

基于多粒度的图对比学习推荐算法

吴鹏飞, 苏凡军*

上海理工大学光电信息与计算机工程学院, 上海

收稿日期: 2024年3月22日; 录用日期: 2024年5月6日; 发布日期: 2024年5月14日

摘要

现有的图对比学习推荐算法, 多局限于节点级或图级的对比学习, 未能综合利用图的信息。针对这个问题, 提出了一种基于多粒度的图对比学习推荐算法(Multi-Granularity Graph Contrastive Learning, MGGCL)。该算法通过融合节点级、子图级和全图级的对比学习, 实现对用户和项目的跨粒度建模。在节点级别引入邻域对比学习方法; 在子图级别采用随机游走生成不同局部结构, 进行子图级对比学习; 在全图级别通过采用相似子图 - 原图采样的组合方式, 构建全图级别的对比视图。最后利用多任务策略对推荐监督任务和对比学习任务进行联合优化, 提升推荐效果。在真实数据集Yelp和Amazon-Book上进行实验, 采用Recall和NDCG指标进行评估。实验结果表明, 提出的模型在两个数据集上相较于基线模型, Recall@20分别提升了4.24%和6.85%, NDCG@20分别提升了4.04%和9.66%。

关键词

推荐系统, 图神经网络, 图对比学习, 多粒度

Multi-Granularity Graph Contrastive Learning Recommender Algorithm

Pengfei Wu, Fanjun Su*

School of Optical Information and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Mar. 22nd, 2024; accepted: May. 6th, 2024; published: May. 14th, 2024

Abstract

The existing graph contrastive learning-based recommendation algorithms are often constrained to node-level or graph-level contrastive learning, limiting their ability to fully leverage the infor-

*通讯作者。

mation present in the graph. To address this limitation, a Multi-Granularity Graph Contrastive Learning (MGGCL) approach was proposed for recommendation. The MGGCL approach integrates contrastive learning at multiple granularities to enable modeling of users and items across different scales. At the node level, a neighborhood contrastive learning method is introduced. At the subgraph level, diverse local structures are generated using random walks for subgraph-level contrastive learning. At the full graph level, a contrastive view of the entire graph is constructed by combining similar subgraphs with the original graph through sampling. In the end, a multi-task strategy is employed to jointly optimize the recommendation supervision task and the contrastive learning task, resulting in improved recommendation performance. Experimental evaluations conducted on real-world datasets, namely Yelp and Amazon-Book, employ the Recall and NDCG metrics for assessment. The results demonstrate significant enhancements of the proposed model compared to the baseline models, with Recall@20 improving by 4.24% and 6.85%, and NDCG@20 improving by 4.04% and 9.66% on the Yelp and Amazon-Book datasets, respectively.

Keywords

Recommender System, Graph Neural Network, Graph Contrastive Learning, Multi-Granularity

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

互联网技术的飞速发展、互联网的广泛普及和信息产生的快速增长带来了海量的信息可供人们获取,然而这也导致了信息过载。在这个背景下,推荐系统(Recommender System)的重要性变得尤为突出。推荐系统可以根据用户的兴趣特点和历史行为,向用户推荐感兴趣的信息[1]。传统的推荐算法可以分为:基于内容的推荐方法、基于协同过滤的推荐方法、混合推荐方法[2]。近年来,基于图神经网络(Graph Neural Network, GNN)的推荐算法应用广泛。GNN 将交互数据建模为图(例如,用户-项目交互图),并利用了图的连通性,能够聚合用户和项目的邻居节点信息,从而更为有效地学习用户和项目的嵌入用于推荐[3]。

大多数基于 GNN 的推荐算法都遵循(半)监督学习范式,需要足够的标记数据来进行模型训练[4],然而许多实际的推荐场景都面临数据稀疏性问题[5]。为了获得更具泛化性和可迁移性的表示,解决数据稀疏问题,出现了自监督学习(Self-Supervised Learning, SSL)范式[6] [7],它使 GNN 能够从未标记的图数据构建任务中学习。作为一种先进的 SSL 技术,图对比学习(Graph Contrastive Learning, GCL) [8] [9] [10] 被用于推荐系统。图对比学习通过在图数据中构造变换视图来生成正样本对,并从图中选择其他样本作为负样本,最大化正样本对之间的相似度和最小化负样本对之间的相似度。从而捕获图中不同对象间的不变特性,并学习到对下游任务有用的图表示[11]。

在上述背景下,将图对比学习应用于推荐成为研究热点。Zhu 等人提出了 GCA [12],根据节点中心度计算节点和边的重要性,自适应地丢弃节点和边,进行数据增强用以图对比学习。Wang 等人通过随机增强策略,提出了 SGL [13],通过按一定比例对点和边进行随机删除来构建结构扰动视图,再通过图编码器来提取节点的表征,构造自监督学习任务。Yu 等人不采用基于结构扰动的图增强方法,而是在嵌入空间中添加均匀噪声来创建对比视图,提出了 SimGCL [14]。Lin 等人考虑到图中的潜在邻居,将用户(项目)结构上的邻居和语义空间上的邻居包含在正对比对中,提出了 NCL [15]。Cai 等人提出了 LightGCL [16],利用奇异值分解进行对比增强,而不是手工制作增强视图。Jiang 等人分别使用图生成模型和图去噪模型

作为可训练的对比视图生成器, 提出了 AdaGCL [17], 生成自适应的对比视图进行对比学习, 提升推荐效果。

尽管图对比学习证明了其有效性, 但现有的研究仍存在一定的局限性。图数据的丰富性不仅体现在节点及其特征上, 还包括节点间的连接性和图的整体拓扑结构[18]。图对比学习如果仅在节点级别进行[19][20], 虽能捕捉到节点的局部特性, 但有可能会忽略掉图的全局结构信息, 如社区结构、中心性和聚类系数等宏观拓扑特征, 这些全局信息对于理解整个图的结构和功能至关重要[21]。图对比学习如果仅在全图级别进行[14][22], 虽然有助于捕捉整体的图结构, 但可能会导致丢失重要节点的特征和属性, 还可能忽视图中不同节点之间的局部关系和连接模式, 如节点间的同质性或异质性[23]。

针对以上问题, 本文提出了一种基于多粒度的图对比学习推荐算法(Multi-Granularity Graph Contrastive Learning, MGGCL)。MGGCL 通过节点级对比学习、子图级对比学习和全图级对比学习相结合, 从而实现用户对用户和项目的跨粒度建模。在节点级别, MGGCL 使用邻域对比学习方法, 考虑到用户-项目二分图的结构特点, 有效地对用户(项目)与其同结构的邻居之间的关系进行建模。在子图级别, MGGCL 使用随机游走的思路为节点生成不同的局部结构, 进行子图级对比学习。在全图级别, MGGCL 通过“相似子图-原图采样”的组合来构成全图级别的对比视图进行对比学习, 把握整个图的全局结构。相比单一粒度下的研究, MGGCL 算法通过综合多粒度的对比学习, 实现了对图数据中各个方面进行深入而全面的建模。并有效指导图编码器[24]的参数更新过程, 以生成高质量的嵌入向量, 从而进一步提升推荐效果。

本文通过真实数据集 Yelp 和 Amazon-Book 对 MGGCL 进行了验证, 并设置了对比实验、消融实验证明了该模型的有效性。另外对超参数进行了敏感性分析。

2. 符号和问题定义

2.1. 符号定义

给定用户集 $U = \{u\}$ 和项目集 $I = \{i\}$, 当发生隐式反馈(如点击、购买、收藏等)时, 观察到的隐式反馈矩阵即交互矩阵表示为 $R \in \{0, 1\}^{|U| \times |I|}$ 。如果存在交互则 $R_{u,i} = 1$, 不存在交互则 $R_{u,i} = 0$ 。

基于交互矩阵 R , 可以建模用户项目交互图 $G = \{V, E\}$, 其中节点集 $V = \{U \cup I\}$ 表示涉及的所有用户和项目, 边集 $E = \{(u, i) | u \in U, i \in I, R_{u,i} = 1\}$ 表示观察到的交互。对于用户 u 、项目 i , N_u 、 N_i 分别表示其在交互图 G 中的邻居。

2.2. 问题定义

给定一个包含 X 个样本的数据集 D , 其中每个样本 $(u, i, r_{u,i})$ 表示用户 u 与项目 i 之间的交互。本文的核心任务是设计并训练出一个模型。该模型基于图编码器和图对比学习方法, 能够从无标签信息中学习用户 u 和项目 i 之间的关联关系表示以及一般偏好表示, 最终根据性能评价指标对用户 u 给出前 K 个项目推荐。

3. 模型设计

3.1. 框架概述

所提出的 MGGCL 模型的整体框架如图 1 所示, 包含输入模块、对比学习模块、推荐模块三个主要模块。首先是输入模块, 输入模块接受用户和项目的交互数据, 即用户项目二分图, 作为模型的原始输入数据。接着是对比学习模块, 在该模块中实现了基于多粒度的对比学习方法。具体而言, 该模块包括节点对比学习、子图对比学习、全图对比学习和图编码器。通过在不同粒度下进行对比学习, 可以充分

理解图的信息, 并捕捉图的全局结构。三个粒度下的对比损失和图编码器的 BPR 损失组合生成整体损失函数, 用于指导模型参数的更新, 提高嵌入的质量。图编码器执行从高维图数据到低维向量嵌入的转换, 提取图的结构和特征信息。最后是推荐模块, 该模块根据用户和项目的最终嵌入向量的内积得分进行排序, 生成最终的 top-k 推荐序列。在下文中将重点介绍对比学习模块。

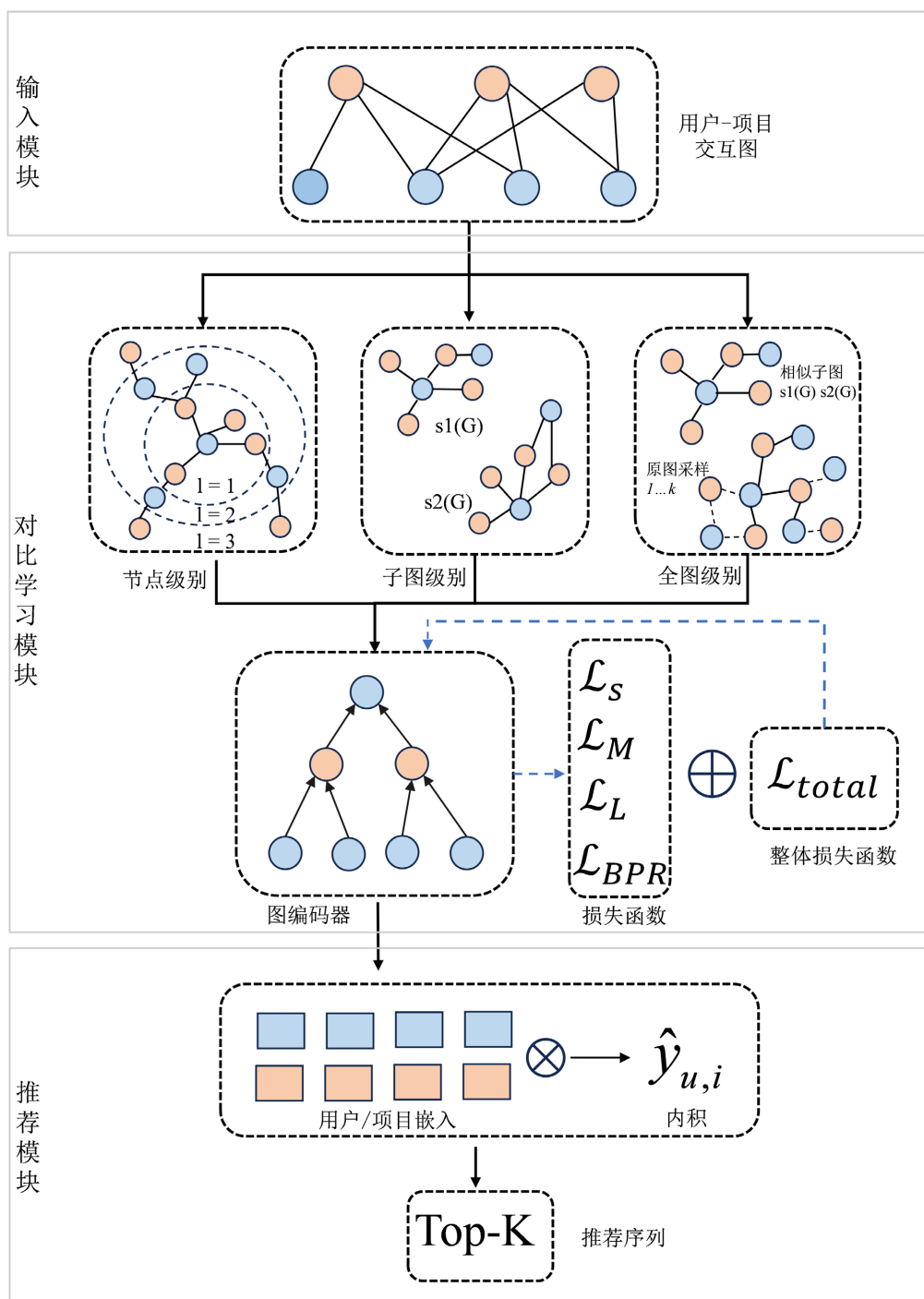


Figure 1. Model structure of MGGCL

图 1. MGGCL 模型结构

3.2. 图编码器

图编码器将用户和项目的复杂交互关系映射到低维表示空间中, 从而为推荐任务提供更好的特征表达。本文所采用的图编码器是 LightGCN [24], 这是一种轻量级图卷积网络。LightGCN 简化了信息传递和聚合的过程, 从而降低了计算复杂度, 并减少了信息损失。其采用简单的加权和聚合器, 放弃了之前方法中的特征变换和非线性激活以及自连接[3]。用户 u 的传播公式和最终嵌入表示如下:

$$\begin{aligned} e_u^{(k+1)} &= \sum_{i \in N_u} \frac{1}{\sqrt{|N_u|} \sqrt{|N_i|}} e_i^{(k)}, \\ e_u &= \frac{1}{K+1} \sum_{k=0}^K e_u^{(k)}, \end{aligned} \quad (1)$$

其中, N_u 表示用户 u 的邻居集合, K 为 LightGCN 的层数, $e_i^{(k)}$ 表示项目第 k 层的嵌入, e_u 表示用户的最终嵌入表示。项目 i 的嵌入表示 e_i 与公式(1)类似。

LightGCN 的模型预测, 即用户 u 和项目 i 交互的可能性, 被定义为用户和项目最终表示的内积:

$$\hat{y}_{u,i} = e_u^T e_i \quad (2)$$

为了优化模型参数, 并在推荐任务中取得更好地表现, 引入贝叶斯个性化排名(BPR, Bayesian Personalized Ranking)损失[25]。BPR 损失使得已观察到的交互行为的预测分数高于未观察到的交互行为, 公式如下:

$$\mathcal{L}_{BPR} = - \sum_{(u,i,j) \in \mathcal{O}} \log \sigma(\hat{y}_{u,i} - \hat{y}_{u,j}) \quad (3)$$

其中, $\sigma(\cdot)$ 是 Sigmoid 函数, $\mathcal{O} = \{u, i, j | R_{u,i} = 1, R_{u,j} = 0\}$ 表示成对训练数据, (u, i) 表示已观察到的正样本, 即 u 与 i 发生交互, (u, j) 表示未观察到的负样本, 即 u 与 j 未发生交互。BPR 损失使模型学习的正样本的预测分数高于负样本的预测分数。

3.3. 节点对比学习

在节点级别的对比学习上, 选用结构邻域对比学习的方法。此方法将每个用户(或项目)与它的结构邻居进行对比。当用户与项目产生交互时, 更有可能选择与自己较为相似的其他用户所交互过的项目, 又由于用户项目交互图 G 是一个天然的二分图, 其中节点与其偶数跳节点具有相似性。

通过这种方式, 可以从图编码器的偶数层(例如 2、4、6)输出中获得齐次邻域表示。根据这些表示可以有效地对用户或项目与其同构结构邻居之间的关系进行建模。将用户自身的嵌入和其偶数层的对应嵌入视为正对, 其他视为负对。在对比损失 InfoNCE [26]的基础上, 提出了节点对比学习损失函数:

$$\mathcal{L}_S^U = \sum_{u \in U} -\log \frac{\exp((e_u^{(k)} \cdot e_u^{(0)} / \tau))}{\sum_{v \in U} \exp((e_u^{(k)} \cdot e_v^{(0)} / \tau))} \quad (4)$$

其中 $e_u^{(k)}$ 是图编码器第 k 层的嵌入输出且 k 是偶数, $e_u^{(0)}$ 和 $e_v^{(0)}$ 分别表示用户 u 和用户 v 的初始嵌入($u \neq v$), τ 是 softmax 的温度超参数。同样的方式可以得到项目的结构对比损失函数:

$$\mathcal{L}_S^I = \sum_{i \in I} -\log \frac{\exp((e_i^{(k)} \cdot e_i^{(0)} / \tau))}{\sum_{j \in I} \exp((e_i^{(k)} \cdot e_j^{(0)} / \tau))} \quad (5)$$

完整的节点对比损失函数是上述两种损失的加权和:

$$\mathcal{L}_S = \mathcal{L}_S^U + \alpha \mathcal{L}_S^I \quad (6)$$

其中, α 是一个超参数, 用来平衡结构对比学习中两个损失的权重。

3.4. 子图对比学

与节点级和全图级对比学习相比, 子图级对比学习更侧重于局部结构信息的学习。它在原始图上采样子图, 并在子图粒度上进行对比学习。本文采用了一种基于随机游走[27]的方法来生成子图。随机游走策略可以保留用户-项目交互图 G 的局部结构信息和连接关系, 从而揭示网络中相互连接紧密或属性相似的节点聚集区域。

具体而言, 本文的随机游走策略是在每个图卷积网络的层级上都独立地生成一个子图[13], 这与传统的所有层共享一个子图的方法有所不同[27] [28]。为了达到这一目的, 本文使用对层感知的掩码矩阵实现对每层进行不同的边丢弃操作, 从而生成随机游走路径, 以此实现从节点出发随机游走生成子图的目的。公式如下:

$$\begin{aligned} s1(G) &= (V, M_1^{(l)} \odot E), \\ s2(G) &= (V, M_2^{(l)} \odot E), \end{aligned} \quad (7)$$

其中, $s1(G)$ 、 $s2(G)$ 表示随机游走生成的子图, $M_1^{(l)}, M_2^{(l)} \in \{0, 1\}^{|E|}$ 分别是第 L 层的两个执行随机边丢弃的掩码矩阵。

在建立了原图的子图扩展视图之后, 子图对比学习方法将同一节点在不同视图中的嵌入视为正样本对(比如, $\{(e'_u, e''_u) | u \in U\}$), 而将不同节点在各自视图中的嵌入视为负对(比如, $\{(e'_u, e''_v) | u, v \in U, u \neq v\}$)。通过对比学习, 鼓励同一节点在不同视图下的特征一致, 并强调不同节点间的特征差异。用户的子图对比学习损失函数如下式(8)。同理可得项目的子图对比损失函数。结合两个损失, 可得子图对比学习的损失函数如下式(9):

$$\mathcal{L}_M^U = \sum_{u \in U} -\log \frac{\exp((e'_u, e''_u)/\tau)}{\sum_{v \in U} \exp((e'_u, e''_v)/\tau)} \quad (8)$$

$$\mathcal{L}_M = \mathcal{L}_M^U + \alpha \mathcal{L}_M^I \quad (9)$$

其中, α 是一个超参数, 用来平衡子图对比学习中用户和项目损失的权重。

3.5. 全图对比学

全图对比学习区别于基于节点的对比学习方法, 旨在捕捉整个图的全局特征, 以实现对图整体结构和内在信息的宏观理解。与上一节讨论的子图对比学习相比, 全图对比学习不局限于子图范围内的结构和特性, 而是专注于学习和比较整个图的一致性表示和综合特征, 为理解 and 应用图数据提供了一个更加全面的视角。

具体而言, 在用户-项目交互图 G 的基础上, 通过随机游走算法生成两个结构不同的子图 $s1(G)$ 和 $s2(G)$, 如公式(7)所示。然后在两个子图上分别利用 LightGCN 编码器前向传播得到整个子图的表示:

$$\begin{aligned} Z_{p1} &= \text{LightGCN}(s1(G)), \\ Z_{p2} &= \text{LightGCN}(s2(G)), \end{aligned} \quad (10)$$

其中, Z_{p1} 和 Z_{p2} 表示子图 $s1(G)$ 和 $s2(G)$ 的整体图嵌入。

对于在全图对比学习中如何划分正负样本, 主要基于两个考虑: 1. 正样本需要保证代表相似的结构, 才能学习到有效的图表示。随机游走保持了原图大部分结构, 可视为一个干扰不大的视图, 所以其表示可作为正样本。2. 负样本需要有一定结构差异。原图不同采样带来的结构变化较大, 所以其表示可作为负样本。所以在本节中, 采用“相似子图-原图采样”的组合, 正样本对由相似子图构成, 负样本对由原图采样构成, 可以保证两者结构上有一定区分度。具体负采样公式如下:

$$Z_k = \text{LightGCN}(G, \text{dropout} = \rho_k) \quad (11)$$

其中, Z_k 表示在图 G 上进行第 k 次采样所生成的整体图嵌入, ρ_k 表示第 k 次采样时的 dropout 比例。将这样获得的 $\{Z_1, Z_2, \dots, Z_k\}$ 作为负样本, 全图对比损失函数如下:

$$\mathcal{L}_L = -\log \frac{\exp((Z_{p1} \cdot Z_{p2} / \tau))}{\sum_{k=1}^K \exp((Z_{p1} \cdot Z_k / \tau))} \quad (12)$$

3.6. 整体损失函数

为了帮助推荐模型更有效地对用户和项目之间的交互进行建模, 本文将上述提及的多粒度对比学习损失作为补充, 并利用多任务学习策略与 BPR 损失相结合, 得到的模型整体损失函数如下:

$$\mathcal{L}_{total} = \mathcal{L}_{BPR} + \lambda_1 \mathcal{L}_S + \lambda_2 \mathcal{L}_M + \lambda_3 \mathcal{L}_L + \lambda_4 \|\Theta\|_2 \quad (13)$$

其中 λ_1 、 λ_2 、 λ_3 和 λ_4 分别是控制所提出的不同粒度对比学习权重和正则化项权重的超参数, 而 Θ 表示模型参数的集合。

4. 实验结果与分析

4.1. 实验设置

4.1.1. 数据集

本文采用两个公开数据集 Yelp [24] 和 Amazon-Book [13] 来进行实验。Yelp 是一个公开可用的点评网站数据集, 该数据集包含了来自 8 个主要城市的超过 5 万个商家, 150 万条评论和社交关系。Amazon-Book 是一个公开的大规模数据集, 主要包含了 Amazon 网站上用户对书籍的评分数据。表 1 显示了这两个数据集的统计信息。将数据集以 7:1:2 的比率划分成训练集、验证集和测试集。

Table 1. Dataset statistics

表 1. 数据集统计信息

数据集	用户数	项目数	交互	稀疏度
Yelp	19,539	21,266	450,884	0.11%
Amazon-Book	52,463	91,599	2,984,108	0.11%

4.1.2. 评价指标

本文采取 topK 推荐方法进行推荐, 其中 $K = 20$, 采用推荐系统任务中常用的召回率 Recall 和归一化折损累计增益 NDCG (Normalized Discounted Cumulative Gain) 作为性能评价指标。Recall 是衡量 topK 推荐系统性能的指标之一, 用于量化推荐列表中正确包含的用户历史交互商品数量占用户在测试集中所

有历史交互商品数量的比例。召回率越高模型推荐效果越好。假定测试集全部用户集合为 U ，对任意用户 $u \in U$ ，topK 的推荐列表为 L_u ，测试集中用户 u 的真实交互列表为 L_u^{test} ，则模型召回率的计算公式为：

$$recall@K = \sum_{u \in U} \frac{1}{|U|} \frac{|L_u \cap L_u^{test}|}{|L_u^{test}|} \quad (14)$$

NDCG 衡量了推荐列表中不同位置推荐结果的相关性。NDCG 越高，意味着对用户更相关的项目被放在了推荐列表的前列，说明推荐性能越好。公式如下：

$$NDCG_K = \frac{DCG_K}{IDCG_K} \quad (15)$$

$$DCG_K = rel_1 + \sum_{i=2}^K \frac{rel_i}{\log_2(i+1)} \quad (16)$$

$$IDCG_K = \sum_{i=1}^{|REL|} \frac{2^{rel_i} - 1}{\log_2(i+1)} \quad (17)$$

其中， K 表示最有可能向用户推荐的前 K 个项目， rel_i 表示处于位置 i 的推荐项目的相关性， $|REL|$ 表示相关性最高的前 K 个项目的集合。

4.1.3. 对比模型

为了综合评估 MGGCL 的性能，本文采用如下对比模型。这些模型都是近年来一些经典的、有代表性的模型：

- (1) BPRMF [29]: 一种在矩阵分解框架下优化 BPR 损失函数的经典协同过滤算法。
- (2) NeuMF [30]: 一种用多层感知器代替矩阵分解模型中点积的推荐算法。
- (3) NGCF [3]: 一种基于图的协同过滤方法，使用标准的图卷积网络，并在消息传递过程中将二阶特征交互编码到信息中。
- (4) LightGCN [24]: 一种简单而高效的推荐算法，在 NGCF 基础上去掉了特征变换和非线性操作，保留了图卷积网络中的邻域聚合与线性传播用于推荐。
- (5) SGL [13]: 引入了自监督学习，对用户 - 项目交互图使用结构扰动生成对比视图，进行对比学习。本文采用 SGL-RW 作为 SGL 的实例化。
- (6) MGGCL: 本文提出的基于多粒度的图对比学习推荐模型。

4.1.4. 参数设置

为了进行公平地比较，本文所有对比模型均采用其原始论文中给出的最佳参数，并使用 SELFREC [31] 框架实现了所有模型。SELFREC 框架是一个用于自监督推荐的 Python 框架，集成了常用的数据集和指标。在实验过程中，使用 Adam [32] 优化器对所有方法进行优化，在所有嵌入上都使用了 Xavier [33] 初始化。嵌入大小为 64，batch size 为 2048，学习率为 10^{-3} ，温度超参数 τ 为 0.05，dropout 的概率为 0.1，超参数 λ_1 为 10^{-6} ，超参数 λ_2 、 λ_3 为 10^{-7} ，超参数 α 为 1，超参数 λ_4 为 10^{-4} 。

4.2. 实验结果与分析

4.2.1. 性能对比

本文提出的 MGGCL 算法与其他经典推荐算法在两个真实数据集上的实验结果如表 2 所示。其中 Improv. 代表本文算法相较于现有最优基线算法在各项指标上的提升比例。为了更形象直观，实验结果用

图 2 进行了对比显示。

Table 2. Performance comparison of different recommendation models

表 2. 不同推荐模型的性能比

模型	Yelp		Amazon-Book	
	Recall@20	NDCG@20	Recall@20	NDCG@20
BPRMF	0.0433	0.0354	0.0250	0.0196
NeuMF	0.0451	0.0363	0.0258	0.0200
NGCF	0.0579	0.0477	0.0337	0.0261
LightGCN	0.0639	0.0525	0.0410	0.0315
SGL	0.0661	0.0545	0.0453	0.0352
MGGCL	0.0689	0.0567	0.0484	0.0386
Improv.	4.24%	4.04%	6.85%	9.66%

观察实验结果可以发现, 传统方法如 BPRMF 在数据集上表现不好, 这表明内积不足以捕获用户和项目之间的复杂关系。NeuMF 在所有情况下都优于 BPRMF, 证明了用户和项目嵌入之间的非线性特征交互的重要性。但以上两种方法都没有显式地对嵌入学习过程中的连通性进行建模, 这很容易导致次优的表示结果。NGCF 直接利用了图卷积网络对用户-项目的高阶连接性进行建模, 大大提升了推荐性能。LightGCN 取得的结果优于 NGCF, 这表明去掉了特征变换和非线性操作带来的简化架构的有效性和健壮性。SGL 在两个数据集上的性能一致优于其他方法, 这表明对比学习对提高推荐性能是有效的。但是 SGL 只从一个维度上进行对比学习, 并没有从多个粒度上把握图的全局结构。

如表 2、图 2 所示, 本文提出的 MGGCL 模型对比各基准算法在 Recall@20 和 NDCG@20 指标上均取得了最好的效果。在两个数据集上, MGGCL 较基准模型中表现最好的模型(SGL)在 Recall@20 指标上分别提升了 4.24%、6.85%, 在 NDCG@20 指标上分别提升了 4.04%、9.66%。这种优势主要是由多粒度的图对比学习所带来的, MGGCL 可以在不同粒度下考虑图的结构信息, 以更加整体和全面的视角利用图的全局结构。

4.2.2. 消融实验

为了分析 MGGCL 中各组件的性能, 验证不同粒度下的对比学习模块对改进推荐性能的有效性, 定义了以下算法与 LightGCN 和 MGGCL 进行对比实验:

- (1) V1: 在原模型中去掉节点对比学习模块
- (2) V2: 在原模型中去掉子图对比学习模块
- (3) V3: 在原模型中去掉全图对比学习模块

在两个数据集 Yelp 和 Amazon-Book 上的消融实验结果如图 3 所示。由图 3 可知, 删除每一种不同粒度的对比学习模块都会导致 MGGCL 性能的下降, 但三个变体的性能都好于基准 LightGCN。这证明了三种不同粒度的对比学习模块对改进图推荐模型的有效性, 并说明了每个模块各自的重要性, 它们相辅相成, 在不同方面提升了整体的性能。

4.2.3. 超参数敏感性分析

- (1) 温度超参数 τ

如上文所述, 式(4)、式(5)、式(8)和式(12)中定义的温度超参数 τ 在对比学习中起着重要作用。 τ 主要用于调整对比损失函数中的 softmax 函数, 从而影响模型对不同样本间差异的敏感度[34]。为了分析温度

超参数对 MGGCL 的影响, 将 τ 的范围设置在 0.01 到 0.10 之间, 在 Yelp 数据集上分析温度超参数 τ 的敏

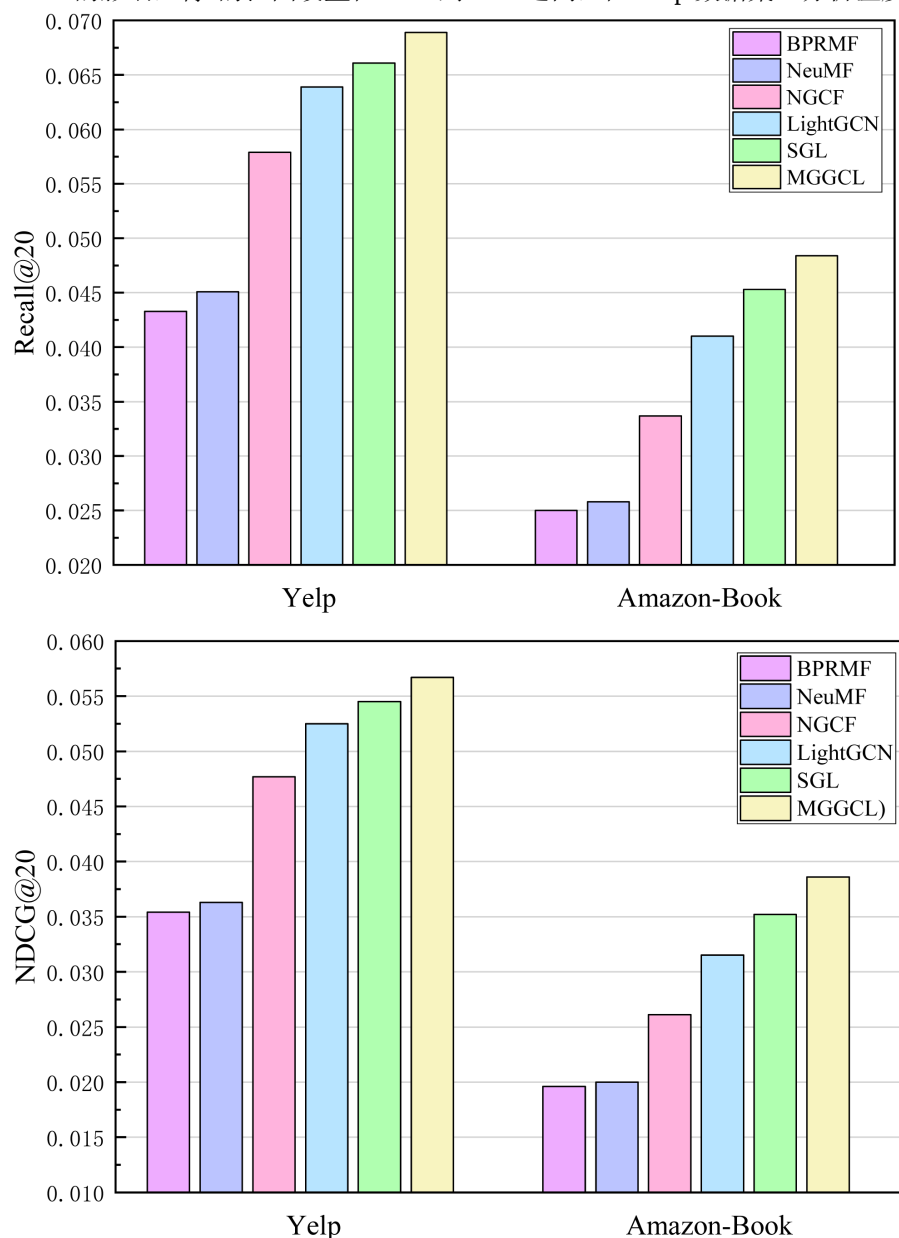


Figure 2. Performance comparison of different recommendation models

图 2. 不同推荐模型的性能比

感性, 实验结果如图 4 所示。

可以观察到, 将 τ 的值设置太大(例如, 0.100)会导致模型性能下降, 并且增加模型收敛所需的训练周期数。这表明模型在区分难以辨认的负样本(hard negatives)与易于辨认的负样本(easy negatives)方面的能力不足。将 τ 设置为太小的值(例如, 0.010), 也会对模型性能产生不利影响。少数负样本会对优化过程产生过大的影响, 导致多负样本场景下的梯度多样性丧失, 这是自监督学习目标的关键优势。本实验中将温度超参数设置成 0.05, 可以产生良好的推荐性能。

(2) 超参数 α

在上文公式(6)、公式(9)中, 超参数 α 用来平衡不同粒度下对比学习中用户和项目损失的权重。为了

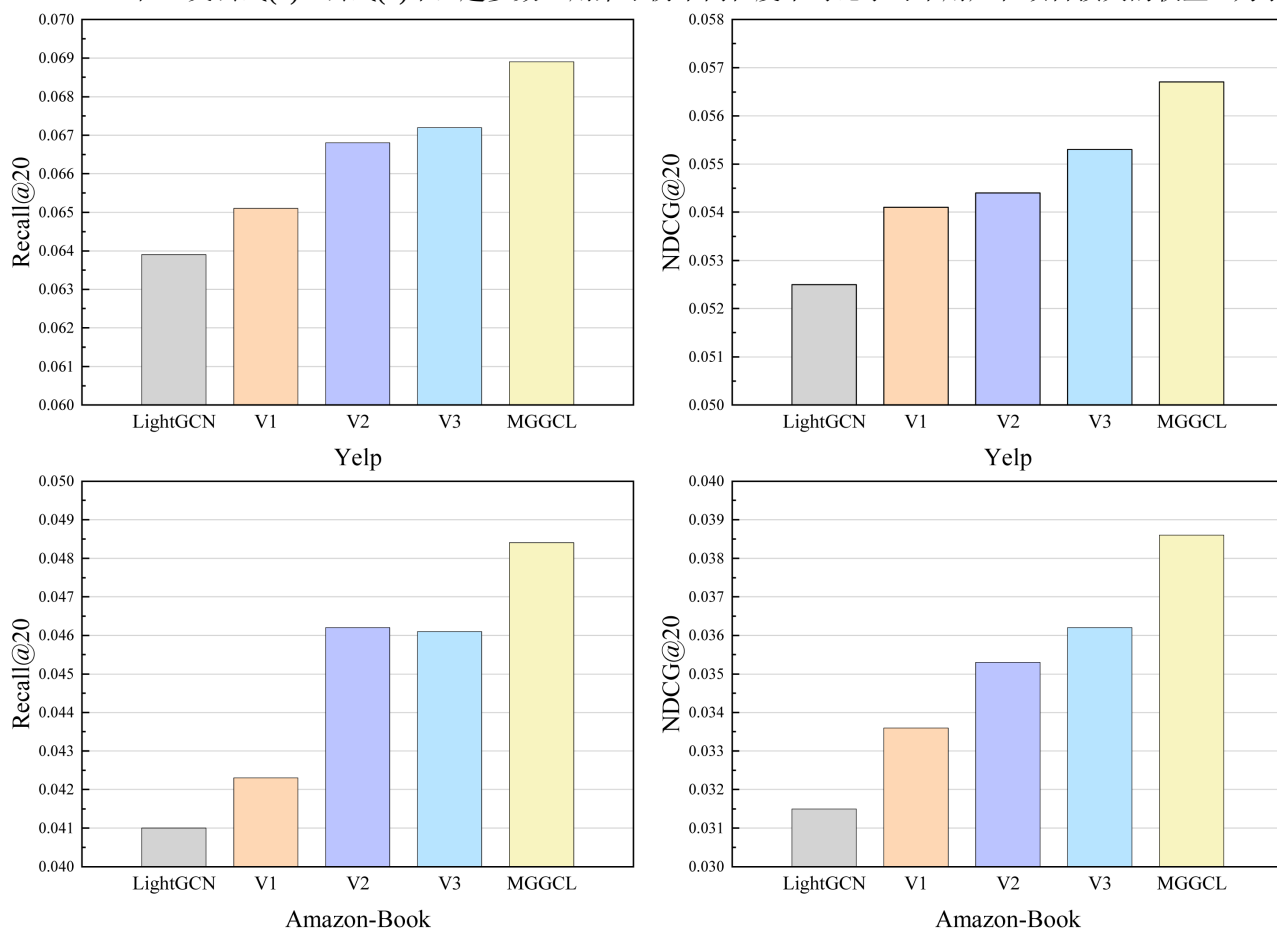


Figure 3. Results of ablation experiment

图 3. 消融实验结果

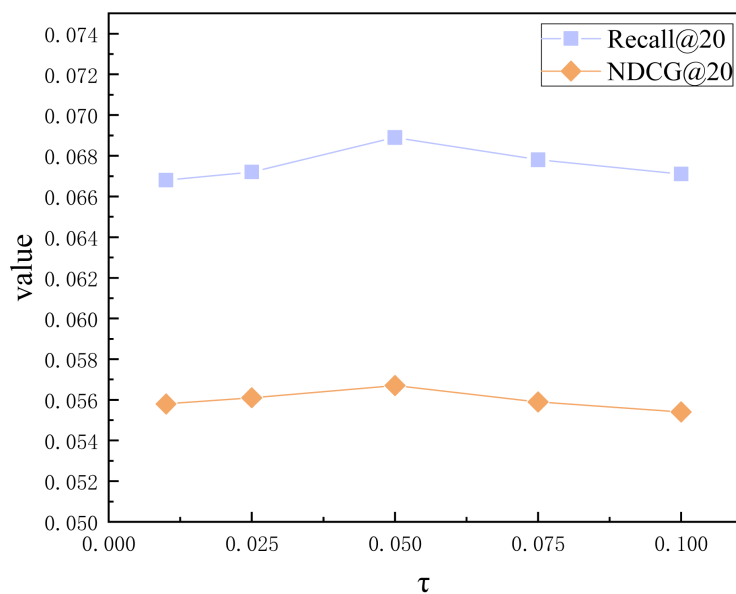
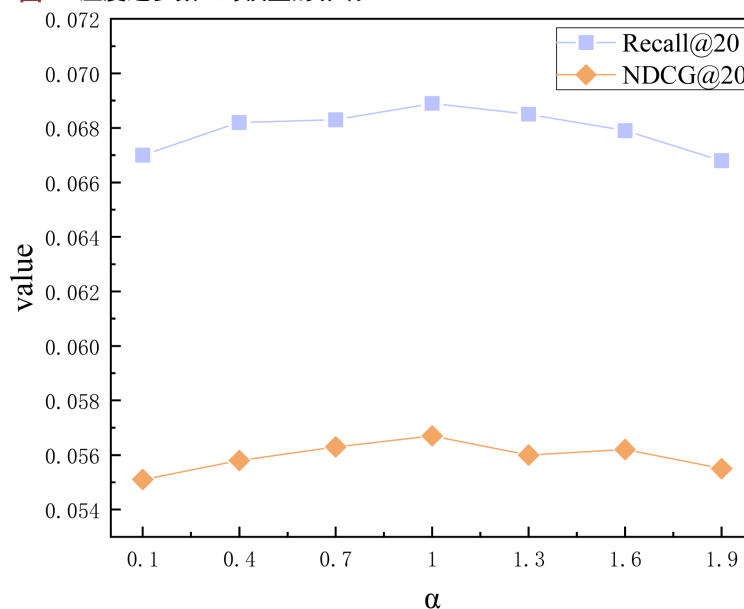


Figure 4. The impact of the temperature hyperparameter τ on the model**图 4.** 温度超参数 τ 对模型的影响**Figure 5.** The impact of the hyperparameter α on the model**图 5.** 超参数 α 对模型的影响

分析 α 对模型性能的影响, 将 α 的值在 0.1 到 2 的范围内进行调整, 在 Yelp 数据集上分析超参数 α 的敏感性, 实验结果如图 5 所示。

可以观察到, 当超参数 α 的值设置为 1 时, 在 Yelp 数据集上的两个指标均能表现出最好的性能。这表明在 MGGCL 中用户和项目的对比损失被认为同等重要, 还意味着模型在用户和项目的表征学习中达到了一种协调, 从而使得推荐算法在捕捉用户的偏好和项目的特性方面均匀有效。

5. 结束语

本文提出了一种名为 MGGCL 的图对比学习推荐算法, 采用多粒度策略, 结合节点级、子图级和全图级对比学习, 有效建模用户和项目, 克服了现有模型局限于单一粒度的问题。在未来的工作中, 需要进一步探索在不同粒度下更优的图对比学习策略, 以适应更多种类的推荐任务和数据环境。

参考文献

- [1] Ricci, F., Rokach, L. and Shapira, B. (2010) Introduction to Recommender Systems Handbook. In: Ricci, F., Ed., *Recommender Systems Handbook*, Springer US, Boston, 1-35. https://doi.org/10.1007/978-0-387-85820-3_1
- [2] Balabanović, M. and Shoham, Y. (1997) Fab: Content-Based, Collaborative Recommendation. *Communication of the ACM*, **40**, 66-72. <https://doi.org/10.1145/245108.245124>
- [3] Wang, X., He, X., Wang, M., et al. (2019) Neural Graph Collaborative Filtering. *SIGIR'19: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Paris, 21-25 July 2019, 165-174. <https://doi.org/10.1145/3331184.3331267>
- [4] Lee, J., Lee, I. and Kang, J. (2019) Self-Attention Graph Pooling. *International Conference on Machine Learning*. PMLR, Long Beach, 9-15 June 2019, 3734-3743.
- [5] Liu, W., Su, J., Chen, C., et al. (2021) Leveraging Distribution Alignment via Stein Path for Cross-Domain Cold-Start Recommendation. *Advances in Neural Information Processing Systems*, **34**, 19223-19234.
- [6] Manessi, F. and Rozza, A. (2021) Graph-Based Neural Network Models with Multiple Self-Supervised Auxiliary Tasks. *Pattern Recognition Letters*, **148**, 15-21. <https://doi.org/10.1016/j.patrec.2021.04.021>

- [7] Hu, Z., Dong, Y., Wang, K., *et al.* (2020) Gpt-Gnn: Generative Pre-Training of Graph Neural Networks. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 6-10 July 2020, 1857-1867. <https://doi.org/10.1145/3394486.3403237>
- [8] Zhu, Y., Xu, Y., Liu, Q., *et al.* (2021) An Empirical Study of Graph Contrastive Learning. *35th Conference on Neural Information Processing Systems (NeurIPS 2021)*, 6-14 December 2021, 1-13.
- [9] Chen, T., Kornblith, S., Norouzi, M., *et al.* (2020) A Simple Framework for Contrastive Learning of Visual Representations.
- [10] Gao, T., Yao, X. and Chen, D. (2021) SimCSE: Simple Contrastive Learning of Sentence Embeddings. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, November 2021, 6894-6910. <https://doi.org/10.18653/v1/2021.emnlp-main.552>
- [11] Veličković, P., Fedus, W., Hamilton, W.L., *et al.* (2018) Deep Graph Infomax. arXiv preprint arXiv:1809.10341.
- [12] Zhu, Y., Xu, Y., Yu, F., *et al.* (2021) Graph Contrastive Learning with Adaptive Augmentation. *Proceedings of the Web Conference*, Ljubljana, 19-23 April 2021, 2069-2080. <https://doi.org/10.1145/3442381.3449802>
- [13] Wu, J., Wang, X., Feng, F., *et al.* (2021) Self-Supervised Graph Learning for Recommendation. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 11-15 July 2021, 726-735. <https://doi.org/10.1145/3404835.3462862>
- [14] Yu, J., Yin, H., Xia, X., *et al.* (2021) Are Graph Augmentations Necessary? Simple Graph Contrastive Learning for Recommendation. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Madrid, 11-15 July 2022, 1294-1303. <https://doi.org/10.1145/3477495.3531937>
- [15] Lin, Z., Tian, C., Hou, Y., *et al.* (2022) Improving Graph Collaborative Filtering with Neighborhood-Enriched Contrastive Learning. *Proceedings of the ACM Web Conference*, Lyon, 25-29 April 2022, 2320-2329. <https://doi.org/10.1145/3485447.3512104>
- [16] Cai, X., Huang, C., Xia, L., *et al.* (2023) LightGCL: Simple Yet Effective Graph Contrastive Learning for Recommendation.
- [17] Jiang, Y., Huang, C. and Huang, L. (2023) Adaptive Graph Contrastive Learning for Recommendation. *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Long Beach, 6-10 August 2023, 4252-4261. <https://doi.org/10.1145/3580305.3599768>
- [18] Hajij, M., Zamzmi, G., Papamarkou, T., *et al.* (2022) Topological Deep Learning: Going Beyond Graph Data.
- [19] Zhu, Y., Xu, Y., Yu, F., *et al.* (2020) Deep Graph Contrastive Representation Learning.
- [20] Chi, H. and Ma, Y. (2022) Enhancing Graph Contrastive Learning with Node Similarity.
- [21] You, Y., Chen, T., Sui, Y., *et al.* (2020) Graph Contrastive Learning with Augmentation. *Advances in Neural Information Processing Systems*, **33**, 5812-5823.
- [22] Sun, F.Y., Hoffmann, J., Verma, V., *et al.* (2019) Infograph: Unsupervised and Semi-Supervised Graph-Level Representation Learning via Mutual Information Maximization. *8th International Conference on Learning Representations, ICLR 2020*, Addis Ababa, 26-30 April 2020, 2250-2266.
- [23] You, Y., Chen, T., Shen, Y., *et al.* (2021) Graph Contrastive Learning Automated. *International Conference on Machine Learning*. PMLR, 18-24 July 2021, 12121-12132.
- [24] He, X., Deng, K., Wang, X., *et al.* (2020) Lightgcn: Simplifying and Powering Graph Convolution Network for Recommendation. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Xi'an, 25-30 July 2020, 639-648. <https://doi.org/10.1145/3397271.3401063>
- [25] Rendle, S., Freudenthaler, C., Gantner, Z., *et al.* (2012) BPR: Bayesian Personalized Ranking from Implicit Feedback.
- [26] Oord, A., Li, Y. and Vinyals, O. (2018) Representation Learning with Contrastive Predictive Coding.
- [27] Perozzi, B., Al-Rfou, R. and Skiena, S. (2014) Deepwalk: Online Learning of Social Representations. *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 24-27 August 2014, 701-710. <https://doi.org/10.1145/2623330.2623732>
- [28] Grover, A. and Leskovec, J. (2016) Node2vec: Scalable Feature Learning for Networks. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, 13-17 August 2016, 855-864. <https://doi.org/10.1145/2939672.2939754>
- [29] Rendle, S., Freudenthaler, C., Gantner, Z., *et al.* (2014) Bayesian Personalized Ranking from Implicit Feedback. *Proceedings of Uncertainty in Artificial Intelligence*, Quebec, 23-27 July 2014, 452-461.
- [30] He, X., Liao, L., Zhang, H., *et al.* (2017) Neural Collaborative Filtering. *Proceedings of the 26th International Conference on World Wide Web*, Perth, 3-7 April 2017, 173-182. <https://doi.org/10.1145/3038912.3052569>
- [31] Yu, J., Xia, X., Chen, T., *et al.* (2023) XSimGCL: Towards Extremely Simple Graph Contrastive Learning for Rec-

- ommendation. *IEEE Transactions on Knowledge and Data Engineering*, **36**, 913-926.
<https://doi.org/10.1109/TKDE.2023.3288135>
- [32] Kingma, D. P. and Ba, J. (2014) Adam: A Method for Stochastic Optimization.
 - [33] Glorot, X. and Bengio, Y. (2010) Understanding the Difficulty of Training Deep Feed forward Neural Networks. *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, Sardinia, 13-15 May 2010, 249-256.
 - [34] Wang, F. and Liu, H. (2020) Understanding the Behaviour of Contrastive Loss. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, 20-25 June 2021, 2495-2504.