

基于注意力归纳偏差的自注意力顺序推荐

摘要

推荐系统在现代互联网服务中扮演着至关重要的角色，广泛应用于电商、流媒体、社交网络等领域。顺序推荐 (SR) 作为推荐系统的一种重要形式，旨在通过分析用户的历史交互数据预测其下一步可能的行为。近年来，Transformer 模型因其自注意力机制在捕捉序列间长距离依赖性方面的优势，成为 SR 领域的主流技术。然而，现有 Transformer 模型存在“过度平滑”问题，容易忽略序列中的高频细节模式，削弱了对用户短期兴趣的捕捉能力；此外，缺乏显式的归纳偏差 (Inductive Bias)，导致泛化能力不足。为了解决上述问题，研究学者提出了一种新颖的 SR 模型——BSARec。该模型通过引入傅里叶变换，将频域信息融入到模型中，显著增强了对用户行为模式的建模能力。具体而言，BSARec 结合了高通滤波与低通滤波，能够同时捕捉用户的短期兴趣和长期偏好，并通过设计频率重定标器解决了传统 Transformer 模型中自注意力机制的“过度平滑”问题，提高了模型的泛化能力和准确性。基于 BSARec 模型的成功经验，本文进一步提出了一种创新的顺序推荐 FBSARec 模型。FBSARec 在 BSARec 的基础上，引入了超越自注意力交叉融合模块，显著提升了对原始数据特征的提取能力。具体而言，FBSARec 不仅继承了 BSARec 的优点，还通过将傅里叶变换引入到自注意力机制中，实现了频域信息的有效融入，从而增强了对用户行为模式的建模能力。结合高通滤波与低通滤波，FBSARec 能够更全面地捕捉用户的短期兴趣和长期偏好，提供更准确的兴趣表示。实验结果表明，单独使用归纳偏置注意力融合自注意力提取的特征表示的模型性能效果最佳。消融实验显示，归纳偏置注意力在特征表示中发挥了关键作用。然而，多次实验也揭示了超越自注意力交叉融合模块对模型的改进效果有限，提示未来研究可以集中在优化这一模块的设计，以进一步提升模型的整体性能。未来的工作将继续探索如何更好地整合频域信息和自注意力机制，以期进一步提高顺序推荐系统的性能和泛化能力。此外，还将考虑多模态数据融合、动态偏好建模、解释性增强以及大规模数据集测试等方向，以全面提高推荐系统的实用性和鲁棒性。

关键词：注意力机制；顺序推荐；傅里叶变换；特征融合

1 引言

推荐系统在现代互联网服务中扮演着至关重要的角色，广泛应用于电商、流媒体、社交网络等领域。顺序推荐 (Sequential Recommendation, SR) 作为推荐系统的一种重要形式，旨在通过分析用户的历史交互数据预测其下一步可能的行为。近年来，随着 Transformer 模型的成功，特别是其自注意力机制在捕捉序列间长距离依赖性方面的优势，使其成为 SR 领域的主流技术。然而，尽管 Transformer 模型表现出色，仍然面临一些亟需解决的问题。

首先，Transformer 模型中的自注意力机制因其固有的低通滤波特性，容易忽略序列中的高频细节模式。这种“过度平滑”现象会削弱模型对用户短期兴趣的捕捉能力，使其在处理动态偏好变化时表现不佳。其次，现有模型缺乏显式的归纳偏差（Inductive Bias），在建模复杂序列模式时容易过度依赖训练数据，导致泛化能力不足。

为了解决上述问题，研究学者提出了一种新颖的 SR 模型——Beyond Self-Attention for Sequential Recommendation (BSARec)。该模型通过引入傅里叶变换，将频域信息融入到模型中，从而增强对用户行为模式的建模能力。具体而言，BSARec 结合了高通滤波与低通滤波，能够同时捕捉用户的短期兴趣和长期偏好。此外，通过设计频率重定标器，模型在解决过度平滑问题的同时，提高了推荐的准确性和泛化能力。

基于 BSARec 模型的成功经验，本文进一步设计并实现了自注意力与归纳偏置注意力交叉融合模型 FBSARec (Fourier-Based Self-Attention for Sequential Recommendation)。FBSARec 的创新点在于通过引入超越自注意力交叉融合模块，显著提升了模型对原始数据特征的提取能力，并从多个角度综合考量了原始数据。具体而言，该模型通过将傅里叶变换引入到自注意力机制中，实现了频域信息的有效融入，从而增强了对用户行为模式的建模能力。结合高通滤波与低通滤波，FBSARec 能够同时捕捉用户的短期兴趣和长期偏好，提供更全面的兴趣表示。

2 相关工作

2.1 顺序推荐系统的研究进展

顺序推荐 (Sequential Recommendation, SR) 是一个旨在预测用户未来兴趣点的重要研究领域，它通过分析用户的历史行为序列来提供个性化推荐。早期的顺序推荐方法大多基于马尔科夫链模型，这些方法依赖于状态转移概率来捕捉用户行为的时间序列特性。例如，FPMC (Factorized Personalized Markov Chain) 模型结合了矩阵分解和马尔科夫链的思想，不仅能够建模用户的长期偏好，还能考虑短期行为的影响。然而，传统方法通常局限于局部转移模式的识别，对于复杂且全局性的依赖关系则显得力不从心。

随着深度学习技术的发展，顺序推荐进入了以神经网络为核心的新纪元。循环神经网络 (Recurrent Neural Networks, RNNs) 及其变体，如长短期记忆网络 (LSTM) 和门控循环单元 (GRU)，成为了处理序列数据的主流选择。特别是 GRU4Rec [3] 模型，它利用 GRU 有效解决了长序列中的信息丢失问题，能够更好地捕捉长时间跨度内的用户兴趣变化。此外，卷积神经网络 (Convolutional Neural Networks, CNNs) 也被引入到顺序推荐中，如 Caser [10] 模型将用户的行为序列转化为二维图像形式，并通过卷积操作同时捕捉序列中的局部和全局模式，从而提升了推荐系统的性能。

尽管上述基于神经网络的方法在很多方面取得了显著进步，但它们仍然存在一定的局限性。例如，在捕捉长距离依赖关系时，RNN 及其变体可能会遇到梯度消失或梯度爆炸的问题，导致对远距离历史行为的学习效果不佳。同样，CNN 在处理非常长的序列时，也难以维持高效率的同时保持对所有相关上下文的敏感度。为了解决这些问题，研究者们开始探索更先进的架构和技术，包括自注意机制 (Self-Attention Mechanism)、图神经网络 (Graph Neural Networks, GNNs) 等，以期进一步提高顺序推荐的效果和鲁棒性。

近年来，自注意机制的应用为顺序推荐带来了新的突破。自注意模型，如 Transformer，

能够并行处理整个序列中的所有元素，并通过计算各元素之间的注意力得分，实现对任意位置间依赖关系的有效建模。这种能力使得自注意模型特别适合处理包含复杂交互模式的长序列数据，极大地改善了推荐系统的准确性和响应速度。与此同时，GNNs 的引入为理解用户行为提供了新的视角，尤其是在社交网络、商品共现网络等场景下，通过模拟节点间的关联，GNNs 能够揭示出隐藏在用户行为背后的深层结构信息。

2.2 Transformer 在顺序推荐中的应用

近年来，Transformer 架构因其强大的自注意力机制（Self-Attention Mechanism）在自然语言处理和计算机视觉领域取得了巨大成功，并迅速渗透到顺序推荐任务中。这类模型通过引入自注意力机制，能够有效地捕捉用户行为序列中的长距离依赖关系，显著提升了预测的准确性。典型的应用包括 SASRec [4] 和 BERT4Rec [9] 等模型，它们不仅提高了推荐系统的性能，还为用户行为提供了更具解释性的表示，使我们能够更好地理解用户的兴趣演变。

然而，尽管 Transformer 模型带来了诸多优势，但在应用于顺序推荐时也暴露了一些问题。首先，自注意力机制的低通滤波特性可能导致序列嵌入的过度平滑 (oversmoothing)，使得模型难以捕捉用户短期偏好或突然的兴趣变化。这种现象尤其在高维数据集上更为明显，因为自注意力机制倾向于平均化不同位置的信息，从而削弱了对局部特征的关注。其次，缺乏明确的归纳偏差设计可能使模型过度依赖训练数据，限制其泛化能力。具体来说，如果模型仅依赖于训练数据中的模式，那么它在面对未曾见过的数据时可能会表现不佳，无法很好地推广到新的情境。

针对这些问题，近年来的研究尝试从多个角度进行改进。例如，FMLPRec [12] 和 DuoRec [7] 通过增强注意力权重或结合对比学习的方法来缓解上述挑战。这些方法旨在提高模型对短期偏好的敏感度，并引入额外的归纳偏差以增强泛化能力。具体而言，FMLPRec [12] 通过对注意力权重施加频率调制，增强了模型对不同频率信息的区分能力；而 DuoRec [7] 则利用对比学习框架，鼓励模型在正样本与负样本之间形成更清晰的边界，进而提升模型的表达能力和鲁棒性。不过，尽管这些努力取得了一定成效，但效果仍不完全理想，特别是在处理复杂多变的用户行为时，模型的表现仍有待进一步优化。

2.3 傅里叶变换在深度学习中的应用

傅里叶变换作为一种经典的信号处理方法，近年来被引入到深度学习领域，用于增强模型的频率建模能力。在图神经网络中，傅里叶变换常被用来分析和处理节点特征的频率分布，从而缓解过度平滑问题。在推荐系统中，FMLPRec [12] 等方法利用频域信息对模型的注意力机制进行优化，试图在低频和高频信号之间找到平衡。

尽管这些方法初步展示了傅里叶变换的潜力，但其在序列推荐中的应用仍处于起步阶段。特别是如何结合傅里叶变换与自注意力机制，以同时捕捉用户行为的长期偏好和短期动态，仍是一个开放性问题。

2.4 过度平滑问题与对策

过度平滑问题最初是在图神经网络 (Graph Neural Networks, GNNs) 中被提出的，它指的是随着网络层数的增加，节点表示逐渐趋于同质化，丧失了区分性。这一现象不仅限于 GNNs，

在 Transformer 等其他深度学习模型中也有体现。具体来说，在 Transformer 模型中，随着自注意力机制和多层结构的应用，序列嵌入的特征表达可能会逐渐失去多样性，导致模型的预测性能下降。这种现象特别影响到对用户行为序列的建模，因为这些序列通常包含丰富的短期偏好变化和长期兴趣趋势。

为了解决过度平滑问题，现有研究主要通过修改网络结构和引入频率域建模两种方式来增强不同层次特征的多样性。一方面，通过引入层级归一化（Layer Normalization）、残差连接（Residual Connections）以及注意力校准（Attention Calibration）等技术，可以在一定程度上缓解节点表示的同质化，提高模型的表达能力。此外，一些研究还探索了新的激活函数或正则化方法，以防止信息在传递过程中被过度平均化。另一方面，频率域建模通过将时间域信号转换到频域，可以更好地捕捉数据中的周期性和非周期性成分，从而增强模型对复杂模式的理解。例如，FMLPRec 模型通过全局滤波器调整频域信息，能够有效地缓解过度平滑问题，并改善对用户动态兴趣变化的捕捉。此外，结合离散傅里叶变换（DFT）或图傅里叶变换（GFT），提出了基于频谱分析的方法，如 Spectral-GCN 和 Graph Wavelet Network，这些方法进一步提升了模型的鲁棒性和泛化能力。然而，尽管这些方法取得了一定成效，但在应对复杂序列行为和动态兴趣变化时仍显不足。现有的频率域建模方法大多假设数据具有固定的周期性结构，这在实际应用中并不总是成立，尤其是在处理高度动态和异构的数据集时。

3 本文方法

3.1 本文方法概述

作者提出了自注意力与归纳偏置注意力交叉融合的 FBSARec 模型，旨在通过结合特征提取、频域转换、超越自注意交叉融合机制、频率重调器、前馈神经网络和预测模块来更准确地捕捉用户行为模式。具体而言，首先构建了一个特征提取模块，该模块将每个用户的交互项目按时间顺序排列，并通过嵌入矩阵 $\mathbf{M} \in \mathbb{R}^{|\mathcal{V}| \times D}$ 将其映射到潜在空间中，其中 D 表示潜在维度大小。为了提高模型对项目相对位置的敏感性，添加了可训练的位置嵌入 $\mathbf{P} \in \mathbb{R}^{N \times D}$ ，并应用 dropout 和层归一化以增强泛化能力。

接着，作者引入了傅里叶域数值转化模块，利用离散傅里叶变换（DFT）将时域信号投影到频域，区分低频分量（LFC）和高频分量（HFC），以此表示用户的长期兴趣和短期兴趣变化。图傅里叶变换（GFT）被用来推广传统的 DFT，更好地适应复杂数据结构。

为了提升模型表现力，作者创新设计了超越自注意交叉融合模块，包含传统的多头自注意（MSA）并结合基于归纳偏差的滤波器 $\mathbf{A}_{\text{IB}}^\ell$ ，用于调整不同频率分量的比例。通过参数 α 和 β 控制自注意机制与归纳偏差之间的平衡及低频和高频信息的混合比例。同时，对自注意力与归纳偏置注意力进行交叉融合。

此外，框架还包括点对点前馈神经网络（PFFN），在多头自注意力机制之后引入非线性变换，结合残差连接和层归一化技术，确保模型训练的稳定性和效率。

最后，在预测模块中，通过计算用户对项目的偏好得分进行最终推荐。BSARec 整体模型框架详细介绍，如图 1 所示：

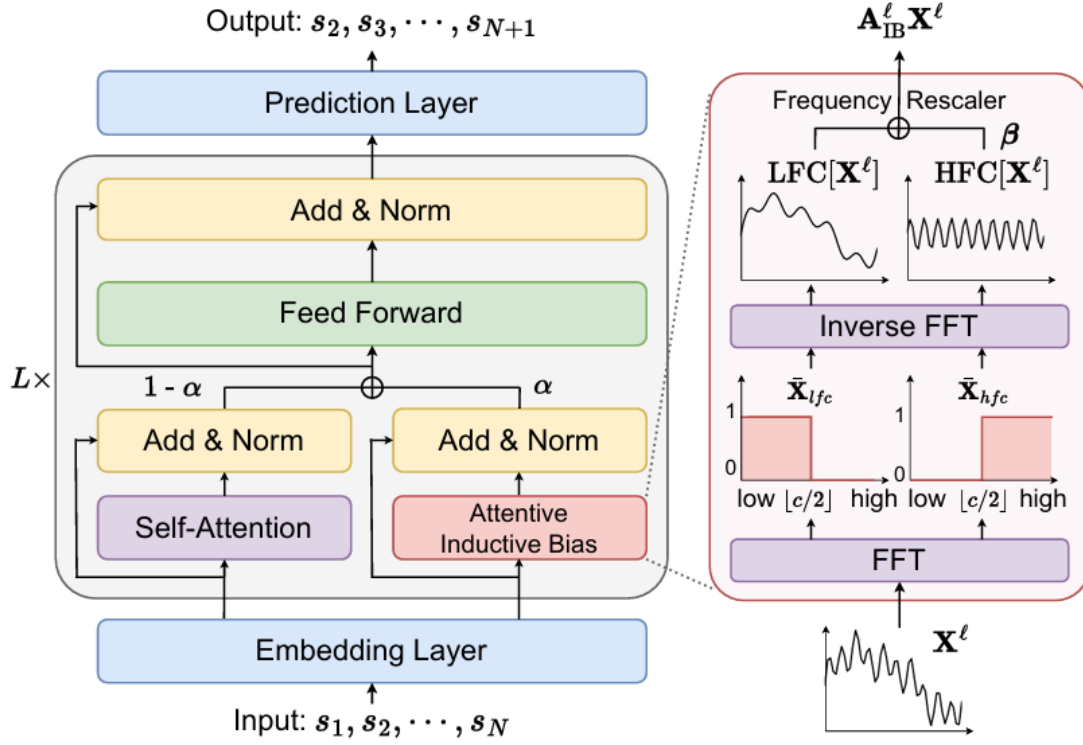


图 1. BSARec 的架构图。BSA 编码器同时使用带有频率重定标器的归纳偏置和原始自注意力。

3.2 特征提取模块

在序列推荐系统中，特征提取模块的目标是预测用户与项目之间的潜在交互。给定一组用户集合 \mathcal{U} 和项目集合 \mathcal{V} ，我们可以将每个用户 $u \in \mathcal{U}$ 的历史交互项目按时间顺序排序为序列 $S^u = [v_1^u, v_2^u, \dots, v_{|S^u|}^u]$ ，其中 v_i^u 表示序列中的第 i 个交互项目。我们的目标是推荐一个 Top- k 的项目列表作为潜在的下一个交互项目。形式上，我们预测 $p(v_{|S^u|+1}^u = v | S^u)$ 。

给定用户的动作序列 S^u 和最大序列长度 N ，如果 $|S^u| > N$ ，则通过移除最早的项目来截断序列；如果 $|S^u| < N$ ，则用 0 填充以获得固定长度的序列 $s = (s_1, s_2, \dots, s_N)$ 。使用项目嵌入矩阵 $\mathbf{M} \in \mathbb{R}^{|\mathcal{V}| \times D}$ ，我们定义序列的嵌入表示 \mathbf{E}^u ，其中 D 是潜在维度大小，且 $\mathbf{E}_i^u = \mathbf{M}_{s_i}$ 。为了使模型对项目的相对位置敏感，我们采用位置嵌入来注入额外的位置信息，同时保持项目嵌入相同的嵌入维度。一个可训练的位置嵌入 $\mathbf{P} \in \mathbb{R}^{N \times D}$ 被添加到按顺序排列的项目嵌入矩阵 \mathbf{E}^u 中。此外，还实现了 dropout 和层归一化：

$$\mathbf{E}^u = \text{Dropout}(\text{LayerNorm}(\mathbf{E}^u + \mathbf{P})). \quad (1)$$

上述公式 1 展示了如何通过添加位置嵌入、应用层归一化和 dropout 来增强项目嵌入矩阵，从而提高模型对项目相对位置的敏感性。

3.3 傅里叶域数值转化模块

离散傅里叶变换 (DFT) 是数字信号处理 (DSP) 中的核心工具，将一系列值投影到频域 (或傅里叶域)。我们通常使用 $\mathcal{F} : \mathbb{R}^N \rightarrow \mathbb{C}^N$ 来表示 DFT，其逆变换 IDFT 为 $\mathcal{F}^{-1} : \mathbb{C}^N \rightarrow$

\mathbb{R}^N 。应用 \mathcal{F} 到一个信号等价于从左侧乘以一个 DFT 矩阵。该矩阵的行由傅里叶基组成，即 $f_j = [e^{2\pi i(j-1) \cdot 0} \dots e^{2\pi i(j-1)(N-1)}]^T / \sqrt{N} \in \mathbb{R}^N$ ，其中 i 是虚数单位， j 表示第 j 行。

对于信号 x 的频谱，记为 $\bar{x} = \mathcal{F}x$ 。我们可以定义 $\bar{x}_{\text{lfc}} \in \mathbb{C}^c$ 包含 \bar{x} 的最低 c 个元素，而 $\bar{x}_{\text{hfc}} \in \mathbb{C}^{N-c}$ 作为包含剩余元素的向量。低频分量 (LFC) 定义如下：

$$\text{LFC}[x] = [f_1, f_2, \dots, f_c] \bar{x}_{\text{lfc}} \in \mathbb{R}^N. \quad (2)$$

相反地，高频分量 (HFC) 定义如下：

$$\text{HFC}[x] = [f_{c+1}, f_{c+2}, \dots, f_N] \bar{x}_{\text{hfc}} \in \mathbb{R}^N. \quad (3)$$

图傅里叶变换 (GFT) 可以被认为是 DFT 向图的推广。换句话说，DFT 是 GFT 的特例，其中使用了 N 个节点的环形图，如图 2 所示。实际上，DFT 是一种将一系列值投影到环形图拉普拉斯矩阵的特征空间的方法（这与傅里叶域相同）。

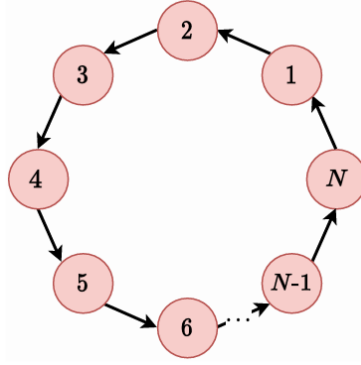


图 2. 具有 N 个节点的环形图。

频率概念也可以用环形图来描述。具有不同符号信号的相邻节点的数量对应于频率。因此，低频信息意味着在 N 个节点上的一系列信号，这些信号的符号不经常变化。在我们的 SR 工作中， N 个节点意味着 N 个项目的嵌入，这样的低频信息意味着用户长期的兴趣，用户长期兴趣和短期兴趣在傅里叶域的体现如图 3 所示。

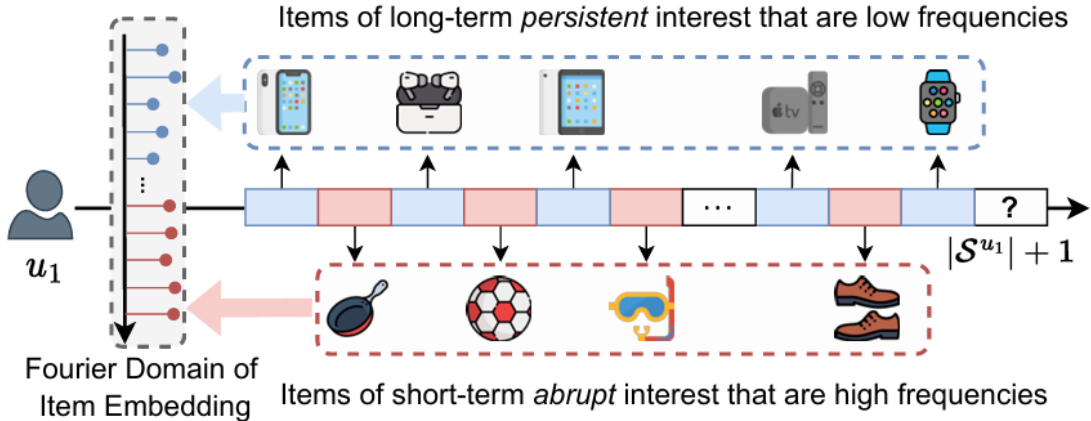


图 3. 顺序推荐系统中高频和低频信号的示例说明。 u_1 的长期持续兴趣和品味构成了嵌入傅里叶域中的低频，而 u_1 兴趣的突然短期变化则对应于高频。

3.4 超越自注意交叉融合模块

设 $\tilde{\mathbf{A}}^\ell$ 为超越自注意 (BSA), $\mathbf{A}_{\text{IB}}^\ell$ 为第 ℓ 层的重新缩放滤波矩阵, \mathbf{X}^ℓ 为第 ℓ 层的输入。当 $\ell = 0$ 时, 我们设置 $\mathbf{X}^0 = \mathbf{E}^u$ 。我们使用以下 BSA 层:

$$\mathbf{S}^\ell = \tilde{\mathbf{A}}^\ell \mathbf{X}^\ell = \alpha \mathbf{A}_{\text{IB}}^\ell \mathbf{X}^\ell + (1 - \alpha) \mathbf{A}^\ell \mathbf{X}^\ell, \quad (4)$$

其中第一项对应于 DSP, 其中利用了离散傅里叶变换, $\alpha \leq 1$ 是一个系数, 用于 (去) 强调归纳偏差。因此, 我们的主要设计点是在验证的归纳偏差和可训练的自注意之间进行权衡。同时, 分别 \mathbf{S}^ℓ 和 $\mathbf{A}^\ell \mathbf{X}^\ell$ 放入前馈神经网络获取特征张量, 再将获得的特征张量进行加权求和, 获得最终的表示张量。

对于在 BSARec 中使用的多头版本, 多头自注意 (MSA) 定义如下:

$$\hat{\mathbf{X}}^\ell = \text{MSA}(\mathbf{X}^\ell) = [\mathbf{S}^1, \mathbf{S}^2, \dots, \mathbf{S}^h] \mathbf{W}^O, \quad (5)$$

其中 h 是头的数量, 投影矩阵 $\mathbf{W}^O \in \mathbb{R}^{D \times D}$ 是可学习参数。

3.5 基于频率重调器的注意归纳偏差模块

本文提出了一种滤波器, 该滤波器注入了注意归纳偏差, 并同时通过将其分为低频和高频分量来调整频率的比例:

$$\mathbf{A}_{\text{IB}}^\ell \mathbf{X}^\ell = \text{LFC}[\mathbf{X}^\ell] + \beta \cdot \text{HFC}[\mathbf{X}^\ell], \quad (6)$$

其中 β 是一个可训练参数, 用于缩放高通滤波器。特别地, β 可以是一个具有 D 维度的向量或一个标量参数。

我们注意到公式 6 中使用了 DFT, 这假设了图 2 中的环形图——从自注意的角度来看, 这种归纳偏差表明购买的项目受到其前一个项目的影。这种注意归纳偏差不需要训练, 因为我们知道它在 SR 中是普遍存在的。

然而, 我们不仅利用归纳偏差, 还提取其低频和高频信息, 学习如何在公式 6 中最优地混合它们。具体来说, 假设一个由 N 个物品嵌入组成的环形图。LFC[·] 提取它们的共同信号, 这些信号不会随着环形图拓扑结构的变化而显著变化, 而 HFC[·] 提取局部波动信号, 如图 3 所示。通过选择性地利用高通信息, 我们可以防止过度平滑问题, 如图 4 所示。如果仅依赖 LFC[·], 我们无法防止过度平滑问题。

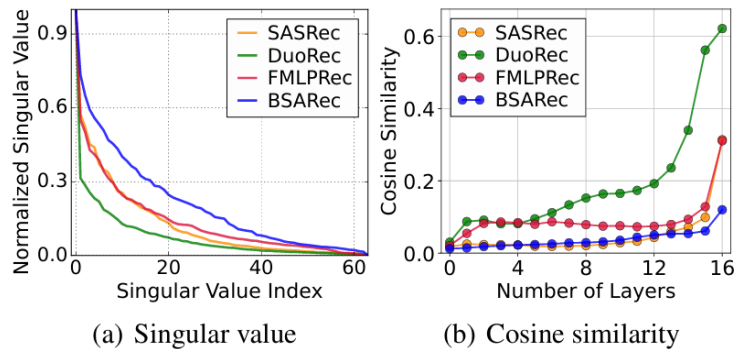


图 4. LastFM 中过度平滑的可视化。用户序列输出嵌入的奇异值和余弦相似度。

此外，我们还在公式 4 中学习自注意矩阵 \mathbf{A}^ℓ ，并将其与我们的注意归纳偏差 $\mathbf{A}_{\text{IB}}^\ell$ 结合。通过将 \mathbf{A}^ℓ 与 $\tilde{\mathbf{A}}^\ell$ 分离，自注意机制专注于捕捉非显式的注意力。

3.6 前馈神经网络模块

多头自注意力函数主要基于线性投影。点对点前馈神经网络 (Point-wise Feed-Forward Network, PFFN) 用于向自注意力块引入非线性。该过程定义如下：

$$\tilde{\mathbf{X}}^\ell = \text{GELU}(\hat{\mathbf{X}}^\ell \mathbf{W}_1^\ell + \mathbf{b}_1^\ell) \mathbf{W}_2^\ell + \mathbf{b}_2^\ell, \quad (7)$$

其中 $\mathbf{W}_1^\ell, \mathbf{W}_2^\ell \in \mathbb{R}^{D \times D}$ 和 $\mathbf{b}_1^\ell, \mathbf{b}_2^\ell \in \mathbb{R}^{D \times D}$ 是可学习的参数。dropout 层、残差连接和层归一化操作如下应用：

$$\mathbf{X}^{\ell+1} = \text{LayerNorm}(\mathbf{X}^\ell + \tilde{\mathbf{X}}^\ell + \text{Dropout}(\tilde{\mathbf{X}}^\ell)). \quad (8)$$

这里：

- \mathbf{X}^ℓ 表示第 ℓ 层的输入特征；
- $\hat{\mathbf{X}}^\ell$ 表示经过多头自注意力机制处理后的特征；
- \mathbf{W}_1^ℓ 和 \mathbf{W}_2^ℓ 分别是第一层和第二层的权重矩阵；
- \mathbf{b}_1^ℓ 和 \mathbf{b}_2^ℓ 分别是第一层和第二层的偏置向量；
- $\text{GELU}(\cdot)$ 是 GELU 激活函数；
- $\text{Dropout}(\cdot)$ 是 dropout 层，用于防止过拟合；
- $\text{LayerNorm}(\cdot)$ 是层归一化操作，用于稳定训练过程。

通过上述公式，点对点前馈神经网络模块能够有效地引入非线性，并通过残差连接和层归一化进一步增强模型的性能。

3.7 预测模块

在 BSARec 的最后一层中，我们计算用户对项目 i 的偏好得分，该得分基于用户的交互历史。这个得分由以下公式给出：

$$\hat{y}_i = p(v_{|\mathcal{S}^u|+1}^u = v | \mathcal{S}^u) = \mathbf{e}_v^T \mathbf{X}_{|\mathcal{S}^u|}^L, \quad (9)$$

其中： \mathbf{e}_v 是项目 v 从矩阵 \mathbf{M} 中得到的表示； $\mathbf{X}_{|\mathcal{S}^u|}^L$ 是第 L 层块在步骤 $|\mathcal{S}^u|$ 的输出；这个点积运算计算了这两个向量之间的相似度，从而给出偏好得分 \hat{y}_i 。

具体来说， \mathbf{e}_v 表示项目 v 的嵌入向量，而 $\mathbf{X}_{|\mathcal{S}^u|}^L$ 表示用户 u 在其交互序列 \mathcal{S}^u 的最后一个项目的特征向量。通过点积运算，我们可以计算出用户对项目 v 的偏好程度。

3.8 损失函数定义

在序列推荐 (Sequential Recommendation, SR) 任务中, 交叉熵 (Cross-Entropy, CE) 损失函数通常用于下一个项目预测任务。该任务被视为在整个项目集上的分类任务 [11] [7] [1]。我们采用交叉熵损失来优化模型参数, 定义如下:

$$\mathcal{L} = -\log \frac{\exp(\hat{y}_g)}{\sum_{i \in |\mathcal{V}|} \exp(\hat{y}_i)}, \quad (10)$$

其中: $g \in |\mathcal{V}|$ 是真实标签项目。 \hat{y}_g 是模型对真实标签项目的预测得分。 \hat{y}_i 是模型对所有项目 i 的预测得分。 \mathcal{V} 是项目集合。

通过上述公式, 交叉熵损失函数计算了模型预测得分与真实标签之间的差异。具体来说, 它通过计算每个项目的预测得分的指数和, 然后除以真实标签项目的预测得分的指数, 最后取负对数得到损失值。这种损失函数能够有效地衡量模型预测的准确性和置信度。

4 复现细节

4.1 与已有开源代码对比

本文参考了 BSARec 模型开源的代码, 但是在其基础上对 BSARec 模型结构进行重新设计, 还创新独立设计了超越自注意力交叉融合模块, 并且重新设计了实验部分对工作进行性能测试, 最终形成了本文的 FBSARec 模型。

4.2 创新点

FBSARec 模型的创新点在于通过引入超越自注意力交叉融合模块, 显著提升了模型对原始数据特征的提取能力, 并从多个角度综合考量了原始数据。具体而言, 该模型通过将傅里叶变换引入到自注意力机制中, 实现了频域信息的有效融入, 从而增强了对用户行为模式的建模能力。结合高通滤波与低通滤波, FBSARec 能够同时捕捉用户的短期兴趣和长期偏好, 提供更全面的兴趣表示。

此外, FBSARec 设计了频率重定标器, 有效解决了过度平滑问题, 提高了推荐的准确性和泛化能力。通过对自注意力获得的特征和归纳偏置注意力获得的特征进行加权求和, FBSARec 综合考虑了用户数据的基本特征以及低频和高频兴趣特征, 从而更好地建模用户兴趣, 提高推荐的准确性和及时性。

作者的主要设计点在于在验证的归纳偏差和可训练的自注意之间进行权衡。通过分别将 \mathbf{S}^ℓ 和 $\mathbf{A}^\ell \mathbf{X}^\ell$ 输入前馈神经网络获取特征张量, 再对这些特征张量进行加权求和, 最终获得综合表示张量, 从而进一步提升模型的表现力。

综上所述, FBSARec 模型通过引入频域信息、优化自注意力机制和增强特征表示, 显著提升了推荐系统的性能和泛化能力。

4.3 实验环境搭建

本课题使用的编程语言是 Python3.9, 并且使用深度学习框架 Pytorch2.1.0 搭建模型, 在 Linux ubuntu 22.04 的操作系统, VSCode 平台进行编程。机器的 CPU 和 GPU 配置分别是:

Intel(R) Core(TM) i9-14900K, NVIDIA GeForce RTX 4090。

4.4 所用模型

为了验证我们模型的有效性，我们将我们的方法与已知的稀疏推荐系统（Sparse Recommendation, SR）基线模型进行比较。这些基线模型分为三类：

- 基于 RNN 或 CNN 的序列模型：GRU4Rec [3] 和 Caser [10]。GRU4Rec 使用门控循环单元（Gated Recurrent Unit, GRU）来捕捉用户的行为序列，其更新公式如下：

$$\mathbf{h}_t = \text{GRU}(\mathbf{x}_t, \mathbf{h}_{t-1}) \quad (11)$$

其中 \mathbf{h}_t 是时间步 t 的隐藏状态， \mathbf{x}_t 是当前时间步的输入。

- 基于 Transformer 的序列模型：SASRec [4]、BERT4Rec [9] 和 FMLPRec [12]。SASRec 使用自注意力机制来捕捉用户行为序列中的依赖关系，其注意力机制公式如下：

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right) \mathbf{V} \quad (12)$$

其中 \mathbf{Q} 、 \mathbf{K} 和 \mathbf{V} 分别是查询、键和值矩阵， d_k 是键向量的维度。

- 基于 Transformer 的序列模型并结合对比学习：DuoRec [7] 和 FEARec [1]。DuoRec 结合了对比学习和自注意力机制，通过最大化不同样本之间的相似性和最小化相同样本之间的差异来增强模型的泛化能力。

4.5 数据集选择

本文在六个不同的稀疏推荐系统（Sparse Recommendation, SR）数据集上评估了我们的模型，这些数据集的稀疏性和领域各不相同。具体包括：i) Amazon Beauty、ii) Amazon Sports、iii) Amazon Toys [6]、iv) Yelp、v) ML-1M [2] 以及 vi) LastFM。

本文遵循 Shin 等人 [8] 使用的的数据预处理步骤，其中所有的评论和评分都被视为隐式反馈（Implicit Feedback）。具体而言，对于每个数据集，我们首先将用户与项目的交互表示为一个二值矩阵 $\mathbf{R} \in \{0, 1\}^{N \times M}$ ，其中 N 是用户的数量， M 是项目的数量。如果用户 u 对项目 i 进行了交互，则 $\mathbf{R}_{ui} = 1$ ；否则， $\mathbf{R}_{ui} = 0$ 。

为了进一步处理数据集中的稀疏性问题，我们采用了负采样技术。对于每个正样本 (u, i) ，我们随机选择 k 个未交互的项目作为负样本。这样可以确保训练过程中正负样本的比例平衡，从而提高模型的泛化能力。

4.6 评估指标

为了衡量推荐的准确性，我们通常使用广泛使用的 Top- k 指标，包括 HR@ k （Hit Rate）和 NDCG@ k （Normalized Discounted Cumulative Gain），来评估推荐列表。其中 k 被设置为 5、10 和 20。为了确保公平和全面的比较，我们分析了在整个项目集上的排名结果，而不进行负样本采样 [5]。

HR@k (Hit Rate)

Hit Rate 是一个常用的评估指标，用于衡量推荐列表中至少有一个正确项目的概率。其定义如下：

$$\text{HR@}k = \frac{\sum_{u=1}^N \mathbb{I}(r_u \in L_u^k)}{N}$$

其中 $\mathbb{I}(r_u \in L_u^k)$ 是指示函数，如果用户 u 的真实项目 r_u 在前 k 个推荐项目列表 L_u^k 中，则返回 1，否则返回 0； N 是用户的总数。

NDCG@k (Normalized Discounted Cumulative Gain)

NDCG 是一种考虑推荐项目排序的评估指标，它根据项目的相关性对推荐列表进行加权。其定义如下：

$$\text{NDCG@}k = \frac{\text{DCG@}k}{\text{IDCG@}k}$$

其中 $\text{DCG@}k$ 表示在前 k 个推荐项目中的累积增益，计算公式为：

$$\text{DCG@}k = \sum_{i=1}^k \frac{2^{rel_i} - 1}{\log_2(i + 1)}$$

这里 rel_i 是第 i 个项目的相关性得分。 $\text{IDCG@}k$ 表示理想情况下的累积增益，即所有项目按相关性从高到低排序时的 DCG 值。

4.7 参数设置

我们在实验中采用了以下推荐的超参数。系数 α 从集合 $\{0.1, 0.3, 0.5, 0.7, 0.9\}$ 中选择，而 c 从集合 $\{1, 3, 5, 7, 9\}$ 中选择。FBSA 块的数量 L 被设置为 2，Transformer 中的头数 h 从集合 $\{1, 2, 4\}$ 中选择。特征维度 D 被设置为 64，最大序列长度 N 被设置为 50。

在训练过程中，我们使用 Adam 优化器进行优化，学习率从集合 $\{5 \times 10^{-4}, 1 \times 10^{-3}\}$ 中选择，批量大小被设置为 256。

5 实验结果分析

实验中，超越自注意力交叉融合模块使用自注意力提取的特征表示的结果如表 1 所示。实验中，超越自注意力交叉融合模块使用归纳偏置注意力提取的特征表示的结果如表 2 所示。实验中，对自注意力提取的特征表示与归纳偏置注意力提取的特征表示进行交叉融合的结果如表 3 所示。实验中，对归纳偏置注意力融合自注意力提取的特征表示与自注意力提取的特征表示进行交叉融合的结果如表 4 所示。实验中，对归纳偏置注意力融合自注意力提取的特征表示与归纳偏置注意力提取的特征表示进行交叉融合的结果如表 5 所示。实验中，对归纳偏置注意力融合自注意力提取的特征表示与自注意力提取的特征表示，和归纳偏置注意力提取的特征表示进行交叉融合的结果如表 6 所示。实验结果表明，单独使用归纳偏置注意力融合自注意力提取的特征表示的模型性能效果最佳，消融实验表明其中归纳偏置注意力提取的特征表示发挥着最大的作用。同时多次实验结果表明，超越自注意力交叉融合模块对模型的改进效果一般，仍然有待改进。

6 总结与展望

本文围绕顺序推荐 (Sequential Recommendation, SR) 领域中的关键问题进行了深入探讨,并提出了一种创新的模型——FBSARec (Fourier-Based Self-Attention for Sequential Recommendation)。通过引入傅里叶变换和超越自注意力交叉融合模块, FBSARec 模型显著提升了对用户行为模式的建模能力。

具体而言, FBSARec 结合了高通滤波与低通滤波,能够同时捕捉用户的短期兴趣和长期偏好,解决了传统 Transformer 模型中自注意力机制的“过度平滑”问题。此外,通过设计频率重定标器, FBSARec 提高了模型的泛化能力,减少了对训练数据的依赖,从而在复杂序列模式建模中表现出色。作者的主要设计点在于在验证的归纳偏差和可训练的自注意之间进行权衡,通过分别将 \mathbf{S}^ℓ 和 $\mathbf{A}^\ell \mathbf{X}^\ell$ 输入前馈神经网络获取特征张量,并对这些特征张量进行加权求和,最终获得综合表示张量,进一步提升了模型的表现力。

实验结果表明,单独使用归纳偏置注意力融合自注意力提取的特征表示的模型性能效果最佳。消融实验显示,归纳偏置注意力在特征表示中发挥了关键作用。然而,多次实验也揭示了超越自注意力交叉融合模块对模型的改进效果有限,这提示我们未来的研究可以集中在优化这一模块的设计,以进一步提升模型的整体性能。

FBSARec 模型在多个方面仍然存在一些值得进一步探索的方向。首先,目前超越自注意力交叉融合模块的效果尚未达到预期。未来的研究可以尝试不同的架构设计和参数调整,以提高其对模型性能的贡献。其次,除了用户的历史交互数据外,还可以考虑融合其他类型的多模态数据(如文本、图像等),以更全面地理解用户的行为和偏好。此外,进一步研究如何更好地捕捉用户的动态偏好变化,特别是短期兴趣的快速变化,以提高推荐系统的实时性和准确性也是一个重要的方向。当前深度学习模型大多为黑箱模型,缺乏解释性。未来可以探索如何使推荐系统更加透明和可解释,以便更好地服务于实际应用。最后,在更大规模的数据集上的表现仍有待检验。未来的工作应继续扩展实验范围,确保模型在不同场景下的稳定性和鲁棒性。

综上所述,本文不仅提出了一个创新的模型 FBSARec,还在理论上探讨了归纳偏置注意力的重要性,并通过实验证明了其在推荐系统中的有效性。未来的工作将继续探索如何更好地整合频域信息和自注意力机制,以期进一步提高顺序推荐系统的性能和泛化能力。

参考文献

- [1] X. Du, H. Yuan, P. Zhao, J. Qu, F. Zhuang, G. Liu, Y. Liu, and V. S. Sheng. Frequency enhanced hybrid attention network for sequential recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 78–88, 2023.
- [2] F. Maxwell Harper and Joseph A. Konstan. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TIIS)*, 5(4):1–19, 2015.
- [3] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. In *International Conference on Learning Representations (ICLR)*, 2016.

- [4] W.-C. Kang and J. McAuley. Self-attentive sequential recommendation. In *IEEE International Conference on Data Mining (ICDM)*, pages 197–206, 2018.
- [5] Wahid Krichene and Steffen Rendle. On sampled metrics for item recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, pages 1748–1757. ACM, 2020.
- [6] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–52. ACM, 2015.
- [7] R. Qiu, Z. Huang, H. Yin, and Z. Wang. Contrastive learning for representation degeneration problem in sequential recommendation. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 813–823, 2022.
- [8] Yongjun Shin, Joonseok Choi, Heeyoung Wi, and Nojun Park. An attentive inductive bias for sequential recommendation beyond the self-attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8984–8992, 2024.
- [9] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1441–1450, 2019.
- [10] Jia Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM)*, pages 565–573. ACM, 2018.
- [11] T. Zhang, P. Zhao, Y. Liu, V. S. Sheng, J. Xu, D. Wang, G. Liu, and X. Zhou. Feature-level deeper self attention network for sequential recommendation. In *IJCAI*, pages 4320–4326, 2019.
- [12] K. Zhou, H. Yu, W. X. Zhao, and J.-R. Wen. Filter-enhanced mlp is all you need for sequential recommendation. In *Proceedings of the Web Conference 2022 (TheWebConf)*, pages 2388–2399, 2022.

附录

实验结果表格

表 1. 超越自注意力交叉融合模块使用自注意力提取的特征表示的结果对比

Datasets	Metric	Caser	GRU4Rec	SASRec	BERT4Rec	FMLPRec	DuoRec	FEARec	BSARec	FBSARec	Improv
Beauty	HR@5	0.0125	0.0169	0.0340	0.0469	0.0346	0.0707	0.0706	<u>0.0736</u>	0.0678	-7.90%
	HR@10	0.0225	0.0304	0.0531	0.0705	0.0559	0.0965	0.0982	<u>0.1008</u>	0.0915	-9.22%
	HR@20	0.0403	0.0527	0.0823	0.1073	0.0869	0.1313	0.1352	<u>0.1373</u>	0.1255	-8.59%
	NDCG@5	0.0076	0.0104	0.0221	0.0311	0.0222	0.0501	0.0512	<u>0.0523</u>	0.0489	-6.46%
	NDCG@10	0.0108	0.0147	0.0356	0.0387	0.0291	0.0584	0.0601	<u>0.0611</u>	0.0565	-7.53%
	NDCG@20	0.0153	0.0203	0.0188	0.0480	0.0369	0.0671	0.0694	<u>0.0703</u>	0.0650	-7.54%
Sports	HR@5	0.0091	0.0118	0.0118	0.0275	0.0220	0.0396	0.0411	<u>0.0426</u>	0.0385	-9.62%
	HR@10	0.0163	0.0187	0.0298	0.0428	0.0336	0.0569	0.0589	<u>0.0612</u>	0.0544	-11.11%
	HR@20	0.0260	0.0303	0.0459	0.0649	0.0525	0.0791	0.0836	<u>0.0858</u>	0.0778	-9.32%
	NDCG@5	0.0056	0.0079	0.0124	0.0180	0.0146	0.0276	0.0286	<u>0.0300</u>	0.0274	-8.67%
	NDCG@10	0.0080	0.0101	0.0159	0.0229	0.0183	0.0331	0.0343	<u>0.0360</u>	0.0325	-9.72%
	NDCG@20	0.0104	0.0131	0.0200	0.0284	0.0231	0.0387	0.0405	<u>0.0422</u>	0.0384	-8.96%
Toys	HR@5	0.0095	0.0121	0.0440	0.0412	0.0432	0.0770	0.0783	<u>0.0805</u>	0.0773	-4.00%
	HR@10	0.0161	0.0211	0.0652	0.0635	0.0671	0.1034	0.1054	<u>0.1081</u>	0.1030	-4.72%
	HR@20	0.0268	0.0348	0.0929	0.0939	0.0974	0.1369	0.1397	<u>0.1435</u>	0.1329	-7.40%
	NDCG@5	0.0058	0.0077	0.0297	0.0282	0.0288	0.0568	0.0574	<u>0.0589</u>	0.0573	-2.75%
	NDCG@10	0.0079	0.0106	0.0366	0.0353	0.0365	0.0653	0.0661	<u>0.0679</u>	0.0656	-3.38%
	NDCG@20	0.0106	0.0140	0.0435	0.0430	0.0441	0.737	0.0747	<u>0.0768</u>	0.0731	-4.81%
Yelp	HR@5	0.0117	0.0130	0.0149	0.0256	0.0159	0.0271	0.0262	<u>0.0275</u>	0.0233	-15.27%
	HR@10	0.0197	0.0221	0.0249	0.0433	0.0287	0.0442	0.0437	<u>0.0465</u>	0.0395	-15.05%
	HR@20	0.0337	0.0383	0.0424	0.0717	0.0490	0.0717	0.0691	<u>0.0746</u>	0.0633	-15.14%
	NDCG@5	0.0070	0.0080	0.0091	0.0159	0.0100	0.0170	0.0165	<u>0.0170</u>	0.0144	-15.29%
	NDCG@10	0.0096	0.0109	0.0123	0.0216	0.0142	0.0225	0.0221	<u>0.0231</u>	0.0196	-15.15%
	NDCG@20	0.0131	0.0150	0.0167	0.0287	0.0192	0.0294	0.0285	<u>0.0302</u>	0.0256	-15.23%
LastFM	HR@5	0.0303	0.0312	0.0413	0.0294	0.0367	0.0431	0.0431	<u>0.0523</u>	0.0394	-24.66%
	HR@10	0.0431	0.0404	0.0633	0.0459	0.0560	0.0624	0.0587	<u>0.0807</u>	0.0587	-27.27%
	HR@20	0.0642	0.0541	0.0927	0.0596	0.0826	0.0963	0.0826	<u>0.1174</u>	0.0899	-23.46%
	NDCG@5	0.0227	0.0217	0.0284	0.0198	0.0243	0.0300	0.0304	<u>0.0344</u>	0.0279	-18.89%
	NDCG@10	0.0268	0.0245	0.0355	0.0252	0.0306	0.0361	0.0354	<u>0.0435</u>	0.0341	-21.61%
	NDCG@20	0.0321	0.0280	0.0429	0.0286	0.0372	0.0446	0.0414	<u>0.0526</u>	0.0420	-20.15%
ML-1M	HR@5	0.0927	0.1005	0.1374	0.1512	0.1316	0.1838	0.1834	<u>0.1944</u>	0.1932	-0.62%
	HR@10	0.1556	0.1657	0.2137	0.2346	0.2065	0.2704	0.2705	<u>0.2757</u>	0.2833	2.76%
	HR@20	0.2488	0.2664	0.3245	0.3440	0.3137	0.3738	0.3714	<u>0.3884</u>	0.3863	-0.54%
	NDCG@5	0.0592	0.0619	0.0873	0.1021	0.0846	0.1252	0.1236	<u>0.1306</u>	0.1325	1.46%
	NDCG@10	0.0795	0.0828	0.1116	0.1289	0.1087	0.1530	0.1516	<u>0.1568</u>	0.1616	3.06%
	NDCG@20	0.1028	0.1081	0.1395	0.1564	0.1356	0.1790	0.1771	<u>0.1851</u>	0.1873	1.19%

表 2. 超越自注意力交叉融合模块使用归纳偏置注意力提取的特征表示的结果对比

Datasets	Metric	BSARec	FBSARec	Improv
Beauty	HR@5	<u>0.0736</u>	0.0711	-3.40%
	NDCG@5	<u>0.0523</u>	0.0508	-2.87%
	HR@10	<u>0.1008</u>	0.0981	-2.68%
	NDCG@10	<u>0.0611</u>	0.0595	-2.62%
继续下页				

表 2 – 续前页

Datasets	Metric	BSARec	FBSARec	Improv
	HR@20	<u>0.1373</u>	0.1352	-1.53%
	NDCG@20	<u>0.0703</u>	0.0688	-2.14%
Sports	HR@5	<u>0.0426</u>	0.0420	-1.41%
	NDCG@5	<u>0.0300</u>	0.0294	-2.00%
	HR@10	<u>0.0612</u>	0.0589	-3.76%
	NDCG@10	<u>0.0360</u>	0.0348	-3.33%
	HR@20	<u>0.0858</u>	0.0832	-3.03%
	NDCG@20	<u>0.0422</u>	0.0410	-2.84%
Toys	HR@5	<u>0.0805</u>	0.0793	-1.49%
	NDCG@5	<u>0.0589</u>	0.0576	-2.24%
	HR@10	<u>0.1081</u>	0.1076	-0.46%
	NDCG@10	<u>0.0679</u>	0.0667	-1.77%
	HR@20	<u>0.1435</u>	0.1422	-0.91%
	NDCG@20	<u>0.0768</u>	0.0755	-1.69%
Yelp	HR@5	<u>0.0275</u>	0.0256	-6.91%
	NDCG@5	<u>0.0170</u>	0.0160	-5.88%
	HR@10	<u>0.0465</u>	0.0422	-9.25%
	NDCG@10	<u>0.0231</u>	0.0213	-7.79%
	HR@20	<u>0.0746</u>	0.0689	-7.64%
	NDCG@20	<u>0.0302</u>	0.0280	-7.28%
LastFM	HR@5	<u>0.0523</u>	0.0450	-14.00%
	NDCG@5	<u>0.0344</u>	0.0288	-16.28%
	HR@10	<u>0.0807</u>	0.0615	-23.79%
	NDCG@10	<u>0.0435</u>	0.0340	-21.84%
	HR@20	<u>0.1174</u>	0.0927	-20.95%
	NDCG@20	<u>0.0526</u>	0.0417	-20.72%
ML-1M	HR@5	<u>0.1944</u>	0.1917	-1.44%
	NDCG@5	<u>0.1306</u>	0.1260	-3.52%
	HR@10	<u>0.2757</u>	0.2772	0.54%
	NDCG@10	<u>0.1568</u>	0.1534	-2.17%
	HR@20	<u>0.3884</u>	0.3825	-1.52%
	NDCG@20	<u>0.1851</u>	0.1801	-2.70%

表 3. 对自注意力提取的特征表示与归纳偏置注意力提取的特征表示进行交叉融合的结果对比

Datasets	Metric	BSARec	FBSARec	Improv
Beauty	HR@5	0.0736	0.0693	-5.84%
	NDCG@5	0.0523	0.0500	-4.40%
	HR@10	0.1008	0.0955	-5.26%
	NDCG@10	0.0611	0.0583	-4.58%
	HR@20	0.1373	0.1292	-5.90%
	NDCG@20	0.0703	0.0668	-4.98%
Sports	HR@5	0.0426	0.0394	-7.51%
	NDCG@5	0.0300	0.0272	-9.33%
	HR@10	0.0612	0.0558	-8.82%
	NDCG@10	0.0360	0.0325	-9.72%
	HR@20	0.0858	0.0776	-9.56%
	NDCG@20	0.0422	0.0379	-10.19%
Toys	HR@5	0.0805	0.0780	-3.11%
	NDCG@5	0.0589	0.0578	-1.87%
	HR@10	0.1081	0.1034	-4.35%
	NDCG@10	0.0679	0.0660	-2.80%
	HR@20	0.1435	0.1385	-3.48%
	NDCG@20	0.0768	0.0748	-2.60%
Yelp	HR@5	0.0275	0.0242	-12.00%
	NDCG@5	0.0170	0.0157	-7.65%
	HR@10	0.0465	0.0411	-11.61%
	NDCG@10	0.0231	0.0211	-8.66%
	HR@20	0.0746	0.0672	-9.92%
	NDCG@20	0.0302	0.0277	-8.28%
LastFM	HR@5	0.0523	0.0459	-12.20%
	NDCG@5	0.0344	0.0311	-9.59%
	HR@10	0.0807	0.0606	-24.90%
	NDCG@10	0.0435	0.0356	-18.16%
	HR@20	0.1174	0.0844	-27.94%
	NDCG@20	0.0526	0.0418	-20.53%
ML-1M	HR@5	0.1944	0.1856	-4.53%
	NDCG@5	0.1306	0.1236	-5.36%
	HR@10	0.2757	0.2702	-1.99%
	NDCG@10	0.1568	0.1508	-3.83%
	HR@20	0.3884	0.3757	-3.27%
继续下页				

表 3 – 续前页

Datasets	Metric	BSARec	FBSARec	Improv
	NDCG@20	0.1851	0.1774	-4.16%

表 4. 对归纳偏置注意力融合自注意力提取的特征表示与自注意力提取的特征表示进行交叉融合的结果对比

Datasets	Metric	BSARec	FBSARec	Improv
Beauty	HR@5	0.0736	0.0698	-5.15%
	NDCG@5	0.0523	0.0502	-4.02%
	HR@10	0.1008	0.0969	-3.87%
	NDCG@10	0.0611	0.0590	-3.44%
	HR@20	0.1373	0.1320	-3.89%
	NDCG@20	0.0703	0.0678	-3.56%
Sports	HR@5	0.0426	0.0401	-5.87%
	NDCG@5	0.0300	0.0284	-5.33%
	HR@10	0.0612	0.0582	-4.90%
	NDCG@10	0.0360	0.0342	-5.00%
	HR@20	0.0858	0.0824	-3.97%
	NDCG@20	0.0422	0.0403	-4.49%
Toys	HR@5	0.0805	0.0791	-1.75%
	NDCG@5	0.0589	0.0574	-2.55%
	HR@10	0.1081	0.1061	-1.85%
	NDCG@10	0.0679	0.0661	-2.66%
	HR@20	0.1435	0.1398	-2.58%
	NDCG@20	0.0768	0.0746	-2.86%
Yelp	HR@5	0.0275	0.0252	-8.36%
	NDCG@5	0.0170	0.0160	-5.88%
	HR@10	0.0465	0.0427	-8.17%
	NDCG@10	0.0231	0.0216	-6.47%
	HR@20	0.0746	0.0701	-6.05%
	NDCG@20	0.0302	0.0285	-5.63%
LastFM	HR@5	0.0523	0.0459	-12.10%
	NDCG@5	0.0344	0.0335	-2.61%
	HR@10	0.0807	0.0697	-13.61%
	NDCG@10	0.0435	0.0412	-5.28%
	HR@20	0.1174	0.0991	-15.56%
	NDCG@20	0.0526	0.0486	-7.58%
	HR@5	0.1944	0.1902	-2.15%

继续下页

表 4 – 续前页

Datasets	Metric	BSARec	FBSARec	Improv
ML-1M	NDCG@5	0.1306	0.1289	-1.30%
	HR@10	0.2757	0.2750	-0.25%
	NDCG@10	0.1568	0.1562	-0.38%
	HR@20	0.3884	0.3825	-1.53%
	NDCG@20	0.1851	0.1832	-1.03%

表 5. 对归纳偏置注意力融合自注意力提取的特征表示与归纳偏置注意力提取的特征表示进行交叉融合的结果对比

Datasets	Metric	BSARec	FBSARec	Improv
Beauty	HR@5	<u>0.0736</u>	0.0705	-1.68%
	NDCG@5	<u>0.0523</u>	0.0510	-2.48%
	HR@10	<u>0.1008</u>	0.0967	-4.03%
	NDCG@10	<u>0.0611</u>	0.0594	-2.79%
	HR@20	<u>0.1373</u>	0.1285	-6.44%
	NDCG@20	<u>0.0703</u>	0.0674	-4.12%
Sports	HR@5	<u>0.0426</u>	0.0407	-4.45%
	NDCG@5	<u>0.0300</u>	0.0285	-5.00%
	HR@10	<u>0.0612</u>	0.0584	-4.58%
	NDCG@10	<u>0.0360</u>	0.0342	-5.00%
	HR@20	<u>0.0858</u>	0.0815	-5.01%
	NDCG@20	<u>0.0422</u>	0.0400	-5.22%
Toys	HR@5	<u>0.0805</u>	0.0773	-3.97%
	NDCG@5	<u>0.0589</u>	0.0572	-2.89%
	HR@10	<u>0.1081</u>	0.1026	-5.05%
	NDCG@10	<u>0.0679</u>	0.0653	-3.82%
	HR@20	<u>0.1435</u>	0.1385	-3.50%
	NDCG@20	<u>0.0768</u>	0.0744	-3.13%
Yelp	HR@5	<u>0.0275</u>	0.0236	-14.18%
	NDCG@5	<u>0.0170</u>	0.0151	-11.18%
	HR@10	<u>0.0465</u>	0.0403	-13.30%
	NDCG@10	<u>0.0231</u>	0.0205	-11.21%
	HR@20	<u>0.0746</u>	0.0668	-10.47%
	NDCG@20	<u>0.0302</u>	0.0271	-10.25%
LastFM	HR@5	<u>0.0523</u>	0.0450	-13.22%
	NDCG@5	<u>0.0344</u>	0.0298	-13.40%
	HR@10	<u>0.0807</u>	0.0587	-27.24%

继续下页

表 5 – 续前页

Datasets	Metric	BSARec	FBSARec	Improv
	NDCG@10	<u>0.0435</u>	0.0342	-21.34%
	HR@20	<u>0.1174</u>	0.0872	-25.74%
	NDCG@20	<u>0.0526</u>	0.0414	-21.26%
ML-1M	HR@5	<u>0.1944</u>	0.1854	-4.64%
	NDCG@5	<u>0.1306</u>	0.1247	-4.51%
	HR@10	<u>0.2757</u>	0.2725	-1.17%
	NDCG@10	<u>0.1568</u>	0.1528	-2.56%
	HR@20	<u>0.3884</u>	0.3815	-1.78%
	NDCG@20	<u>0.1851</u>	0.1804	-2.52%

表 6. 对归纳偏置注意力融合自注意力提取的特征表示与自注意力提取的特征表示, 和归纳偏置注意力提取的特征表示进行交叉融合的结果对比

Datasets	Metric	BSARec	FBSARec	Improv
Beauty	HR@5	<u>0.0736</u>	0.0699	-5.02%
	NDCG@5	<u>0.0523</u>	0.0496	-5.16%
	HR@10	<u>0.1008</u>	0.0966	-4.23%
	NDCG@10	<u>0.0611</u>	0.0581	-4.91%
	HR@20	<u>0.1373</u>	0.1288	-6.22%
	NDCG@20	<u>0.0703</u>	0.0662	-5.84%
Sports	HR@5	<u>0.0426</u>	0.0390	-8.46%
	NDCG@5	<u>0.0300</u>	0.0276	-7.99%
	HR@10	<u>0.0612</u>	0.0562	-8.18%
	NDCG@10	<u>0.0360</u>	0.0331	-8.06%
	HR@20	<u>0.0858</u>	0.0807	-5.89%
	NDCG@20	<u>0.0422</u>	0.0393	-6.89%
Toys	HR@5	<u>0.0805</u>	0.0784	-1.97%
	NDCG@5	<u>0.0589</u>	0.0577	-2.03%
	HR@10	<u>0.1081</u>	0.1044	-3.42%
	NDCG@10	<u>0.0679</u>	0.0661	-2.65%
	HR@20	<u>0.1435</u>	0.1376	-4.11%
	NDCG@20	<u>0.0768</u>	0.0745	-3.00%
Yelp	HR@5	<u>0.0275</u>	0.0248	-9.82%
	NDCG@5	<u>0.0170</u>	0.0154	-9.41%
	HR@10	<u>0.0465</u>	0.0413	-11.21%
	NDCG@10	<u>0.0231</u>	0.0207	-10.39%
	HR@20	<u>0.0746</u>	0.0662	-11.29%

继续下页

表 6 – 续前页

Datasets	Metric	BSARec	FBSARec	Improv
	NDCG@20	<u>0.0302</u>	0.0270	-10.57%
LastFM	HR@5	<u>0.0523</u>	0.0505	-3.44%
	NDCG@5	<u>0.0344</u>	0.0335	-2.63%
	HR@10	<u>0.0807</u>	0.0670	-17.03%
	NDCG@10	<u>0.0435</u>	0.0387	-11.06%
	HR@20	<u>0.1174</u>	0.0954	-18.72%
	NDCG@20	<u>0.0526</u>	0.0458	-12.94%
ML-1M	HR@5	<u>0.1944</u>	0.1874	-3.61%
	NDCG@5	<u>0.1306</u>	0.1271	-2.68%
	HR@10	<u>0.2757</u>	0.2796	1.42%
	NDCG@10	<u>0.1568</u>	0.1569	0.06%
	HR@20	<u>0.3884</u>	0.3836	-1.24%
	NDCG@20	<u>0.1851</u>	0.1831	-1.08%