Challenges[J]. IE-TE Journal of Research,2021,69(9):6087-6100.

[5] GAO C,ZHENG Y,LI N,et al. A Survey of Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions[J]. Trans. Recomm. Syst.,2023,1(1):1-51.

- [6] HUANG W Z, ZHANG Y X, ZHENG X W, et al. Dynamic Low-Rank Sparse Adaptation for Large Language Models[J]. arXiv:2502.14816, 2025.
- [7] KANG S K, HWANG J Y, KWEON W B, et al. DERRD: A Knowledge Distillation Framework for Recommender System [C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. New York, NY, United States: Association for Computing Machinery, 2020: 605-614.
- [8] KANG S K, HWANG J Y, KWEON W B, et al. ItemsideRanking Regularized Distillation for Recommender System[J]. Information Sciences, 2021, 580(2021): 15-34.
- [9] YU S, HE X L, CHEN K, et al. HKDSME: Heterogeneous Knowledge Distillation for Semi-supervised Singing Melody Extraction Using Harmonic Supervision[C]//Proceedings of the 32nd ACM International Conference on Multimedia. New York, NY, United States: Association for Computing Machinery, 2024: 545-553.
- [10] KANG S K, KWEON W D, DONGHA LEE D, et al. Distillation from Heterogeneous Models for Top- $k$  Recommendation[C]//Proceedings of the ACM Web Conference 2023. New York, NY, United States: Association for Computing Machinery, 2023: 801-811.
- [11] HAN S, MAO H Z, DALLY W J. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding[C]//Proceedings of the International Conference on Learning Representations 2016. Barcelona, Spain: ICLR, 2016: 1131-1135.
- [12] LIU Z, LI J G, SHEN Z Q, et al. Learning Efficient Convolutional Networks through Network Slimming[C]//ICCV 2017: International Conference on Computer Vision. Venice, Italy: IEEE, 2017: 2755-2763.
- [13] MOCANU D C, MOCANU E, STONE P, et al. Evolutionary Training of Sparse Artificial Neural Networks: A Network Science Perspective[J]. arXiv:1707.04780v1, 2017.
- [14] FRANKLE J, CARBIN M. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks[C]//Proceedings of the International Conference on Learning Representations. New Orleans, United States: ICLR, 2019: 8359-8401.
- [15] HINTON G E, VINYALS O, DEAN J. Distilling the Knowledge in a Neural Network[J]. arXiv:1503.02531, 2015.
- [16] LIU D G, CHENG P X, DONG Z H, et al. A General Knowledge Distillation Framework for Counterfactual Recommendation via Uniform Data[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, NY, United States: Association for Computing Machinery, 2020: 831-840.
- [17] HOSSAIN M I, AKHTER S H, HONG C S, et al. Single Teacher, Multiple Perspectives: Teacher Knowledge Augmentation for Enhanced Knowledge Distillation[C]//Proceedings of the International Conference on Learning Representations. Singapore: ICLR, 2025: 503-524.
- [18] ZHANG Y L, SU J, ZHAO H Y. Knowledge-aware and Interactive Multi-view Distillation Recommendation Algorithm [J]. Computer, 2024, 34(3): 587-634.
- [19] CHANG X Q, LI Y M, LI Z C, et al. An Emotional Classification Method Based on Multi-teacher and Multi-student Knowledge Distillation[J]. Journal of Chinese Information Technology, 2024, 38(10): 127-134.
- [20] YOU S, XU C, XU C, et al. Learning from Multiple Teacher Networks[C]//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, United States: Association for Computing Machinery, 2017: 1285-1294.
- [21] CHEN X, ZHANG Y F, XU H T, et al. Adversarial Distillation for Efficient Recommendation with External Knowledge[J]. ACM Trans. Inf. Syst., 2019, 37(1): 12:1-12:28.
- [22] XIA X, YIN H Z, YU J L, et al. On-Device Next-Item Recommendation with Self Supervised Knowledge Distillation[C]//Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, NY, United States: Association for Computing Machinery, 2022: 546-555.
- [23] WEI W, TANG J B, XIA L H, et al. PromptMM: Multi-Modal Knowledge Distillation for Recommendation with Prompt-Tuning[C]//Proceedings of the ACM Web Conference 2024. New York, NY, United States: Association for Computing Machinery, 2024: 3217-3228.
- [24] ZHU J X, MENG X F, XING CH Z, et al. Collaborative filtering recommendation approach fused with graph convolutional attention mechanism[J]. CAAI Transactions on Intelligent Systems, 2023, 18(6): 1295-1304.
- [25] HE F Z, SHI J P. A Diversity Recommendation Method Under the Constraint of Non-uniform Division Matrix[J]. Computer Science and Exploration, 2019, 13(2): 226-238.
- [26] KAUR K, CHADHA M, GUPTA V, et al. Efficient and Responsible Adaptation of Large Language Models for Robust and Equitable Top-k Recommendations [J]. arXiv: 2501.04762, 2025.
- [27] CHEN Z F, GAN W S, WU J Y, et al. Data Scarcity in Recommendation Systems: A Survey[J]. ACM Transactions on Recommender Systems, 2025, 24(9): 3:1-31.
- [28] LEE Y J, KIM K E. Dual Correction Strategy for RankingDistillation in Top-N Recommender System[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management. New York, NY, United States: Association for Computing Machinery, 2021: 3186-3190.
- [29] KEMERTAS M, PISHDAD L, DERPANIS K G, et al. RankMI: A Mutual Information Maximizing Ranking Loss[C]//CVPR 2020: IEEE Conference on Computer Vision and Pattern Recognition. Seattle, WA, USA: IEEE, 2020: 14350-14359.
- [30] KANG S K, HWANG J Y, KWEON W B, et al. DERRD: A Knowledge Distillation Framework for Recommender System [C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. New York, NY, United States: Association for Computing Machinery, 2020: 605-614.
- [31] ZENG H S, ZAMANI H, VINAY V. Curriculum Learning for Dense Retrieval Distillation[C]//Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, NY, United States: Association for Computing Machinery, 2022: 1979-1983.
- [32] KWEON W, KANG S K, YU H. Bidirectional Distillation for Top-K Recommender System[C]//Proceedings of the Web Con-

ference 2021. New York, NY, United States: Association for Computing Machinery, 2021:3861-3871.

[33] LEE Y J, KIM K E. Dual Correction Strategy for Ranking Distillation in Top-N Recommender System [C] // Proceedings of the 30th ACM International Conference on Information & Knowledge Management. New York, NY, United States: Association for Computing Machinery, 2021:3186-3190.

[34] RENDLE S, FREUDENTHALER C H, GANTNER Z, et al. BPR: Bayesian personalized ranking from implicit feedback [C] // Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. Arlington, Virginia, United States: AUAI Press, 2009:452-461.

[35] HSIEH C K, YANG L Q, CUI Y, et al. Collaborative metric learning [C] // Proceedings of the 26th International Conference on World Wide Web. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2017:193-201.

[36] HE X N, LIAO L Z, ZHANG H W, et al. Neural collaborative filtering [C] // Proceedings of the 26th International Conference on World Wide Web. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2017:173-182.

[37] HE X N, DENG K, WANG X, et al. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation [C] // Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, NY, United States: Association for Computing Machinery, 2020:639-648.

[38] LIANG D W, KRISHNAN R G, HOFFMAN M D, et al. Variational Autoencoders for Collaborative Filtering [C] // Proceedings of the 2018 World Wide Web Conference. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2018:689-698.

[39] SEDHAIN S, MENON A K, SANNER S, et al. Autorec: Autoencoders meet collaborative filtering [C] // Proceedings of the 24th International Conference on World Wide Web. New York, NY, United States: Association for Computing Machinery, 2015:111-112.

[40] WANG X, LI Y, ZHANG H. DSL: Dynamic Sparse Learning for Efficient and Accurate Recommendation. [C] // Proceedings of the 17th ACM International Conference on Web Search and Data Mining. 2024:123-132.

[41] CHEN J, LIU S, WANG Z. LightGNN: Sparse Scoring and Multi-Level Knowledge Distillation for Efficient Graph Neural Recommendation. [C] // Proceedings of the 18th ACM International Conference on Web Search and Data Mining. 2025:456-465.



**FU Shiqi**, born in 2005, undergraduate. His main research interests include recommendation system and artificial intelligence.



**ZHU Jinxia**, born in 1996, postgraduate. Her main research interests include recommendation system and artificial intelligence.