# 用于序列推荐的图掩码自编码器

Yaowen Ye
elwin@connect.hku.hk
香港大学 中国香港特别行政区

Lianghao Xia 又名_
xia@foxmail.com 香港大学 中国香港特别行政区

Chao Huang*
chaohuang75@gmail.com 香港大学 中国香港特别行政区

arXiv:2305.04619v3 [cs.IR] 1 Jun 2023

## 摘要

尽管一些强大的神经网络架构（例如 Transformer、图神经网络）通过高阶物品依赖建模，在序列推荐中取得了更好的性能，但在标签稀缺的场景下，它们可能缺乏良好的表示能力。为解决标签不足的问题，近期方法中的对比学习（CL）备受关注，通过嵌入对比进行数据增强以实现自监督。然而，由于其对比视图生成策略依赖人工设计，现有的对比学习增强模型：i）难以在多种序列推荐任务中取得稳定一致的性能；ii）可能无法抵御用户行为数据噪声。有鉴于此，我们提出了一种简单而有效的图掩码自编码器增强的序列推荐系统（MAERec），能够自适应、动态地提炼全局物品转移信息，用于自监督增强。该方法天然避免了上述对高质量嵌入对比视图构建的严重依赖问题。为此，我们设计了一种自适应数据重构范式，并将其与长程物品依赖建模相结合，以在序列推荐中实现信息丰富的数据增强。大量实验表明，我们的方法显著优于最先进的基线模型，并且能够在面对数据噪声和稀疏性时学习到更准确的表示。我们实现的模型代码可在 https://github.com/HKUDS/MAERec 获取。

## CCS 概念

• **信息系统** →**推荐系统**。

## 关键词

序列推荐，掩码自编码器，图神经网络，自监督学习

**ACM 参考格式：**
Yaowen Ye、Lianghao Xia 和 Chao Huang。2023。《面向序列推荐的图掩码自编码器》。收录于第*46*届 *ACM* 信息检索研究与发展国际会议论文集（*SIGIR'23*），*2023*年*7*月*23*日至*27*日，中国台湾台北。ACM，中国台湾台北，10页。https://doi.org/10.1145/3539618.3591692

*Chao Huang 是通讯作者。

允许免费制作本作品全部或部分内容的数字或纸质副本，供个人或课堂使用，但前提是不得出于盈利或商业利益而制作或分发副本，且副本首页必须附有本声明及完整的引用信息。对于本作品中由作者以外的其他方拥有版权的组成部分，必须尊重其版权。注明出处后，允许摘录。若要以其他方式复制、重新出版、发布到服务器或分发至列表，则必须事先获得明确许可和/或支付相应费用。请通过 permissions@acm.org 申请授权。*SIGIR'23*，*2023*年*7*月*23*日至*27*日，中国台湾台北© 2023 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/10.1145/3539618.3591692

## 1 引言

序列推荐旨在学习用户兴趣的有效表示，并为不同用户推荐他们可能感兴趣的未来物品 [43–45]。鉴于现实生活中的用户偏好会随时间变化，例如在电子商务 [34] 和视频流媒体 [37] 网站上，这项任务已受到广泛关注。为了捕捉物品间的高阶转移关系，研究人员进行了大量探索 [5, 25, 35, 38] ，致力于提出各种图神经网络，通过在相邻物品之间递归传播信息来改进推荐效果。

序列推荐系统中的用户行为序列通常呈现长尾分布，即大量用户仅与极少数物品交互 [20, 23]。尽管图神经网络（GNN）在全监督方式下处理序列推荐任务方面取得了可喜成果，但标签数据稀疏问题可能会显著降低模型的表征性能 [40, 47]。为解决标签不足问题，近期已有研究尝试 [24, 26, 40] 将对比学习（CL）的优势引入序列推荐系统，提供辅助自监督信号作为增强信号。遵循互信息最大化（InfoMax）框架，这些对比学习方法的核心思想是基于InfoNCE目标，使从两个增强视图编码得到的嵌入达成一致性。它们采用各种增强方案，根据不同启发式规则破坏物品序列结构，例.如，CL4SRec [40], 中的随机项目掩码和重排序、DouRec [26], 中基于丢弃的语义保持，以及 CoSeRec [24]中的替换和插入操作。

尽管近期取得了进展，现有基于对比学习的序列推荐系统仍受到以下因素的严重限制：
**(1)手工设计的对比增强**。最先进的对比学习（CL）方法仍然严重依赖基于启发式方法手动设计的数据破坏方案，以构造用于嵌入对比的视图。随后，最大化正样本对内部的表示一致性，并最小化负样本之间的表示一致性 [40]。在 InfoMax 原则的内在设计指引下，当前对比式自监督的成功在很大程度上依赖于高质量的增强和准确的对比视图生成。然而，这通常需要领域知识并耗费大量人力，难以适应监督有限的多样化序列推荐场景。此外，盲目破坏序列结构以生成增强视图，可能会损害重要的转移结构，并削弱短序列上的表示学习效果。

**(2) 用于数据增强的噪声扰动**。数据噪声问题，例如交互行为噪声和流行度偏差，普遍存在于现实生活中的推荐系统中 [3, 30, 39]。经过训练的

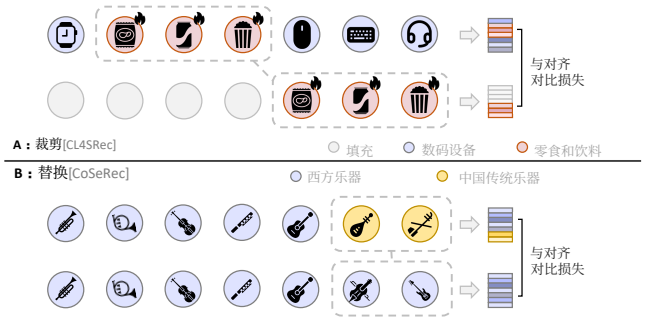



**图 1：现有基于 CL 的序列推荐器中存在噪声和偏差的数据增强示例。**

自监督学习（SSL）模型很容易受到虚假的物品转移相关性的误导，进而导致性能不佳。尤其是，噪声信息可能沿着物品之间的连接传播，从而对采用对比式数据增强技术的表征学习产生负面影响。面对数据噪声问题，对比式自监督学习模型很可能会对含噪标签过拟合，导致性能下降。因此，为了构建能够在实际场景中抵御噪声扰动的经自监督学习增强的序列推荐模型，我们需要通过自适应且选择性的数据增强来缓解自监督过程中的噪声影响，从而对交互动态进行编码。

以图 1 所示的现有基于 CL 的序列推荐器所采用的增强策略为例，用户 A 是一位数字产品爱好者，购买了许多数字产品。受在线零售商新年促销的影响，他/她还购买了一些热门零食和饮料，这归因于物品的位置偏差。然而，CL4SRec [40] 中的裁剪操作可能会丢弃重要的交互数据，并保留有偏信息用于增强。在这种情况下，噪声增强视图之间的对比对齐不可避免地会损害自监督质量，导致用户偏好建模产生误导。类似地，对用户 B 的物品序列进行替换增强（CoSeRec [24]中）可能会丢失关于长尾物品（例.如，二胡、琵琶）的有限但重要的交互数据。其他一些更受欢迎的乐器（例.如，吉他）会被用来进行替换，这可能会降低对交互标签有限的长尾物品的推荐效果。

鉴于上述局限性，一个有趣的问题自然浮现：如何构建易于适配且抗噪的自监督学习增强型序列推荐系统？受近期基于自编码器的掩码技术在图像数据增强领域取得成功的启发，[11, 13], 以掩码数据重建为目标的生成式自监督学习能够自然地减少对人工构造高质量对比视图的严重依赖，从而实现精准的嵌入对齐。

**贡献**。在本研究中，我们提出了一种新方法，通过鲁棒且易于适配的自监督学习来增强序列推荐系统。为此，我们提出了一种轻量且有原则的图掩码自编码器模型（命名为 MAERec），以实现自监督学习增强过程的自动化。MAERec 能够根据不断变化的序列推荐环境，自适应且动态地提取用于重建的有效信号。

首先，为了减轻增强过程中的噪声影响，我们自动识别语义一致性更高的物品，通过一种可学习的掩码方案。我们引入任务自适应正则化，以增强对下游推荐任务具有感知能力的自监督。其次，我们有选择地掩码掉更具帮助性的物品转移路径，并将其输入图自编码器进行重构。通过这种方式，自适应增强与序列物品依赖关系的建模无缝结合，从而消除交互噪声和与任务无关的信息所引入的偏差。最后，根据 MAERec 的自适应增强，我们使用抗噪的自监督学习信号补充主要推荐任务。

通过这种方式，我们的 MAERec 模型不仅能够进行强力增强，以训练鲁棒的序列推荐器，还能通过生成式重建任务，利用适应任务的语义来调节自监督学习。我们的模型无需启发式方法，并且能够泛化到不同数据集上的各种序列交互模式。在各种设置下进行的大量实验表明，与多种基线方法相比，我们的算法具有优越性。模型消融实验和鲁棒性研究证实，自适应 SSL 增强能够减轻数据噪声和稀疏性的影响。此外，与基线方法相比，我们的方法也实现了相当的效率。

## 2 预备知识与相关工作

**序列推荐**。假设 U 和 V 分别表示用户集合和物品集合。每个用户（$u \in \mathrm{U}$）都关联着一个按时间顺序排列的历史交互物品序列，记为 $S^u = (s_1^u, s_2^u, \cdots \ s_{l_u}^u)$。$s_t^u \in \mathrm{V}$ 表示用户 $u$ 交互的第 $t$ 个物品，$l_u$ 是 $u$ 的物品序列 $S^u$ 的长度。给定用户的交互序列 $S^u$，序列推荐的目标是预测用户 $u$ 最有可能交互（例如，点击或购买）的下一个物品 $s_{l_u+1}^u$。

为研究序列推荐问题，研究人员开发了大量方法，采用不同技术来编码序列行为模式，例如，循环神经网络（RNN）[15], 卷积神经网络 [29, 41]。受Transformer架构的启发，自注意力机制已被用于处理用户行为序列中的成对物品相关性 [9, 10, 19]。此外，随着图神经网络（GNN）的发展，研究人员基于图神经网络框架构建了图增强表示编码器，通过多层消息传递来捕捉用户的长期序列偏好，例如SRGNN [38], GCE-GNN [35], 和SURGE [2]。然而，大多数现有的监督方法都面临标签稀疏问题，这严重限制了模型在现实生活中的推荐系统中的表示能力。

**面向序列推荐器的对比学习（CL）**。近年来，CL 因能够应对标签监督不足的挑战 [1]并学习物品序列依赖关系的准确表示而备受关注 [22]。通常，采用对比学习的序列推荐模型旨在从未标注数据中挖掘自监督信号 [42, 46]。在数据增强方面，现有基于 CL 的推荐系统过度依赖人工设计来构造对比表示视图，以实现嵌入的一致性。

通过对用户行为序列进行各种增强操作来破坏数据，从而生成对比视图。例如，

CL4SRec [40] 通过随机对序列进行掩码和重排序操作来构建增强对比视图。DuoRec [26] 采用丢弃掩码进行模型级增强。为捕捉按时间排序的交互行为中的多样化用户意图，ICLRec [7] 提出通过对比式自监督对与用户意图相对应的潜变量进行建模。这些对比学习方法的有效性在很大程度上取决于生成的正负样本质量是否足够高，而这在不同推荐场景中并不总能得到保证。

**用于表征学习的掩码自编码器**。受生成式自监督学习在语言表征领域取得成功的启发， [8], 掩码自编码器（MAE）被提出作为理解图像的有效学习器 [11]。在这些基于 MAE 的视觉学习器中，编码器-解码器框架用于进行像素级图像重建。此外，视觉-语言掩码自编码器也已被开发出来，用于学习多模态嵌入，以配对图文实例 [13]。受先掩码后预测范式的启发 [6], , 近期研究已将 MAE 引入图表征学习。例如，GraphMAE [16] 侧重于通过随机掩码节点特征，并根据缩放余弦误差进行重建。S2GAE [28] 则随机删除输入图结构中的节点级连接，以进行重建。在此基础上，我们的方法融合了掩码自编码的优势，并通过自适应图掩码加以改进，从而为序列推荐提供有意义的自监督信号。

## 3 方法

在本节中，我们将介绍 MAERec 推荐系统，并在图 2 中展示其模型架构概览。

### 3.1 自适应转移路径掩码

不同于一些现有掩码自编码器 [16, 28], 采用的随机掩码策略，我们的图掩码机制旨在实现自适应自监督增强。通过这种方式，图掩码自编码器能够在多样化的推荐数据上，自动且动态地提炼用于掩码的有用信息。此外，为了使模型能够保留长程物品转移的全局上下文，我们提出对包含高阶物品连接的图路径信息进行掩码，而非对单个节点或边进行掩码。

为了利用跨序列物品转移信息增强我们的模型，并释放图掩码自编码器的潜力，我们在序列上构建了一个全局物品-物品转移图。

**全局物品转移图**。为捕捉不同用户之间的依赖关系，我们通过考察用户行为序列（例如， $S^u$、 $u \in \mathrm{U}$），生成一个图 G = (V，E) 来表示不同物品之间的转移关系。物品集合 V 作为转移图的顶点集。对于边集 E，我们遍历所有用户序列，并在每个序列中为每个物品与其 $h$ 跳邻居之间建立一条边。重复边只计数一次。形式上，转移图的边集构建如下：

$$\mathcal{E} = \{(s_t^u, s_{t'}^u) : u \in \mathcal{U}, |t - t'| \le h, 1 \le t, t' \le l_u\} \tag{1}$$

给定构建的物品转移图 G =(V，E)，结合可学习数据增强的整体自适应路径掩码范式策略由几个关键模块组成：**(i)学习掩码**– 自动发现用于掩码路径构建的锚节点集合；**(ii) 转移路径掩码**–对图结构中的路径进行掩码，以继承其中蕴含的信息丰富的转移模式；**(iii) 任务自适应增强**–利用任务自适应上下文重新校准自监督图掩码过程。

*3.1.1* **学习掩码**。 学习掩码模块的目标是学习一种策略，为可学习的图增强器生成锚节点集合 $\mathrm{V}_a$ 。受路径级与图级嵌入 [18, 31],之间 InfoMax 原则的启发，我们提出通过考察物品在全局物品转移图结构 G = (V，E) 上的表示一致性，来衡量物品之间的语义相关性。具体而言，我们在图 G 上采样目标物品节点 $v$ 的$k$跳转移邻居，以生成 $v$ 的转移子图，并利用生成的全局物品转移关系表示其周围上下文。

存在噪声交互或受到流行度偏差影响的物品，其嵌入分布可能较为分散（由虚假图连接编码），从而导致其语义相关性得分受到抑制。因此，我们从 G 中选取语义相关性得分较高的物品作为锚节点 $\mathrm{V}_a$ 。

为使这一方法形式化，我们将节点 $v \in \mathrm{V}$ 与其 $k^{th}$ 阶子图之间的语义相关性$\gamma(v)$定义如下：

$$\gamma(v) = \frac{1}{|\mathcal{N}_v^k|} \sum_{v' \in \mathcal{N}_v^k} \frac{\mathbf{e}_v^T \mathbf{e}_{v'}}{\|\mathbf{e}_v\| \|\mathbf{e}_{v'}\|} \tag{2}$$

其中，全局物品转移图 G 上节点 $v$ 的 $k$ 跳邻居集合记为 $\mathrm{N}_v^k$。物品 $v$ 和 $v'$ （$v' \in \mathrm{N}_v^k$）的嵌入分别记为 $\mathbf{e}_v$ 和 $\mathbf{e}_{v'}$，其中 $\mathbf{e}_v$、$\mathbf{e}_{v'} \in \mathbb{R}^d$。我们观察到，语义相关性得分较高的节点，其邻域结构一致性也较高。这表明，由此类节点生成的路径掩码能够捕获潜在的跨序列物品转移模式，同时包含较少噪声。因此，这些节点对 SSL 重构任务更有益。

**基于 InfoMax 的 SSL 适配**。为了增强学习式掩码组件在数据增强过程中的鲁棒性，我们将 Gumbel 分布噪声 [17] 引入物品语义相关性的确定过程。其形式化表示为：

$$\gamma'(v) = \gamma(v) - \log(-\log(\mu)), \quad \mu \sim \mathrm{Uniform}(0, 1). \tag{3}$$

锚点集合 $\mathrm{V}_a$ 的大小为 $\alpha$ （$|\mathrm{V}_{ac}| = \alpha$），它通过对所有节点的语义相关性进行排序，并根据排序结果确定其选择概率而构建。

为了提升学习式掩码组件在数据增强过程中的适应性和可学习性，我们通过最大化基于 InfoMax 的语义相关性，将自监督学习信号引入语义相关性。形式化地，我们有：

$$\mathcal{L}_{mask} = -\sum_{v \in \mathcal{V}} \gamma'(v) \tag{4}$$

我们定义了增强损失目标 $\mathrm{L}_{mask}$，以便在学习掩码范式中引入额外的自监督信号。优化后的 $\mathrm{L}_{mask}$ 的核心思想是根据.....的影响动态调整语义相关性的推导过程。

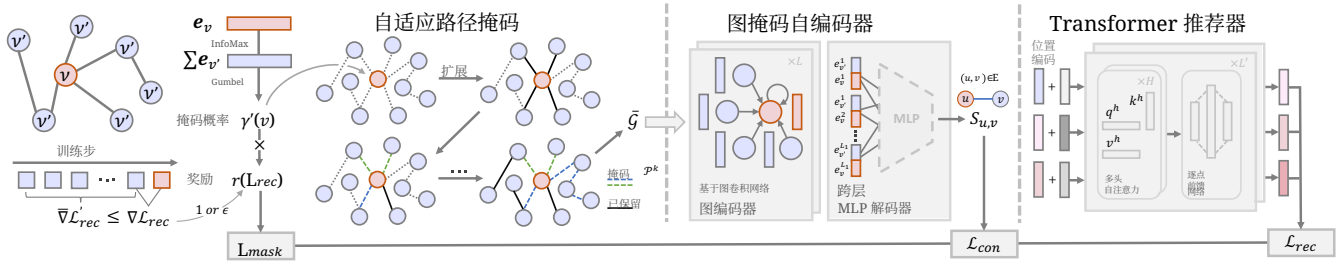



**图2：我们提出的 MAERec 的整体架构。i）通过自适应路径掩码，自动、动态地提炼用于自监督的物品转移模式。ii）使用轻量级图掩码自编码器重建被掩码的物品转移。iii）在主要推荐任务中，使用 Transformer 作为序列编码的骨干网络。**

在下游任务中进行掩码。这种自动化的学习掩码过程使其能够很好地适应不断变化的序列推荐场景。

*3.1.2* **转移路径掩码**。 使用学习掩码模块生成锚节点集合 $V_a$后，我们提出利用图路径掩码处理信息丰富的物品—物品转移模式，并将其作为基于自监督学习的重建信号。为此，我们的转移路径掩码机制应具备两个关键特性：（1）重建的物品转移关系不仅应包含序列内物品关联，还应涵盖序列间物品依赖关系；（2）应保留来自近期和过往交互的多样化序列模式，以体现物品的短期和长期依赖关系。

为实现上述目标，我们的转移路径掩码机制包含两个步骤：扩展 和采样。具体而言，我们基于递归随机游走过程确定需要掩码的路径，该过程包含转移图 $G = (V, E)$ 上一系列逐物品连接。给定关于表示一致性的 $k$ 阶语义相关性推导，我们将路径掩码机制正式定义如下：

$$\mathcal{P}^k = \begin{cases} \mathcal{V}_a, & \text{if } k = 1 \\ \mathcal{P}^{k-1} \cup \varphi(\mathcal{N}(\mathcal{P}^{k-1}), p^k) & \text{otherwise} \end{cases} \tag{5}$$

$P^k$ 表示包含的物品转移连接集合。这里，我们将 $N(P^{k-1})$ 定义为通过 E 中的边与 $P^{k-1}$ 中的节点直接连接的节点集合。为了引导随机游走过程，我们使用丢弃率 $0 < p < 1$，它用于对集合中的元素执行采样操作 $\varphi(\cdot, \cdot)$。在框架的扩展步骤中，我们从已识别的锚节点 $V_a$ 开始随机游走，通过函数 $\varphi(\cdot, \cdot)$递归添加相连的物品，以采样转移路径。在丢弃步骤中，我们选择连接被丢弃节点的物品转移关系，作为通过掩码自编码器进行重建的转移路径。由此生成的锚节点路径掩码包含长度各异的序列，最大长度为 $2k$。通过这种方式，我们的方法能够捕捉用户行为复杂且动态的特性。此外，我们的框架还使锚节点在序列中的位置更加多样化，从而注入短期和长期自监督信号。

*3.1.3* **任务自适应增强**。 为减轻与任务无关的信息和数据方差的影响，我们通过任务自适应增强改进了增强器，使其能够处理更加多样的用户行为序列。为此，我们引入了一个任务自适应函数，引导掩码学习范式，通过掩码更具信息量的物品转移路径进行自监督，从而提升模型的泛化能力。否则，重建与任务无关的信息会削弱自监督学习模型的表征能力。我们引入了一个任务自适应函数，记为 $r(\cdot)$:

$$r(\mathcal{L}_{rec}) = \begin{cases} 1 & \text{if } \nabla\mathcal{L}_{rec} > \overline{\nabla}\mathcal{L}'_{rec} \\ \epsilon & \text{otherwise} \end{cases} \tag{6}$$

$L_{rec}$ 表示下游任务的损失函数。 $\nabla L_{rec}$ 表示当前训练步与上一步之间的差异，而 $\overline{\nabla} L_{rec}$ 表示过去 $\delta > 1$ 步中 $L_{rec}$ 变化的平均值。数据特定掩码的贡献通过相应重建自监督学习损失带来的收益来衡量。此函数使可学习掩码范式能够根据当前掩码对下游任务的影响调整其掩码策略。我们使用 $\epsilon < 1$ 作为一个较小的常数。如果当前掩码对应的自监督重构任务使 $L_{rec}$下降得更快，则对应的掩码损失 $L_{mask}$较小。反之，掩码损失会更大，并引导学习掩码模块更容易适应。

给定估计得到的任务特定损失奖励 $r(L_{rec})$，我们通过任务自适应正则化 $L_{mask} = -r(L_{rec}) \sum_{v \in \mathcal{V}} \gamma_v(v)$，增强式 4 中基于信息最大化的掩码学习。整体的自适应路径掩码过程可表述如下：

$$\overline{\mathcal{G}} = \text{Path-Mask}(\mathcal{G}, \mathcal{V}_a) = (\mathcal{V}, \mathcal{E} \backslash \mathcal{P}^k) \tag{7}$$

其中，$\overline{G}$ 表示包含掩码转移路径的增强图。$V_a$ 表示自适应选择的锚节点集合，$P^k$ 表示由 $V_a$生成的路径中的边集合。

## 3.2 图掩码自编码器

通过自适应路径掩码得到图 G 后，我们将其输入图自编码器框架，以执行用于自监督的重构任务。为提高模型效率，我们采用轻量级的编码器-解码器框架。具体而言，我们使用简化的图卷积网络作为图嵌入编码器，并使用跨层多层感知机作为转移路径重构解码器。

*3.2.1* **基于图卷积网络的图编码器**。受 [4, 14]，提出的轻量级图卷积网络在效果和效率方面的启发，我们省略了在图 G 上进行消息传递时繁重的变换和激活操作，如下所示：

$$\mathbf{e}_v^{l+1} = \mathbf{e}_v^l + \sum_{v' \in \mathcal{N}_v} \mathbf{e}_{v'}^l; \quad \tilde{\mathbf{e}}_v = \sum_{l=1}^{L} \mathbf{e}_v^l \tag{8}$$

其中，$L$ 表示总层数，$\mathrm{N}_v$ 表示 $v$的一跳邻域。$\mathbf{e}_v^l, \mathbf{e}_{v'}^l \in \mathbb{R}^d$ 是第 $l$层中$v, v\prime \in \mathrm{V}$的嵌入。$d$ 表示潜在空间的嵌入维度。在最后一层中，我们添加残差连接以缓解梯度消失问题 [12]。具体而言，我们将所有隐藏层的表示相加，得到物品 $v$的最终嵌入 $\tilde{\mathbf{e}}_v$。通过这个基于 GCN 的图编码器，所提出的模型能够将掩码图的结构信息编码为物品嵌入，用于推荐以及掩码边的重构。

*3.2.2* **转移路径重建解码器**。对物品嵌入进行编码后，解码器负责利用增强后的 G 上的掩码路径，重建缺失的物品转移关系。为了解决 GCN 的过平滑问题，我们使用跨层多层感知机（MLP）作为解码器。对于一条被掩码的物品-物品边（$v$，$v\prime$），我们使用编码器每一层中的物品嵌入 $v$ 和 $v\prime$ 来构建对应于物品转移模式的边嵌入。

$$\mathbf{e}_{v,\ v'} = \Big\|_{i,\ j=1}^{L} \mathbf{e}_v^i \odot \mathbf{e}_{v'}^{\ j} \tag{9}$$

编码器图卷积网络中的层数用 $L$表示，拼接操作用 $\|$ 表示，逐元素乘法用 $\odot$表示。构建边嵌入后，将其输入 MLP，以预测目标物品转换路径的标签（真或假）。

*3.2.3* **基于自监督学习的重建目标**。所提出的模型不同于传统图自编码器，它通过恢复经由我们的方法学习得到的掩码全局物品转换路径来生成自监督信号。该自监督重构任务的辅助目标可表示为：

$$\mathcal{L}_{con} = -\sum_{(v,v') \in \mathcal{E} \setminus \mathcal{P}^k} \log \frac{\exp(s_{v,v'})}{\sum_{v'' \in \mathcal{V}} \exp(s_{v,v''})} \tag{10}$$

对应于 $\mathbf{e}_{v,\ v'}$ 和 $\mathbf{e}_{v,\ v''}$ 的 MLP 输出分别用 $s_{v,\ v'}$ 和 $s_{v,\ v''}$表示。它们表示 $(v,\ v\prime)$ 是掩码物品-物品转换的概率。为加快优化过程，采用负采样来采样负边 $(v,\ v\prime\prime)$。

## 3.3 Transformer序列编码器 er

SASRec [19] 和 Bert4Rec [27], 等模型证明了 Transformer 在推荐系统中学习序列行为模式的有效性，这启发我们在主要的有监督序列推荐任务中使用 Transformer。

*3.3.1* **嵌入层**。为每个物品 $v \in \mathrm{V}$分配一个可学习的嵌入 $\mathbf{e}_v$，用于在训练过程中生成转移路径掩码以及进行图编码。在嵌入在我们的 Transformer 层中，将可学习的位置嵌入 $\mathbf{p}_i \in \mathbb{R}^d$添加到输入序列中位置 $i$处物品的初始嵌入上。随后，用户 $u$ 交互过的长度为 $l$的物品序列 $S^u$ 的初始序列嵌入按如下方式获得：

$$\mathbf{E}_u = [(\tilde{\mathbf{e}}_{s_1^u} + \mathbf{p}_1), (\tilde{\mathbf{e}}_{s_2^u} + \mathbf{p}_2), \cdots (\tilde{\mathbf{e}}_{s_l^u} + \mathbf{p}_l)] \tag{11}$$

生成的物品表示 $\tilde{\mathbf{e}}^u_{s_l}$通过聚合不同图层的信息获得，如公式 8 所示。

*3.3.2* **多头自注意力块**。Transformer架构的核心组件是多头自注意力机制，其可表示如下：

$$\hat{\mathbf{E}}_u^l = \Big\|_{h=1}^{H} \mathbf{A}^h \mathbf{E}_u^l \mathbf{W}_{\mathbf{V}}^{\mathbf{h}}; \quad \mathbf{A}^h = \frac{(\mathbf{E}_u^l \mathbf{W}_Q^h)(\mathbf{E}_u^l \mathbf{W}_K^h)^T}{\sqrt{d/H};} \tag{12}$$

$$\mathbf{E}_u^{l+1} = \mathrm{ReLU}(\hat{\mathbf{E}}_u^l \mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2; \quad \tilde{\mathbf{E}}_u = \sum_{l=1}^{L'} \mathbf{E}_u^l \tag{13}$$

其中，$L\prime$ 和 $H$分别表示多头自注意力块的总数和注意力头的数量。$\mathbf{A}^h$ 表示第 $h$ 个注意力头的注意力值。$\mathbf{W}_Q^h, \mathbf{W}_K^h, \mathbf{W}_V^h \in \mathbb{R}^{d \times (d/H)}$ 是注意力机制中对应查询、键和值的第 $h$ 个注意力头的投影矩阵。参数 $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d \times d}$ 和$\mathbf{b}_1, \mathbf{b}_2 \in \mathbb{R}^d$ 构成逐点前馈网络。通过应用残差连接，得到最终的序列嵌入 $\tilde{\mathbf{E}}u$。值得注意的是，对注意力值以及每个块的输入应用层归一化和 Dropout，以提升模型性能。

## 3.4 多任务学习

为了预测用户序列与物品$v$交互的概率，我们将序列最后一个多头自注意力模块的输出作为序列的最终嵌入，并计算其与目标物品嵌入$\mathbf{e}v$的点积。我们使用交叉熵损失函数来计算主监督推荐任务的优化目标。与 [19, 40], 类似，我们将每个用户序列 $S^u$中长度为 $l_u$的所有子序列作为训练数据，即*i.e.*，$(s_1^u)$，$(s_1^u, s_2^u)$，$\cdots$，$(s_1^u, \cdots\ s_{l_u-1}^u)$。因此，推荐任务的损失函数 $\mathrm{L}_{rec}$ 表示为：

$$\mathcal{L}_{rec} = -\sum_{u \in \mathcal{U}} \sum_{1 \le t \le l_u} \log \sigma(\tilde{\mathbf{E}}_{u,t} \cdot \tilde{\mathbf{e}}_{s_u^{t+1}}) + \log(1 - \sigma(\tilde{\mathbf{E}}_{u,t} \cdot \tilde{\mathbf{e}}_{v_t^-})) \tag{14}$$

$\tilde{\mathbf{E}}u,\ t$表示$(s_1^u,\ \cdots\ s_t^u)$的序列嵌入，$v_t^- \notin S_u$是针对第 $t$个交互物品随机采样的负物品。

为了通过自监督学习增强来增强我们的推荐器，我们采用多任务学习训练模型，将图掩码自编码器生成的自监督学习信号融入其中。

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{mask} + \mathcal{L}_{con} + \lambda \|\Theta\|_{\mathrm{F}}^2 \tag{15}$$

$\mathrm{L}_{mask}$ 表示基于 InfoMax 的 SSL 适配损失，而 $\mathrm{L}_{con}$ 表示掩码物品转移重建的自监督学习损失。$\Theta$表示模型参数，$\lambda$ 是权重衰减系数。

## 3.5 深入模型讨论

为了探究所提出的模型如何通过自适应转移路径掩蔽来避免引入噪声数据，我们针对特定物品的掩码损失进行了理论分析。假设

**表 1：数据集统计信息**

| 数据集 | #用户 | #物品 | #inter. | 平均长度 | 密度 |
|---|---|---|---|---|---|
| 图书 | 93,403 | 54,756 | 506637 | 5.45 | $9.94 \times 10^{-5}$ |
| Toys | 116,429 | 54,784 | 478460 | 4.11 | $7.50 \times 10^{-5}$ |
| Retailrocket | 91,655 | 43,886 | 452546 | 4.94 | $1.12 \times 10^{-4}$ |

物品 $v \in$ V是一个噪声物品，其相关重建可能会对使用所提供的自监督信号进行表示学习产生负面影响。在这种情况下，其对应的掩码损失相对于其嵌入$\mathbf{e}_v$的梯度为：

$$\frac{\partial \mathcal{L}_{mask}(v)}{\partial \mathbf{e}_v} = -\frac{r(\mathcal{L}_{rec})}{|\mathcal{N}_v^k|} \sum_{v' \in \mathcal{N}_v^k} \left( \frac{\mathbf{e}_{v'}}{\|\mathbf{e}_v\| \|\mathbf{e}_{v'}\|} - \frac{(\mathbf{e}_v^T \mathbf{e}_{v'}) \mathbf{e}_v}{\|\mathbf{e}_v\|^3 \|\mathbf{e}_{v'}\|} \right).$$

我们进一步计算梯度的范数如下：

$$\left\| \frac{\partial \mathcal{L}_{mask}(v)}{\partial \mathbf{e}_v} \right\| = -\frac{r(\mathcal{L}_{rec})}{|\mathcal{N}_v^k| \|\mathbf{e}_v\|} \sum_{v' \in \mathcal{N}_v^k} (1 - \bar{\mathbf{e}}_v^T \bar{\mathbf{e}}_{v'}) \tag{16}$$

在上述公式中，$\bar{\mathbf{e}}_v = \mathbf{e}_v / \|\mathbf{e}_v\|$和 $\bar{\mathbf{e}}_{v'} = \mathbf{e}_{v'} / \|\mathbf{e}_{v'}\|$，而项 $\bar{\mathbf{e}}_v^T \bar{\mathbf{e}}_{v'}$ 表示两个物品之间的余弦相似度。因此，随着 $v$ 与其邻居节点之间的相似度分数降低，总和也会降低，而范数则会增大。这突出了结构一致性较低的物品的梯度，从而通过 SSL 适应改进转移路径掩码范式。因此，与采用手工设计和随机增强的现有对比学习方法相比，我们的模型能够减轻行为噪声的影响。

# 4 评估

## 4.1 实验设置

*4.1.1* **数据集**。 我们在推荐系统领域广泛使用的真实世界数据集上开展实验：Amazon Books、Amazon Toys 和 Retailrocket。前两个数据集来自亚马逊平台，我们使用用户与图书和玩具类别中物品的交互数据版本。另一个基准数据集 Retailrocket 则采集自一个电子商务网站。我们遵循 [35, 40] 中类似的数据预处理步骤，为每位用户生成物品序列，并在表 1 中汇总数据统计信息。

*4.1.2* **评估协议**。 我们采用序列推荐评估中常用的留一策略 [24, 27]。具体而言，对于每个用户序列，我们将最后一次交互的物品作为测试物品。我们采用两种评估指标：命中率（*HR@K*）和归一化折损累计增益（*NDCG@K*） [33, 36]，其中（K=5，10，20），用于评估所有算法的性能。

*4.1.3* **对比算法**。 我们将 MAERec 与 13 种具有竞争力的序列推荐基线方法进行比较，这些方法涵盖了多种技术，可分为四类：（1）基于RNN的方法（即，GRU4Rec [15], NARM [19]）。（2）基于Transformer的方法（即，SASRec [19], BERT4Rec [27]）。（3）基于图神经网络的序列推荐器（即，SRGNN [38], GCE-GNN [35], HyRec [33], 和 SURGE [2]）。（4）自监督学习模型（即，ContraRec [32], CL4SRec [40],CoSeRec [24], DuoRec [26], 和 ICLRec [7]）。基线方法详述如下：

- **GRU4Rec** [15]：该方法使用门控循环单元对用户序列进行编码，并采用基于排序的损失训练模型。
- **NARM** [21]：该模型通过注意力机制改进基于门控循环单元的序列推荐模型，并将其用作混合序列编码器。
- **SASRec** [19]。这是一个开创性模型，使用单向 Transformer 对物品上的用户序列模式进行编码。
- **BERT4Rec** [27]：该模型以双向 Transformer 作为序列编码的骨干网络。它将自注意力与前馈网络结合，用于物品转移建模。

- **SRGNN** [38]：该模型使用门控图神经网络，在转移图上学习物品嵌入，并利用自注意力机制生成序列嵌入以进行预测。

- **GCE-GNN** [35]：该模型包含两种物品嵌入。全局级嵌入捕捉不同序列中物品间的整体关系。局部级物品嵌入则由图卷积网络编码。

- **HyRec** [33]：该模型利用超图卷积网络学习物品间的高阶相关性。它将每个用户视为一条超边，以连接多个交互物品。

- **SURGE** [2]：这是一种最先进的模型，通过聚类并基于度量学习构建物品-物品兴趣图，显式地学习用户兴趣。随后，它使用带有图池化层的图神经网络生成用于预测的嵌入。

- **ContraRec** [32]：该模型采用 Transformer 作为骨干网络，对物品转移进行序列编码。针对数据增强，它提出了一种整体联合学习范式，以利用多种对比自监督信号进行学习。

- **CL4SRec** [40]：这是一种对比方法，通过对序列进行随机扰动，使用物品裁剪、掩码和重新排序来生成对比视图。随后，该方法借助随机增强器，最大化生成的对比视图之间的表示一致性。

- **CoSeRec** [32]：除了 CL4SRec 提出的三种对比视图外，该模型还引入了另外两种利用物品相关性的增强算子，即*i.e.* 替换和插入，用于生成对比学习视图。

- **DuoRec** [26]：该模型引入了带有模型级增强的对比正则化，以改善物品嵌入分布并解决表征退化问题。

- **ICLRec** [7]：该模型首先通过聚类学习潜变量，以表示用户意图。随后，它最大化序列与其对应意图之间的一致性。

*4.1.4* **超参数设置**。 我们使用 PyTorch 实现所提出的 MAERec，并使用 Adam 优化器进行参数推断，学习率为 $1e^{-3}$，批量大小为 256。在图自编码器原理中，我们将 GNN 层数设置为 {1,2,3}。Transformer 层数设为 2，注意力头数设为 4，以获得多维表示。在自适应转移路径掩码模块中，我们每 10 个轮次对 V$_a$进行采样，以提高效率。每个掩码步骤的锚节点数量 $\alpha$从 {50, 100,200, 400} 中进行搜索。参数$k$从 {3,5,7} 中搜索，以确定随机游走路径的最大长度 $2k$。模型训练过程中，权重衰减因子从 $\{1e^{-3}, 1e^{-5}, 1e^{-7}\}$ 中选取。

**表2：不同方法在 Amazon-books、Amazon-toys 和 Retailrocket 数据集上的性能比较。**

| 数据集 | 指标 | GRU4Rec | NARM | SASRec | ContraRec | BERT4Rec | SRGNN | GCE-GNN | HyRec | SURGE | CoSeRec | DuoRec | CL4SRec | ICLRec | MAERec | p值 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 图书 | HR@5 | 0.3523 | 0.4064 | 0.3544 | 0.4312 | 0.5385 | 0.5472 | 0.4087 | 0.4550 | 0.4861 | 0.5684 | 0.5941 | 0.5089 | 0.5208 | **0.6472** | $7e^{-9}$ |
| | NDCG@5 | 0.2517 | 0.2955 | 0.2766 | 0.3366 | 0.4404 | 0.4404 | 0.3178 | 0.3454 | 0.3637 | 0.4540 | 0.4894 | 0.3969 | 0.4146 | **0.5201** | $1e^{-7}$ |
| | HR@10 | 0.4817 | 0.5340 | 0.4393 | 0.5304 | 0.6238 | 0.6458 | 0.5033 | 0.5619 | 0.6193 | 0.6666 | 0.6830 | 0.6144 | 0.6172 | **0.7533** | $1e^{-8}$ |
| | NDCG@10 | 0.2940 | 0.3396 | 0.3058 | 0.3685 | 0.4681 | 0.4723 | 0.3484 | 0.3800 | 0.4068 | 0.4858 | 0.5182 | 0.4310 | 0.4458 | **0.5545** | $1e^{-7}$ |
| | HR@20 | 0.6268 | 0.6725 | 0.5407 | 0.6440 | 0.7121 | 0.7503 | 0.6090 | 0.6778 | 0.7604 | 0.7622 | 0.7706 | 0.7216 | 0.7141 | **0.8543** | $2e^{-7}$ |
| | NDCG@20 | 0.3306 | 0.3737 | 0.3298 | 0.3966 | 0.4904 | 0.4987 | 0.3751 | 0.4092 | 0.4424 | 0.5100 | 0.5403 | 0.4581 | 0.4703 | **0.5801** | $2e^{-7}$ |
| Toys | HR@5 | 0.2247 | 0.2047 | 0.2454 | 0.2989 | 0.3717 | 0.3610 | 0.2597 | 0.2978 | 0.3075 | 0.3869 | 0.4412 | 0.3451 | 0.3483 | **0.4864** | $4e^{-6}$ |
| | NDCG@5 | 0.1543 | 0.1383 | 0.1854 | 0.2318 | 0.2885 | 0.2720 | 0.1880 | 0.2149 | 0.2231 | 0.2939 | 0.3450 | 0.2696 | 0.2655 | **0.3715** | $3e^{-3}$ |
| | HR@10 | 0.3262 | 0.3029 | 0.3299 | 0.3805 | 0.4665 | 0.4674 | 0.3628 | 0.4030 | 0.4210 | 0.4925 | 0.5424 | 0.4417 | 0.4416 | **0.6033** | $7e^{-4}$ |
| | NDCG@10 | 0.1888 | 0.1718 | 0.2155 | 0.2579 | 0.3191 | 0.3064 | 0.2211 | 0.2488 | 0.2596 | 0.3280 | 0.3777 | 0.3007 | 0.2971 | **0.4093** | $1e^{-3}$ |
| | HR@20 | 0.4700 | 0.5012 | 0.4333 | 0.4862 | 0.5780 | 0.5940 | 0.4952 | 0.5251 | 0.5752 | 0.6135 | 0.6583 | 0.5638 | 0.5652 | **0.7343** | $8e^{-2}$ |
| | NDCG@20 | 0.2240 | 0.2470 | 0.2413 | 0.2843 | 0.3472 | 0.3383 | 0.2545 | 0.2797 | 0.2984 | 0.3585 | 0.4070 | 0.3315 | 0.3271 | **0.4424** | $3e^{-2}$ |
| 零售 | HR@5 | 0.4164 | 0.5586 | 0.6136 | 0.6587 | 0.7596 | 0.6973 | 0.4238 | 0.4801 | 0.7091 | 0.8063 | 0.7873 | 0.6663 | 0.7818 | **0.8577** | $2e^{-10}$ |
| | NDCG@5 | 0.2950 | 0.4238 | 0.5077 | 0.5553 | 0.6882 | 0.5974 | 0.3371 | 0.3792 | 0.5902 | 0.7117 | 0.7128 | 0.5752 | 0.6904 | **0.7660** | $4e^{-8}$ |
| | HR@10 | 0.4518 | 0.6757 | 0.6968 | 0.7387 | 0.8084 | 0.7719 | 0.5182 | 0.5722 | 0.7857 | 0.8530 | 0.8291 | 0.7336 | 0.8282 | **0.8959** | $3e^{-8}$ |
| | NDCG@10 | 0.3456 | 0.4600 | 0.5389 | 0.5823 | 0.7039 | 0.6216 | 0.3676 | 0.4091 | 0.6151 | 0.7269 | 0.7264 | 0.5969 | 0.7055 | **0.7784** | $1e^{-7}$ |
| | HR@20 | 0.6095 | 0.7879 | 0.7687 | 0.8140 | 0.8577 | 0.8425 | 0.6275 | 0.6361 | 0.8581 | 0.8951 | 0.8703 | 0.8018 | 0.8733 | **0.9218** | $1e^{-4}$ |
| | NDCG@20 | 0.3830 | 0.4972 | 0.5558 | 0.6009 | 0.7164 | 0.6394 | 0.3951 | 0.4256 | 0.6334 | 0.7376 | 0.7368 | 0.6142 | 0.7169 | **0.7850** | $1e^{-6}$ |

## 4.2 性能比较

表2展示了所有方法在三个基准数据集上的表现，评估指标为 *HR@K* 和 *NDCG@K*，设置为 top-5、top-10 和 top-20。此外，我们进行了显著性检验，以证明 MAERec 优于最强基线方法，其中 a $p$-值 $< 0.05$ 表示提升具有统计显著性。可以得出以下观察结果：

• **观察1：优于最先进的基于图神经网络的模型**。MAERec在不同设置下始终优于所有基于图神经网络的基线方法。现有的基于图神经网络的模型在训练标签不足的短序列上可能会出现过拟合问题。此外，经自监督学习增强的序列推荐模型表现优于基于图神经网络的基线方法，这表明引入自监督信号，在标签有限的情况下增强序列编码是有益的。为缓解数据稀缺问题，我们的MAERec采用图掩码自编码器生成式自监督信号，以实现有效的数据增强。因此，模型能够学习到信息丰富且稳健的表示，从而保留全局物品转移模式。

• **观察2：优于最先进的自监督学习方法**。与所有采用自监督增强的基线方法相比，MAERec 的整体表现最佳。虽然这些方法通过引入自监督信号，通常优于基于图神经网络的模型，但其随机数据增强仍会导致次优表现。这是因为常见自监督学习模型中使用的随机增强策略（例如物品裁剪或重新排序）作用于所有用户序列和所有物品，使模型容易受到序列和物品层面噪声扰动的影响，例如交互噪声和流行度偏差。MAERec 不采用随机增强，而是通过掩码处理由自适应选取的物品生成的信息丰富的转移路径，以一种可学习的策略引入有用的自监督信号。因此，作为一种稳健的数据增强器，新设计的 MAERec 凭借自适应图掩码自编码器，大幅超越了现有的自监督学习基线方法。

• **观察3：基于掩码的数据增强的优势**。一些基线模型也利用掩码操作进行自监督学习，例如 CL4SRec 和 CoSeRec，它们通过对序列中的物品进行掩码来生成对比视图。此外，BERT4Rec 将完形填空目标引入序列编码器，以获得物品嵌入。得益于这种基于掩码的数据增强，BERT4Rec 的表现优于仅使用单向 Transformer 进行序列编码的 SASRec。同时，CoSeRec 和 CL4SRec 也因其数据增强而优于许多其他基线方法。然而，MAERec 是唯一能够针对物品转移模式自动执行掩码与重构范式，并实现有效任务适配的模型，因此取得了更佳表现。

## 4.3 消融研究

本节介绍针对所提出的MAERec算法三个变体开展的消融研究：-**L2M**、-**PA**和-**TA**。该研究旨在考察不同组件对算法整体性能的贡献。研究结果见表3。

*4.3.1* **掩码生成器的影响**。第一个变体 **-L2M** 使用相同比例的随机掩码替代我们设计的掩码生成器。结果表明，非自适应掩码可能会损害重要的转移关系，或在基于掩码的重建任务中引入噪声信息，从而削弱表示并导致性能不佳。

*4.3.2* **转移路径掩码的效果**。 第二种变体-**PA**研究了掩码转移路径而非单个节点的优势。在此变体中，掩码路径的最大长度设为 $k = 1$，这使路径掩码等同于单节点掩码。结果表明，所提出的转移路径掩码显著提升了整体性能，突显了从序列内和序列间两个视角学习掩码物品转移模式的有效性。

**表3：模型消融研究的不同变体。**

| 方法 | Amazon Books | | Retailrocket | |
|---|---|---|---|---|
| | HR@10 | NDCG@10 | HR@10 | NDCG@10 |
| -L2M | 0.7482 | 0.5510 | 0.8800 | 0.7547 |
| -PA | 0.7272 | 0.5239 | 0.8590 | 0.7056 |
| -TA | 0.7507 | 0.5542 | 0.8801 | 0.7564 |
| MAERec | **0.7723** | **0.5720** | **0.8921** | **0.7659** |

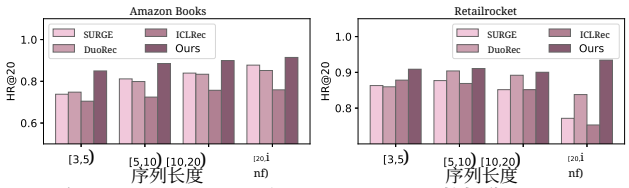



**图 3：在 Amazon Books 和 Retailrocket 数据集上，HR@20 随序列长度的变化。**

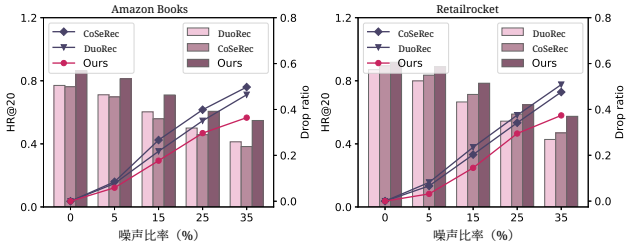



**图4：Amazon Books 和 Retailrocket 上的性能与噪声比率的关系。柱状图显示 HR@20 方面的性能，折线显示性能下降比率。**

*4.3.3* **任务自适应增强的影响。** 带有任务自适应函数 $r(\cdot)$ 的自适应掩码损失也是 MAERec算法的关键组成部分。为了研究其影响，我们在变体 **-TA** 中移除了可学习掩码损失 $L_{mask}$ (公式 4)，从而禁用了任务自适应正则化。结果显示，性能显著下降，这表明任务自适应函数会根据目标推荐任务产生的梯度，引导模型朝着更好的方向训练。借助任务自适应约束，我们的 MAERec 算法能够通过选择性地应用自监督增强，有效防止数据增强器学习与任务无关的信息。

## 4.4 对噪声和稀疏性的优势

*4.4.1* **数据稀疏性下的性能。** 对于包含大量短序列的序列推荐系统而言，数据稀疏性是一个常见问题。为评估所提出的 MAERec算法应对这一问题的鲁棒性，我们根据用户序列中物品序列的长度，将用户序列划分为四组：$[3,5)$, $[5,10)$, $[10,20)$ $[20,\infty)$。图 3 展示了我们的模型和基线方法在 Amazon Books 和 Retailrocket 数据集上的表现。结果表明，在所有稀疏性水平下，我们提出的模型均持续优于基线方法。具体而言，我们的模型在短序列上的表现良好，这有助于解决实际推荐场景中的标签不足问题。这得益于自监督学习增强的转移路径掩码模块，

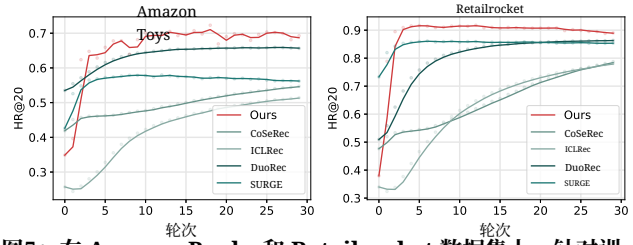



**图5：在 Amazon Books 和 Retailrocket 数据集上，针对训练轮数的测试性能，用于收敛分析。**

该模块以自动且自适应的方式保留了局部和全局物品依赖关系。DuoRec 和 ICLRec 中使用的手工设计的数据增强可能会限制其表示泛化性，导致它们难以根据短序列准确编码用户偏好，这是由交互数据的多样性所致。相比之下，我们的 MAERec算法能够有效地将长物品序列中的知识迁移至短序列建模中，从而使模型能够从标签有限的稀疏数据中学习有意义的表示。

*4.4.2* **数据噪声下的性能表现。** 现实世界中的推荐系统经常面临含噪声的用户交互。为了评估我们提出的MAERec算法对噪声数据的鲁棒性，我们将一定比例（*i.e.*，5%、15%、25%和35%）的用户交互替换为随机采样的负向物品，并在具有人工噪声扰动的数据集上评估模型。如图4所示，在Amazon Books和Retailrocket两个数据集的所有噪声水平下，我们提出的模型均优于基线方法。此外，与基线方法相比，我们的模型始终表现出更低的性能下降比率。

我们将这一优势归因于算法中的自适应转移路径掩码模块。首先，自适应掩码策略利用基于 InfoMax 的掩码概率和掩码损失，避免掩码与噪声节点相连的边，从而促使模型对具有可靠依赖关系的节点进行掩码并学习。这显著增强了模型抵御噪声扰动的鲁棒性。其次，算法中的转移路径是在锚节点的局部子图中生成的，因此，与随机生成的增强序列相比（例如通过随机掩码、随机丢弃和重新排序生成的序列），此类序列中的物品具有更高的语义相关性。因此，从图4可以看出，CoSeRec 对数据噪声的抵抗能力相对不稳定，这表明随机增强方法不如自适应路径掩码。

## 4.5 模型可扩展性研究

我们从收敛速度和训练时间开销两个方面评估所提出模型的效率。所有实验均在单块 NVIDIA GeForce RTX 3090 GPU 上进行。

*4.5.1* **收敛分析。** 为了研究我们提出模型的收敛效率，我们在 Amazon Toys 和 Retailrocket 数据集上，将其训练过程与多种基线方法进行比较，包括基于GCN的方法（SURGE）和基于SSL的方法（DuoRec、CoSeRec、ICLRec）。这些模型的训练过程如图5所示。从结果中可以看出，我们提出的模型以最少的训练轮次取得了最佳性能。它在大约10轮次时达到了报告的最佳性能。

**表4：Amazon Toys 和 Retailrocket 上的训练时间开销。**

| Data | SURGE | ICLRec | CoSeRec | DuoRec | Ours |
|---|---|---|---|---|---|
| 亚马逊零售 | 7.39 小时<br>2.67小时 | 5.28 小时<br>4.3h | 21.01 小时<br>16.18小时 | 3.89 小时<br>4.45小时 | 1.12 小时<br>1.82小时 |

在 Amazon Toys 上为 5 轮左右，在 Retailrocket 上为 5 轮左右，而 CoSeRec 需要 200 多轮才能收敛。我们认为，这种优势源于图掩码自编码器增强的自监督学习任务，它为模型注入了充足的监督信号，使其能够高效地从掩码物品转移图中学习有信息量的物品转移模式。

*4.5.2* **计算成本评估。** 为研究所提出模型的训练计算成本，我们还记录了模型和基线方法的训练时间，如表4所示。结果表明，与所比较的方法相比，我们提出的模型在效率上有显著提升，这说明我们的模型是一种轻量且强大的推荐器，能够实现自动化且任务自适应的掩码自编码。我们的模型借助自适应SSL增强，因而与其他基于SSL的方法相比，所需训练时间要少得多；后者针对下游推荐任务使用非自适应（基于启发式）的增强算子。

## 4.6 超参数研究

我们通过实验研究了若干关键超参数对所提出模型性能的影响。实验结果如图 6 所示，展示了模型在 Amazon Books 和 Amazon Toys 数据集上的 HR@20 和 NDCG@20 相对降幅。观察结果如下：

• **随机游走比率 $p$ 物品转移。**该超参数控制由已识别锚节点触发的掩码边数量，并影响掩码路径的长度，而掩码路径反映了物品转移模式。具体而言，$p$ 的值越大，表示掩码物品转移路径越长。实验表明，$p = 0.5$ 时模型性能最佳。这说明，对长度可变的转移路径进行掩码，有助于模型从更多样化的自监督学习信号中进行学习。

• **掩码转移路径尺度 $k$。**该超参数控制生成的转移路径的最大长度 $2k$。实验表明，将 $k = 1$ 设为（即，仅对锚节点 1 跳范围内的边进行掩码）会导致性能显著下降。有趣的是，我们还观察到，掩码过长

路径可能导致性能下降，因为提供的信息不足以获得准确的重构结果。这可能会削弱辅助自监督学习任务对目标推荐任务的增强效果。

• **锚点节点数量 $\alpha$。** $\alpha$ 决定了转移路径掩码所用种子节点的数量。实验表明，对过少锚节点的转移路径进行掩码会导致性能变差。另一方面，选择过多锚节点进行路径掩码可能会破坏转移图的结构，从而对模型性能产生负面影响。

## 4.7 案例研究

我们在 Retailrocket 数据集上开展了一项案例研究，以探究所提出模型的潜在解释能力。具体而言，我们分别选取了两个语义相关性高和低的物品，

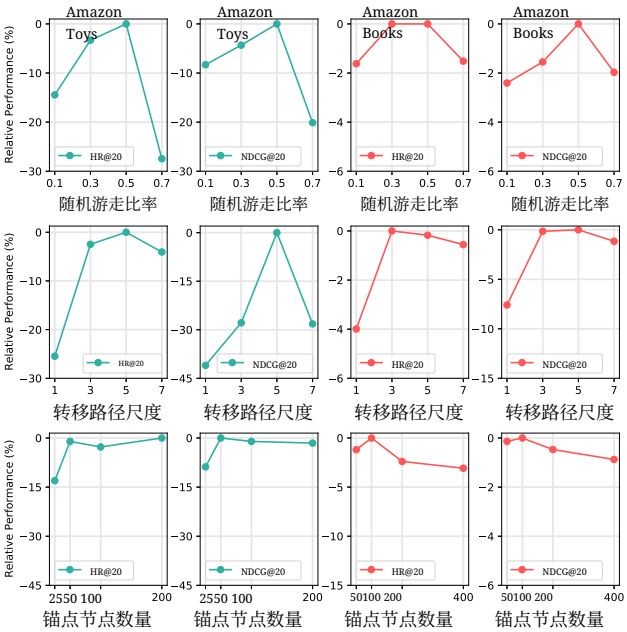



**图 6：MAERec 的超参数研究。**

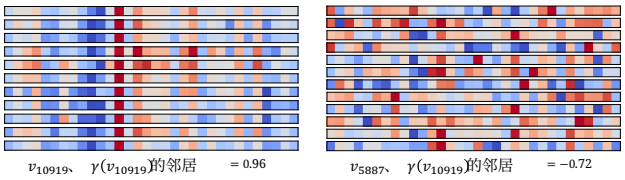



**图 7：学习到的语义相关性反映了转移图中相连物品之间的表示一致性。**

并在图 7 中可视化了它们在物品转移图上的邻居物品嵌入。结果表明，我们的学习掩码模块能够发现与锚节点具有可靠依赖关系的物品，这有助于重构任务。相比之下，语义相关性较低的物品（例如，误点击的物品）在嵌入空间中的邻域缺乏一致性。对这类物品的转移路径进行掩码可能会导致噪声数据增强，并误导序列模式编码的自监督过程。

## 5 结论

本文提出了一种自适应数据增强方法，通过全新的图掩码自编码器增强序列推荐系统。我们提出的 MAERec 借助所设计的自适应路径掩码，利用强大的自监督信号，实现了可控且任务自适应的增强。我们在三个真实世界数据集上进行了广泛实验，结果表明，与最先进的基线方法相比，我们的 MAERec 具有更优越的性能。未来，我们计划将该方法推广到分布外序列，以提升自监督学习训练的稳定性。这将有助于解决序列推荐中训练数据与测试数据之间的数据分布偏移问题，并使模型能够泛化到未来新到达的物品序列。

# 参考文献

[1] Xuheng Cai、Chao Huang、Lianghao Xia 和 Xubin Ren。2023。LightGCL：用于推荐的简单而有效的图对比学习。载于国际学习表征会议（ICLR）。[2] Jianxin Chang、Chen Gao、Yu Zheng、Yiqun Hui、Yanan Niu、Yang Song、Depeng Jin 和 Yong Li。2021。基于图神经网络的序列推荐。载于国际信息检索研究与发展会议（SIGIR）。378–387。[3] Jiawei Chen、Hande Dong、Xiang Wang、Fuli Feng、Meng Wang 和 Xiangnan He。2020。推荐系统中的偏差与去偏：综述及未来方向。信息系统汇刊（TOIS）（2020）。[4] 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人工智能会议（AAAI），第34卷。27–34。[5] Tianwen Chen 和 Raymond Chi-Wing Wong。2020。处理图神经网络在基于会话的推荐中的信息丢失问题。载于国际知识发现与数据挖掘会议（KDD）。1172–1180。[6] Yabo Chen、Yuchen Liu、Dongsheng Jiang、Xiaopeng Zhang、Wenrui Dai、Hongkai Xiong 和 Qi Tian。2022。Sdae：自蒸馏掩码自编码器。载于欧洲计算机视觉会议（ECCV）。Springer，108–124。[7] Yongjun Chen、Zhiwei Liu、Jia Li、Julian McAuley 和 Caiming Xiong。2022。用于序列推荐的意图对比学习。载于ACM网络会议（WWW）。2172–2182。[8] Jacob Devlin、Ming-Wei Chang、Kenton Lee 和 Kristina Toutanova。2018。Bert：用于语言理解的深度双向 Transformer 预训练。载于北美ACL分会会议（NAACL）。[9] Xinyan Fan、Zheng Liu、Jianxun Lian、Wayne Xin Zhao、Xing Xie 和 Ji-Rong Wen。2021。更轻、更优：用于下一项目推荐的低秩分解自注意力网络。载于国际信息检索研究与发展会议（SIGIR）。1733–1737。[10] Ziwei Fan、Zhiwei Liu、Yu Wang、Alice Wang、Zahra Nazari、Lei Zheng、Hao Peng 和 Philip S Yu。2022。通过随机自注意力进行序列推荐。载于ACM网络会议（WWW）。2036–2047。[11] Kaiming He、Xinlei Chen、Saining Xie、Yanghao Li、Piotr Dollár 和 Ross Girshick。2022。掩码自编码器是可扩展的视觉学习器。载于IEEE/CVF计算机视觉与模式识别会议（CVPR）。16000–16009。[12] 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IEEE/CVF计算机视觉与模式识别会议（CVPR）。770–778。[13] Sunan He、Taian Guo、Tao Dai、Ruizhi Qiao、Chen Wu、Xiujun Shu 和 Bo Ren。2022。VLMAE：视觉-语言掩码自编码器。arXiv预印本 arXiv:2208.09374（2022）。[14] Xiangnan He、Kuan Deng、Xiang Wang、Yan Li、Yongdong Zhang 和 Meng Wang。2020。Lightgcn：简化并增强用于推荐的图卷积网络。载于国际信息检索研究与发展会议（SIGIR）。639–648。[15] Balázs Hidasi、Alexandros Karatzoglou、Linas Baltrunas 和 Domonkos Tikk。2016。基于循环神经网络的会话推荐。载于国际学习表征会议（ICLR）。[16] Zhenyu Hou、Xiao Liu、Yuxiao Dong、Chunjie Wang、Jie Tang 等。2022。Graph-MAE：自监督掩码图自编码器。载于国际知识发现与数据挖掘会议（KDD）。[17] Eric Jang、Shixiang Gu 和 Ben Poole。2017。使用 Gumble-Softmax 的类别重参数化。载于国际学习表征会议（ICLR）。[18] Baoyu Jing、Chanyoung Park 和 Hanghang Tong。2021。Hdmi：高阶深度多路复用 InfoMax。载于万维网会议（WWW）。2414–2424。[19] Wang-Cheng Kang 和 Julian McAuley。2018。自注意力序列推荐。载于国际数据挖掘会议（ICDM）。IEEE，197–206。[20] Yejin Kim、Kwangseob Kim、Chanyoung Park 和 Hwanjo Yu。2019。长尾场景下的序列与多样化推荐。。载于国际人工智能联合会议（IJCAI），第19卷。2740–2746。[21] Jing Li、Pengjie Ren、Zhumin Chen、Zhaochun Ren、Tao Lian 和 Jun Ma。2017。神经注意力会话推荐。载于2017年ACM信息与知识管理会议论文集。1419–1428。[22] Guanyu Lin、Chen Gao、Yinfeng Li、Yu Zheng、Zhiheng Li、Depeng Jin 和 Yong Li。2022。用于序列推荐的双重对比网络。载于国际信息检索研究与发展会议（SIGIR）。2686–2691。[23] Siyi Liu 和 Yujia Zheng。2020。长尾会话推荐。载于ACM推荐系统会议（Recsys）。509–514。[24] Zhiwei Liu、Yongjun Chen、Jia Li、Philip S Yu、Julian McAuley 和 Caiming Xiong。2021。通过鲁棒增强进行对比式自监督序列推荐。arXiv预印本 arXiv:2108.06479（2021）。[25] Chen Ma、Liheng Ma、Yingxue Zhang、Jianing Sun、Xue Liu 和 Mark Coates。2020。用于序列推荐的记忆增强图神经网络。载于AAAI人工智能会议（AAAI），第34卷。5045–5052。[26] Ruihong Qiu、Zi Huang、Hongzhi Yin 和 Zijian Wang。2022。用于解决序列推荐中表示退化问题的对比学习。载于国际网络搜索与数据挖掘会议（WSDM）。813–823。[27] Fei Sun、Jun Liu、Jian Wu、Changhua Pei、Xiao Lin、Wenwu Ou 和 Peng Jiang。2019。BERT4Rec：基于 Transformer 双向编码器表示的序列推荐。载于国际信息与知识管理会议（CIKM）。1441–1450。[28] Qiaoyu Tan、Ninghao Liu、Xiao Huang、Rui Chen、Soo-Hyun Choi 和 Xia Hu。2023。S2GAE：自监督图自编码器是具有图掩码能力的通用学习器。载于国际网络搜索与数据挖掘会议（WSDM）。[29] Jiaxi Tang 和 Ke Wang。2018。通过卷积序列嵌入实现个性化 Top-N 序列推荐。载于国际网络搜索与数据挖掘会议（WSDM）。565–573。[30] Changxin Tian、Yuexiang Xie、Yaliang Li、Nan Yang 和 Wayne Xin Zhao。2022。学习去噪不可靠交互，以实现图协同过滤。载于国际信息检索研究与发展会议（SIGIR）。122–132。[31] Petar Velickovic、William Fedus、William L Hamilton、Pietro Liò、Yoshua Bengio 和 R Devon Hjelm。2019。深度图互信息最大化。载于国际学习表征会议（ICLR）。[32] Chenyang Wang、Weizhi Ma、Chong Chen、Min Zhang、Yiqun Liu 和 Shaoping Ma。2023。具有多重对比信号的序列推荐。信息系统汇刊（TOIS）41，1（2023），1–27。[33] Jianling Wang、Kaize Ding、Liangjie Hong、Huan Liu 和 James Caverlee。2020。利用序列超图进行下一物品推荐。载于国际信息检索研究与发展会议（SIGIR）。1101–1110。[34] Jianling Wang、Raphael Louca、Diane Hu、Caitlin Cellier、James Caverlee 和 Liangjie Hong。2020。为情人节购物：电子商务中的购物场景与序列推荐。载于国际网络搜索与数据挖掘会议（WSDM）。645–653。[35] Ziyang Wang、Wei Wei、Gao Cong、Xiao-Li Li、Xian-Ling Mao 和 Minghui Qiu。2020。用于基于会话的推荐的全局上下文增强图神经网络。载于国际信息检索研究与发展会议（SIGIR）。169–178。[36] Wei Wei、Chao Huang、Lianghao Xia、Yong Xu、Jiashu Zhao 和 Dawei Yin。2022。融合行为多重性的推荐对比元学习。载于国际网络搜索与数据挖掘会议（WSDM）。1120–1128。[37] Yinwei Wei、Xiang Wang、Liqiang Nie、Xiangnan He、Richang Hong 和 Tat-Seng Chua。2019。MMGCN：用于微视频个性化推荐的多模态图卷积网络。载于国际多媒体会议（MM）。1437–1445。[38] Shu Wu、Yuyuan Tang、Yanqiao Zhu、Liang Wang、Xing Xie 和 Tieniu Tan。2019。利用图神经网络进行基于会话的推荐。载于 AAAI人工智能会议（AAAI），第33卷。346–353。[39] Lianghao Xia、Chao Huang 和 Chuxu Zhang。2022。用于推荐系统的自监督超图 Transformer。载于国际知识发现与数据挖掘会议（KDD）。2100–2109。[40] Xu Xie、Fei Sun、Zhaoyang Liu、Shiwen Wu、Jinyang Gao、Bolin Ding 和 Bin Cui。2021。用于序列推荐的对比学习。载于国际数据工程会议（ICDE）。[41] An Yan、Shuo Cheng、Wang-Cheng Kang、Mengting Wan 和 Julian McAuley。2019。CosRec：用于序列推荐的二维卷积神经网络。载于国际信息与知识管理会议（CIKM）。2173–2176。[42] Yuhao Yang、Chao Huang、Lianghao Xia、Chunzhen Huang、Da Luo 和 Kangyi Lin。2023。用于序列推荐的去偏对比学习。载于 ACM网络会议（WWW）。1063–1073。[43] Yuhao Yang、Chao Huang、Lianghao Xia、Yuxuan Liang、Yanwei Yu 和 Chenliang Li。2022。用于序列推荐的多行为超图增强 Transformer。载于国际知识发现与数据挖掘会议（KDD）。2263–2274。[44] Wenwen Ye、Shuaiqiang Wang、Xu Chen、Xuepeng Wang、Zheng Qin 和 Dawei Yin。2020。时间至关重要：利用复杂时间信息进行序列推荐。载于国际信息检索研究与发展会议（SIGIR）。1459–1468。[45] Shengyu Zhang、Dong Yao、Zhou Zhao、Tat-Seng Chua 和 Fei Wu。2021。Causerec：用于序列推荐的反事实用户序列合成。载于国际信息检索研究与发展会议（SIGIR）。367–377。[46] Yixin Zhang、Yong Liu、Yonghui Xu、Hao Xiong、Chenyi Lei、Wei He、Lizhen Cui 和 Chunyan Miao。2022。利用图对比学习增强序列推荐。载于国际人工智能联合会议（IJCAI）。[47] Yanqiao Zhu、Yichen Xu、Feng Yu、Qiang Liu、Shu Wu 和 Liang Wang。2021。具有自适应增强的图对比学习。载于 ACM 网络会议（WWW）。