arXiv:2303.06337v1 [cs.IR] 11 Mar 2023

AutoMLP：面向序列推荐的自动化多层感知机

Muyang Li*香港城市大学
悉尼大学
m
uli0371@uni.sydney.edu.au

Zijian Zhang*香港城市大学
吉林大学
zhangzj2114
@mails.jlu.edu.cn

Xiangyu Zhao†香港城市大学
xianzhao@cityu.edu.hk

Wanyu Wang 香港城市大学
w
anyuwang4-c@my.cityu.edu.hk

Minhao Zhao 伏羲人工智能实验室，网易
zhaominghao@corp.netease.com

Runze Wu 伏羲人工智能实验室，网易
wurunze1@corp.netease.com

Ruocheng Guo 字节跳动人工智能实验室
rguo.asu@gmail.com

摘要

序列推荐系统旨在根据用户的历史交互，预测其下一次感兴趣的物品。然而，长期以来的一个难题是如何区分用户的长期/短期兴趣，因为这些兴趣可能具有异质性，对下一次推荐的贡献也各不相同。现有方法通常通过穷举搜索或经验来预先设定短期兴趣长度，但这种做法要么效率极低，要么效果不佳。尽管近期先进的基于 Transformer 的模型能够取得最先进的性能，但它们的计算复杂度会随输入序列长度呈二次增长。为此，本文提出一种新颖的序列推荐系统 AutoMLP，旨在更好地根据用户的历史交互建模其长期/短期兴趣。此外，我们设计了一种自动化且自适应的搜索算法，通过端到端优化确定合适的短期兴趣长度。大量实验表明，AutoMLP 在保持线性计算复杂度的同时，性能可与最先进的方法相媲美。



图 1：用户长期/短期兴趣的示例。用户 1 和用户 2 的目标物品分别与其长期兴趣和短期兴趣相关，而用户 3 则同时受到两者的影响。

CCS 概念

• **信息系统** →**推荐系统**。

关键词

序列推荐、多层感知机、自动化机器学习

ACM 引用格式：

Muyang Li、Zijian Zhang、赵翔宇、Wanyu Wang、Minhao Zhao、Runze Wu 和 Ruocheng Guo。2023。AutoMLP：用于序列推荐的自动化多层感知机。载于*ACM Web Conference 2023*（*WWW '23*），*2023*年*5*月*1*日至*5*日，美国德克萨斯州奥斯汀。美国德克萨斯州奥斯汀，9页。https: //doi.org/10.1145/3543507.3583440

*两位作者对本研究的贡献相同。
†Xiangyu Zhao 是通讯作者。

允许免费为个人或课堂用途制作本作品全部或部分内容的数字副本或纸质副本，前提是不得为牟利或商业利益制作或分发副本，且副本须附有本声明及完整引文

在首页。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。注明出处后可以进行摘要。若要以其他方式复制、重新发表、发布到服务器或分发至列表，必须事先获得特别许可和／或支付费用。请向 permissions@acm.org 申请许可。*WWW '23*，*2023*年*5*月*1*日至*5*日，美国德克萨斯州奥斯汀© 2023 版权所有归所有者／作者所有。出版权已授权给 ACM。ACM国际标准书号 978-1-4503-9416-1/23/04...$15.00 https://doi.org/10.1145/3543507.3583440

1 引言

序列推荐系统在现实世界的服务提供商中发挥着重要作用，例如购买预测[34, 36, 41]，网页推荐 [7]，以及下一兴趣点推荐 [3, 9, 24]。它基于用户的历史行为对序列依赖关系进行建模；由于利用了序列信息 [4, 5, 21]，其表现始终优于静态推荐模型。

做出明智的序列推荐所需的关键信息可归纳为三个方面。第一方面是用户长期序列依赖，与交互物品的整个序列有关。它体现了用户相对稳定的长期兴趣。第二方面是用户短期序列依赖，在动态推荐环境中可能偏离长期兴趣。它描述了用户最近的兴趣，这种兴趣可能暂时独立存在。此外，物品特征对于阐释用户的

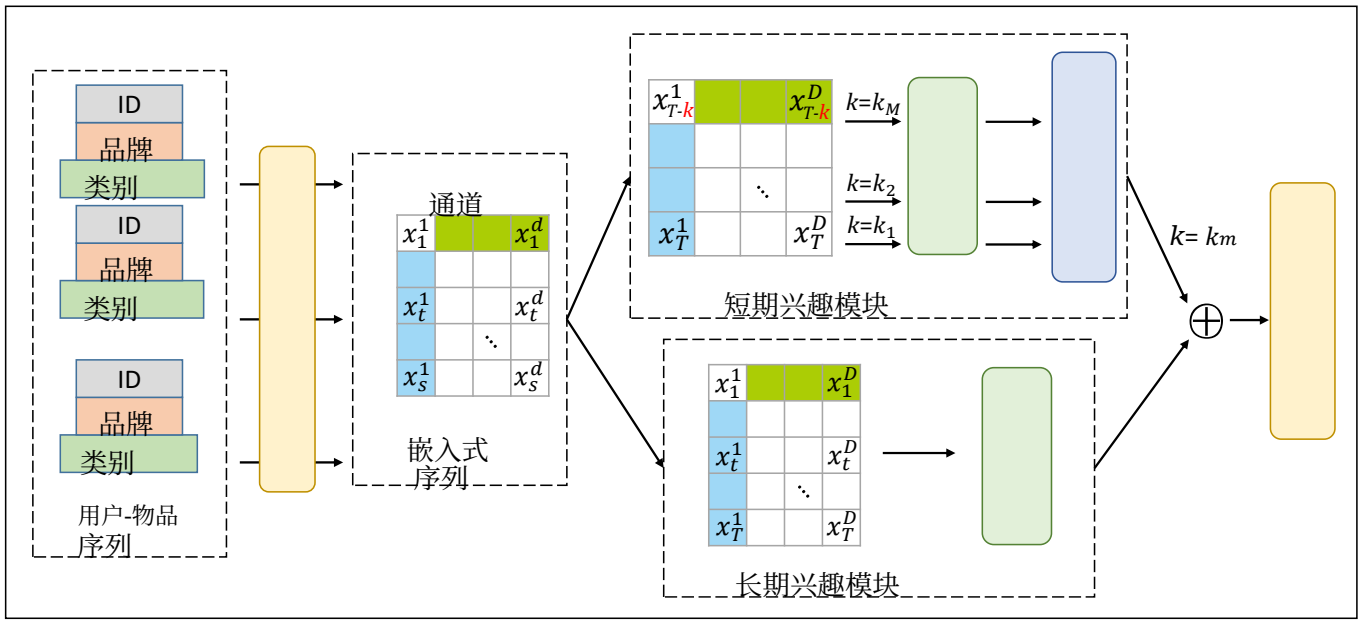

图2：AutoMLP 框架概述。

短期依赖也有重要作用 [19], , 这是序列推荐中的第三方面关键信息。显然，用户在序列推荐中的行为通常同时受到长期兴趣和短期兴趣的影响，这两者从不同角度描述了用户的偏好，如图 1 所示。然而，尽管大多数现有研究擅长捕捉长期兴趣，只有少数研究专门关注相对动态的短期兴趣。

现有的序列推荐方法通常倾向于使用循环神经网络（RNN）[5, 25, 32] 及带有注意力机制的 RNN [14, 33]来建模交互物品之间的序列依赖关系。基于 RNN 的方法能够成功捕捉短期依赖关系，但众所周知，面对长序列时，RNN 容易退化，尽管人们提出了长短期记忆网络（LSTM）和 GRU 等技术来缓解这一问题。此外，近期进展表明，基于 Transformer 的方法的表现大幅优于以往方法 [10, 23, 37]。然而，由于自注意力的特性，这些方法对输入序列的顺序并不敏感，因此必须依赖辅助位置嵌入来学习序列信息。尽管已有研究表明，这种方法有时可能会损害性能 [10, 11]。此外，Transformer 的计算复杂度与输入序列长度呈平方关系，因此处理较长用户交互序列的计算成本不容忽视。再者，由于用户的行为在长期和短期层面上可能存在异质性，即，学到的短期依赖关系可能与长期依赖关系不一致，反之亦然，因此大多数现有工作会分别训练独立模型来捕捉长期和短期兴趣。然而，这些方法通常通过经验做法限制短期序列长度，例如选取最近的固定数量的交互次数，或依据固定的时间间隔 [19, 26, 33]。这些方法显然不够自适应或自动化，因为最佳短期长度会因不同的推荐任务或场景而异。

为了解决上述问题，我们提出了一种简单而高效的方法，称为面向序列推荐的自动化长短期多层感知机（**Automated Long-termShort-term Multi-Layer Perceptron for sequential recommendation**（**AutoMLP**）。AutoMLP 仅由多层感知机模块构成，因此保持了线性时间和空间复杂度。我们设计了一个长期兴趣模块和一个短期兴趣模块，分别用于捕捉长期和短期依赖关系。为了使短期兴趣窗口能够自动适应不同任务，我们利用连续松弛，通过自动机器学习技术将离散序列长度转换为连续且可微的表示，并可通过梯度下降进行优化 [18]。为展示所提方法的有效性，我们在常用基准测试上进行了测试。总而言之，本文的贡献如下：

- 我们提出了一种新的长短期序列推荐模型，该模型仅由多层感知机模块构成。尽管其架构简单且复杂度为线性，但与最先进的方法相比，仍展现出有竞争力的性能。
- 我们设计了一种自动化短期会话长度学习器，用于搜索能够适应给定上下文的局部最优短期兴趣长度，从而增强模型的泛化能力。
- 我们在常用数据集以及来自行业的真实世界私有数据集上进行了大量实验，以测试所提出的方法。

2 框架

本节将详细介绍我们提出的方法 AutoMLP，它能够自适应地选择用户的短期

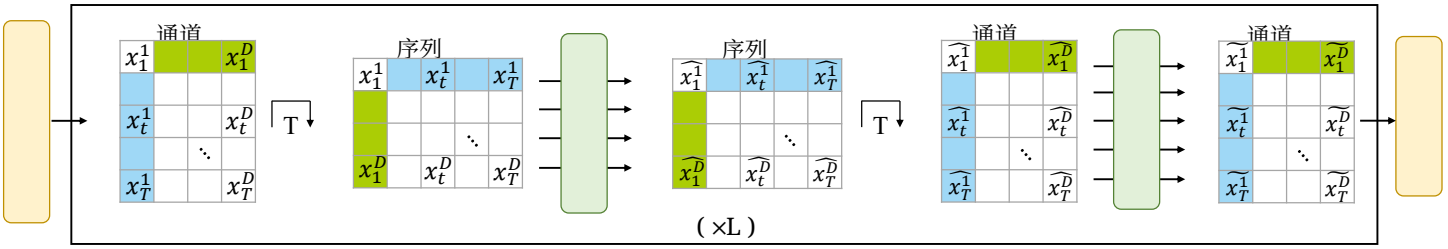

图 3：序列推荐系统多层感知机层（SRSMLP层）的架构。

兴趣长度，并利用多层感知机模块高效捕捉长期和短期兴趣。我们将首先概述 AutoMLP框架，然后详细讨论各个组成部分。

2.1 问题定义

序列推荐的目标是根据用户的历史交互序列，预测用户接下来可能交互的物品。因此，对于用户集合 u 中的每位用户 U，都对应一个从 $S=\{S_1, \ldots\ldots S_u, \ldots\ldots S_U\}$ 中取出的序列，其中 S_u 表示用户 u 的交互序列。在每个序列中，用户交互物品按时间顺序排列，即 $S_u=\{i_1, \ldots\ldots i_t, \ldots\ldots i_T\}$，其中 i_t 是时间戳为 t 时交互的物品，T 是序列长度。于是，序列推荐的目标可形式化定义为：给定用户 u<style id='29'}的历史交互序列 S_u，目标是找到一个函数 f: $S_u \rightarrow i_{T+1}$，其中 i_{T+1} 是要推荐给用户 u 的下一个物品。

2.2 框架概述

现在，我们介绍 AutoMLP 的整体概述。AutoMLP 是一种完全基于 MLP 架构的序列推荐系统，能够针对不同的序列推荐应用自动学习合适的短期用户兴趣长度。AutoMLP 的主体由两个独立的基于 MLP 的网络组成，即长期用户兴趣模块和短期兴趣模块，如图2所示。具体来说，长期用户兴趣模块将整个用户历史行为序列用于预测，因此更侧重于长期序列依赖。

另一方面，短期用户兴趣模块会选取时间 T 之前最近的若干次交互次数，倾向于建模短期序列依赖关系。最近交互次数 k 将由神经架构搜索（NAS）算法 DARTS [18], 确定。该算法利用连续松弛使神经架构搜索空间具有可微性，从而能够通过梯度下降进行优化。最后，两个独立模块的输出将通过一个全连接层融合，以预测用户接下来交互的物品。

2.3 详细架构

本节将详细介绍宏观结构以及我们所提出方法背后的方法论。

2.3.1 嵌入层。 遵循序列推荐系统中 [10, 23, 36], 常用的策略，我们使用查找表将物品ID及其他特征转换为嵌入向量。假设每个物品 i_t 具有 C 个特征，接着我们使用全连接层，将来自不同特征的 C 嵌入向量融合为一个维度预先设定为 D的嵌入向量 $\boldsymbol{x_t} = [x_t^1, \cdots\ x_t^D]$。对于长度为 T的用户交互序列，我们由此得到一个大小为 $T \times D$的嵌入表，即 图2中的嵌入序列。

2.3.2 长期兴趣模块. 现在介绍用于建模长期序列依赖的关键组件，即 长期兴趣模块。如上所述，长期兴趣模块以整个用户交互序列作为输入，并将兴趣编码为用于预测的隐藏表示。编码过程由图 3 中的序列推荐系统多层感知机层（即 *SRSMLP 层* ）完成，该层由两个多层感知机模块组成，分别是序列混合器和通道混合器。SRSMLP 层交替使用这两个多层感知机模块，以捕捉用户交互序列中的序列相关性，以及每个物品嵌入向量中的跨通道相关性。

序列混合器。 图 4 展示了序列混合器的结构示意图。序列混合器旨在学习用户交互序列中的序列信息。序列混合器的输入维度是输入序列的长度，即 T。回想一下，经过嵌入层后，我们得到一个大小为 $T \times D$ 的嵌入表，因此，如图 3 所示，序列混合器作用于转置后的嵌入表，其大小为 $D \times T$。序列混合器执行非线性投影，将输入 $\boldsymbol{x^d}$ 映射为形状相同的输出$\hat{}\boldsymbol{x^d}$，即 $\mathbb{R}^{D\times T} \mapsto \mathbb{R}^{D\times T}$，将序列中的跨物品（序列）信息融合到每个物品的嵌入向量中。具体而言，由于整个序列的输入具有相同的嵌入维度，即 $|[x_1^d, \cdots\ x_T^d]| = T, \forall d \in [1, D]$，我们可以看出，序列混合器能够学习贯穿整个序列的序列相关性，这通常反映了用户长期兴趣的演变。我们将第 l 层序列混合器的输出记为：

$$\hat{\boldsymbol{x}}^{\boldsymbol{d}} = \boldsymbol{x}^{\boldsymbol{d}} + \boldsymbol{W}^{\boldsymbol{2}} g^l(\boldsymbol{W}^{\boldsymbol{1}} LayerNorm(\boldsymbol{x}^{\boldsymbol{d}})) \quad (1)$$

其中 $d = 1, 2, \cdots\ D$。 $\boldsymbol{x}^d$是嵌入表中的第 d个示例。$\hat{}\boldsymbol{x^d}$ 是序列混合器模块的输出， g^l 是第 l层的非线性激活函数，$\boldsymbol{W^1} \in \mathbb{R}^{R_s\times T}$表示序列混合器中第一个全连接层的可学习权重，$\boldsymbol{W^2} \in \mathbb{R}^{T\times R_s}$ 表示

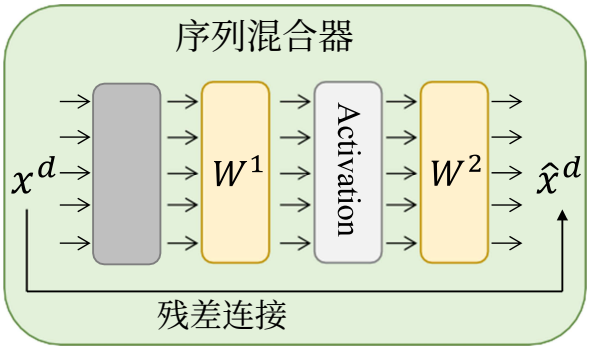

图 4：序列混合器。通道混合器具有类似的架构，只是输入 x^d 和输出 $\hat{x}^d$ 有所不同。

序列混合器中第二个全连接层的可学习权重，R_s 是序列混合器可调节的隐藏层大小。我们采用与MLP混合器 [28]相同的层归一化（$LayerNorm$）和残差连接。

通道混合器。 通道混合器与图 4 中的序列混合器具有相似的宏观结构，二者的关键区别在于用途。通道混合器的目标是学习每个物品嵌入向量内部的相关性。由于嵌入向量的每个维度通常表达不同的语义，综合学习这些表示对于做出有依据的预测至关重要 [13]。此外，经过序列混合器后，序列信息会融合到每个嵌入向量的各个维度中，通道混合器还可以在不同维度之间传递序列信息，从而连贯地学习序列信息的隐藏表示 [28]。因此，通道混合器的输入维度必须与嵌入向量的大小一致，即 D。因此，对于从嵌入层获得的嵌入表，在经过序列混合器处理并编码序列信息后，我们首先将其转置回初始形状，即 $T \times D$。随后，通道混合器将输入 $\boldsymbol{x^t}$ 映射为形状相同的输出 $\hat{\boldsymbol{x}}^{\boldsymbol{t}}$，即 $\mathbb{R}^{T\times D} \mapsto \mathbb{R}^{T\times D}$，将跨通道相关性融合到每个物品的嵌入向量中。我们将第 l 层通道混合器的输出表示为：

$$\hat{\boldsymbol{x}}^{\boldsymbol{t}} = \boldsymbol{x}^{\boldsymbol{t}} + \boldsymbol{W}^{\boldsymbol{4}} g^l(\boldsymbol{W}^{\boldsymbol{3}} LayerNorm(\boldsymbol{x}^{\boldsymbol{t}})) \tag{2}$$

其中，$t = 1, 2, \cdots T$, $\boldsymbol{x^t}$ 是输入，即来自嵌入表的 t^{th} 示例。$\hat{\boldsymbol{x}}^{\boldsymbol{t}}$ 是通道混合器的输出，其中包含跨通道相关性。$\boldsymbol{W^3} \in \mathbb{R}^{R_c \times D}$ 是通道混合器中第一个全连接层的可学习权重，而 $\boldsymbol{W^4} \in \mathbb{R}^{D\times R_c}$ 是第二个全连接层的可学习权重，其中 R_c 表示通道混合器中隐藏层的可调大小。

2.3.3 **短期兴趣模块。** 现在我们介绍短期兴趣模块，用于捕捉用户历史序列中的短期序列依赖关系。短期兴趣模块采用与长期兴趣模块类似的框架，也包含序列混合器和通道混合器。

关键区别在于短期兴趣长度搜索部分

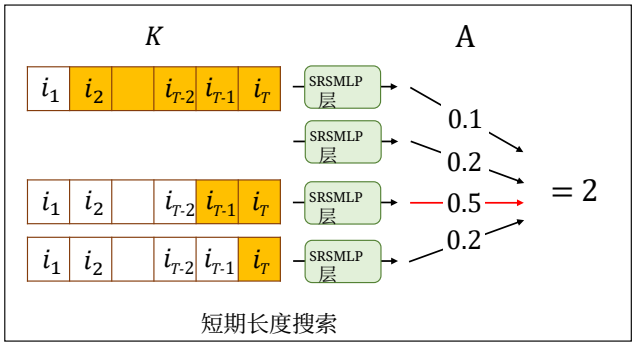

图5：短期会话长度搜索过程。 K **是候选长度集合，由高亮部分表示。**

该部分能够以数据驱动的方式自动选择最佳短期兴趣长度 k。

会话长度搜索。 会话长度搜索部分是一种基于 DARTS [18] 的神经架构搜索（NAS）算法。具体来说，我们首先在最后一个物品 i_T 之前定义一些候选短期长度，以表示用户短期兴趣长度的可能范围，即 也就是 $\boldsymbol{K} = \{k_1, \ldots\ldots, k_m, \ldots\ldots, k_M\}$，如图5所示。然后，我们创建 M 个 SRSMLP，每个对应一个候选 k_m。接下来，如图5所示，我们分别为不同 SRSMLP 的输出分配一组可学习的架构权重 $\boldsymbol{A} = \{\alpha_1, \ldots\ldots, \alpha_m, \ldots\ldots, \alpha_M\}$。之后，我们应用 softmax，将权重转换为连续且可微的近似值：

$$p_m = \frac{\exp(\alpha_m)}{\sum_{m=1}^{M} \exp(\alpha_j)} \tag{3}$$

其中 $m = 1, \cdots M$，且 α_m 是 m^{th} 个候选短期序列对应的可学习权重。根据公式（3），我们现在得到一组连续且可微的权重 $\{p_1, \cdots p_m, \cdots p_M\}$，它们可以自适应地确定短期序列的长度。

讨论。 由于短期兴趣长度的取值与模型性能之间的影响并非单调，因此要确定局部最优值，必须进行穷举搜索。对于较长的用户-物品交互序列而言，由于可能的候选值更多，这种搜索的计算开销极大。因此，这种方法的主要优势在于，无须枚举每一种可能的模型架构并反复训练，就能学习到局部最优的用户短期兴趣长度，从而使选择短期兴趣长度的决策过程更加高效且自适应。

2.3.4 **输出层。** 找到最优的短期序列长度后，我们可以获得长期兴趣模块和最优短期兴趣模块的输出。然后，我们使用一个全连接层来学习它们的联合表示，如下所示：

$$\boldsymbol{h}_T = \boldsymbol{W^o} LayerNorm(\boldsymbol{x}_T^s; \boldsymbol{x}_T^l)) + \boldsymbol{b^o} \tag{4}$$

其中，h_T 是时间步 T 的最终隐藏表示，x_T^s 和 x_T^l 分别是时间步 T 的短期兴趣模块和长期兴趣模块的输出。$\boldsymbol{W^o}$ 是可学习的

权重，用于将组合后的 x_T^s 和 x_T^l 投影到较低维度的表示，即 $\boldsymbol{W^o} \in \mathbb{R}^{D\times 2D}$；$\boldsymbol{b^o} \in \mathbb{R}^D$ 是可学习的偏置向量。最后，h_T 将用于预测用户的下一次交互，我们将在第 2.4.2 节的模型推理部分进一步介绍。

2.4 训练与推理

按照序列推荐中的常见设置，我们的训练损失由交叉熵损失函数定义：

$$\mathcal{L} = -\sum_{S_u \in S}\sum_{t\in[1,\ldots,T]}[log(\sigma(r_{i_t,t})) + \sum_{j\notin S_n} log(1-\sigma(r_{i_j,t}))] \quad (5)$$

其中 σ 表示 sigmoid 函数。$r_{i_t,\ t}$ 是模型对真实物品 i_t 的预测相似度，而 $r_{i_j,\ t}$ 是模型在时间步 t 对采样物品的预测相似度，其中 j 是负采样物品。S 是包含所有用户交互序列的超集。

2.4.1 训练。 训练过程包含两个阶段。第一阶段是搜索阶段，旨在找到代表合适短期长度的局部最优 $\boldsymbol{A}^*$。第二阶段是重新训练阶段，在找到 $\boldsymbol{A}^*$ 后，我们使用最优短期长度重新训练 AutoMLP框架。

搜索阶段。上一节阐述了连续松弛如何使搜索空间 $\boldsymbol{K}$ 可微，从而能够通过反向传播进行优化。现在，我们的目标是联合学习架构权重 $\boldsymbol{A} = \{\alpha_m\}$，以及 AutoMLP $\boldsymbol{W}$ 的所有其他可学习参数。请注意，每个 p_m 都与每个 α_m 相对应，即，学习 $\{p_m\}$ 与学习 $\{\alpha_m\}$ 相同。虽然 $\boldsymbol{A}$ 是 AutoMLP 可学习参数的子集，但已有研究表明，在训练过程中同时更新 $\boldsymbol{W}$ 和 $\boldsymbol{A}$ 会导致过拟合问题，因为它们彼此高度依赖 [39, 40]。因此，按照常见做法，我们将使用训练数据集优化 $\boldsymbol{W}$，同时使用验证数据集优化 $\boldsymbol{A}$。具体而言，我们将其表述为一个双层优化问题 [2], ，其中 $\boldsymbol{A}$ 是上层变量，$\boldsymbol{W}$ 是下层变量 [18]。形式化地，我们将其表示为：

$$\begin{aligned} &\min_{\boldsymbol{A}} \mathcal{L}_{val}(\boldsymbol{W}^*(\boldsymbol{A}), \boldsymbol{A}) \\ &s.t.\ \boldsymbol{W}^*(\boldsymbol{A}) = \arg\min_{\boldsymbol{W}} \mathcal{L}_{train}(\boldsymbol{W}, \boldsymbol{A}) \end{aligned} \quad (6)$$

此外，值得注意的是，求解公式（6）的内部优化（下层变量优化）通常计算开销很大，因为每当我们要优化 $\boldsymbol{A}$ 时，都必须先训练 $\boldsymbol{W}$ 直至收敛。作为替代，我们采用单步近似：

$$\boldsymbol{W}^*(\boldsymbol{A}) \approx \boldsymbol{W} - \xi\nabla_{\boldsymbol{W}}\mathcal{L}_{train}(\boldsymbol{W}, \boldsymbol{A}) \quad (7)$$

其中 ξ 是学习率。采用这种近似方法的动机是通过单步训练而非充分训练来近似 $\boldsymbol{W}^*(\boldsymbol{A})$。搜索阶段的详细训练算法如算法 1所示。

重新训练阶段。 在搜索阶段，我们将所有候选短期序列纳入 AutoMLP 框架，其中次优短期序列可能会对模型性能产生负面影响。因此，我们需要仅使用最优短期序列重新训练 AutoMLP。

算法1 搜索阶段 AutoMLP 的优化

输入：嵌入表 $\boldsymbol{x}$ 和真实交互序列 $\boldsymbol{S}$

输出：训练良好的参数 $\boldsymbol{W}^*$ 和 $\boldsymbol{A}$

当未收敛**时执行**
  抽取一小批验证数据样本
  通过公式（7）估算 $\boldsymbol{W}^*(\boldsymbol{A})$的近似值
  通过梯度下降更新 $\boldsymbol{V}$，以最小化 $\nabla_{\boldsymbol{A}} \mathrm{L}_{val}(\boldsymbol{W}^*(\boldsymbol{A})\ \boldsymbol{A})$
  采样一个由训练数据样本组成的小批量
  通过梯度下降更新 $\boldsymbol{W}$，以最小化 $\nabla_{\boldsymbol{W}} \mathrm{L}_{train}(\boldsymbol{W},\ \boldsymbol{A})$
当循环结束

具体而言，搜索阶段优化完成后，我们得到训练良好的 $\boldsymbol{A}^*$。我们选择 α_m 最高的最优短期长度，并舍弃所有其他候选长度。随后，我们遵循深度推荐系统的常见训练范式，根据长期兴趣模块和最优短期兴趣模块的输出重新训练 AutoMLP 框架，即 通过梯度下降优化损失 $\min_{\boldsymbol{W}} \mathrm{L}_{train}(\boldsymbol{W})$。

2.4.2 推理。我们提出的方法采用序列推荐系统中常用的推理流程 [10, 23,36, 42], ，即计算模型输出与所有候选物品之间的余弦相似度。具体而言，一旦我们从输出层获得隐藏表示 h_T，并将其作为整个序列的最终隐藏表示，便计算 h_T 与所有候选物品的物品嵌入之间的点积 I：

$$p_\iota = \mathrm{softmax}(h_T \cdot E_\iota^T) \quad (8)$$

其中 $\iota = 1,\ 2,\ \cdots\ |I|$，$|I|$ 为候选物品的总数。E_ι 是 i_ι 的物品嵌入，p_ι 是 i_ι 作为下一次可能交互的预测概率。因此，p_ι 最高的候选物品即为预测的下一次交互 i_{T+1}。

2.5 复杂度分析

本节将介绍 AutoMLP 的一项重要优势，即线性计算复杂度。AutoMLP 的复杂度主要由两个多层感知机模块、序列混合器和通道混合器决定。由于具有一个隐藏层的多层感知机的时间复杂度为 $O(input\ units \times hidden\ units \times output\ units)$，因此我们可以得知序列混合器的复杂度为

$$O\ T \times R_s + (\ )。$$

$R_s \times T$

由于此处的决定变量是最大序列长度 T，且 R_s 是独立于 T 的常数，因此可将其改写为 $O(T)$。类似地，通道混合器的复杂度为 $O(D)$。此外，短期模块除序列长度外，结构完全相同，因此短期序列混合器的复杂度为 $O(k)$，短期通道混合器的复杂度也为 $O($ $D)$。因此，AutoMLP 的复杂度为 $O(T + D + k)$，并且关于三个决定变量呈线性增长。

3 实验

在本节中，我们将在推荐系统领域两个常用的基准数据集上，通过大量实验评估所提出的方法 AutoMLP。

表 1：数据集统计信息。

Data	MovieLens	Amazon Beauty
# 交互次数	1,000,209	2,023,070
物品数	3,952	249,274
用户数	6,040	1,210,271

3.1 数据集

MovieLens[1]：MovieLens 是推荐系统领域常用的基准数据集，包含用户对各类电影的评分历史及相应的时间戳。本研究选择 MovieLens-1M 进行实验，该数据集包含超过一百万次交互次数，详细统计信息见表 1。

Amazon Beauty[2]：该数据集包含从亚马逊 [20]，抓取的各种物品的评论和评分；数据集分为多个类别，本文将使用“Beauty”类别，该类别的统计数据见表 1。

为了在更贴近现实的场景中对长期依赖关系进行建模，我们从 MovieLens 和 Amazon Beauty 中筛除了交互次数少于 10 次的用户。关于这些数据集的更多统计信息见表 1。

3.2 实验设置

我们遵循序列推荐中常见的设置，即下一物品预测（留一评估）。在此设置下，我们将用户交互序列中最近的交互作为测试集，倒数第二次交互作为验证集，其余此前的交互作为训练集 [10, 15, 23, 36, 42]。评估时，我们还会为每个真实物品配对 100 个负样本，并根据其流行度进行采样 [8, 23]。

3.3 评估指标

我们采用三种最常用的评估指标：命中率（HR）、归一化折损累计增益（NDCG）和平均倒数排名（MRR）。HR 表示真实物品出现在前 N 个推荐结果中的准确率；NDCG 是一种排序损失，用于衡量真实物品在前 N 个推荐结果中的位置；MRR 则是前 N 个推荐结果中真实物品排名的倒数。

3.4 基线方法

为展示我们所提出方法的有效性，我们将其与几种热门的序列推荐模型进行比较，其中一些模型取得了最先进的性能。根据其设计动机和采用的技术，这些模型可分为三组：

通用序列推荐系统 指若干重要类别中具有代表性的序列推荐模型，即基于马尔可夫链、基于循环神经网络和基于 Transformer 的模型。

• **FPMC** 个性化马尔可夫链分解（FPMC） [22]代表了序列推荐系统中的早期研究，该方法使用马尔可夫链对物品之间的序列依赖关系进行建模。

• **GRU4Rec**Hidasi 等人提出使用循环神经网络进行序列推荐，并通过门控循环单元（GRU）和逐点排序损失 [5]改进了基础循环神经网络。

• **BERT4Rec**Sun 等人采用双向自注意力来改进基于 Transformer 的序列推荐 [23]，，且多项研究表明，BERT4Rec 始终优于 SASRec 等传统自注意力方法 [13, 23, 42]

长短期序列推荐系统着重于对短期序列依赖关系进行建模，同时也保留长期序列依赖关系：

• **NextItNet** Yuan 等人指出了 Caser 的一些现有限制 [26]，，尤其是其在长期序列依赖关系建模方面的不足，并提出了一种生成模型，该模型无需池化操作即可扩大感受野，从而改进基于卷积神经网络的序列推荐系统 [35]。

特征级序列推荐系统是指能够学习物品特征内部信息，以辅助下一物品预测的序列推荐模型。

• **GRU4Rec**$^+$ 是 GRU4Rec 的改进版本，它利用物品特征在特征丰富的场景中实现更准确的预测，并通过并行循环神经网络提高效率 [6]。

• **FDSA** 面向序列推荐的特征级深度自注意力网络（FDSA） [36] 利用物品特征进行序列推荐，并被认为在有监督方法中表现达到最先进水平 [42]。

3.5 实现细节

基线方法和 AutoMLP 均基于 RecBole 框架实现 [38]，。RecBole 是一个用于推荐系统的开源库，可提供无偏的环境来评估我们所提方法的性能。

对于基线方法，我们采用原始论文中建议的超参数。对于论文中未指定的超参数，我们使用网格搜索进行调优。搜索策略是选择在验证集上取得最佳性能的超参数。我们采用早停策略，即，如果接下来的 10 个训练轮次中验证性能没有提升，就停止训练，并使用当前验证性能最佳时的模型。对于 BERT4Rec 等较大型模型，在处理大型数据集时，其空间复杂度可能过高，无法装入 GPU 显存。因此，我们定义了一个“空间高效”设置，即在 32 GB 的 GPU 显存容量限制内搜索最佳超参数。所有实现均使用 Adam [12] 作为优化器，学习率设为 $1e-3$，
β_1 0 9 β_2 0 999
as . , as . .

3.6 整体性能

表2展示了我们提出的方法与多种流行基线方法的整体比较。

根据结果，我们可以得出以下观察：

[1]https://grouplens.org/datasets/movielens/1m/ [2] http://jmcauley.ucsd.edu/data/amazon/

表2：总体性能比较。最佳性能以粗体标示，次佳性能以下划线标示

方法	MovieLens			Beauty			参数
指标	MRR@10	NDCG@10	HR@10	MRR@10	NDCG@10	HR@10	
FPMC	0.2453	0.3088	0.5156	0.0991	0.1251	0.2098	100 M
GRU4Rec	0.3893	0.4553	0.6666	0.1162	0.1435	0.2324	33.5 M
BERT4Rec	0.3535	0.4289	0.6695	0.0907	0.1198	0.2154	17 M
NextItNet	0.2085	0.2642	0.4455	0.1087	0.1393	0.2385	16.8 M
GRU4Rec+	0.3736	0.4412	0.6578	0.1325	0.1638	0.2657	34.1 M
FDSA	0.3725	0.4409	0.6594	0.1305	0.1595	0.2536	34.3 M
AutoMLP	**0.3912***	**0.4593***	**0.6767***	**0.1438***	**0.1754***	**0.2779***	16.8 M

“*”表示统计上显著的改进（即，与原始模型进行 $p < 0.05$双侧 t 检验）。“Param”指 Beauty 数据集上的可训练模型参数数量，M = 表示百万。

• 当物品特征可用时，特征级模型通常比无法处理特征的模型表现更好，这表明物品特征在序列推荐中的重要性。

• AutoMLP 在所有数据集上都保持了具有竞争力的性能。具体而言，AutoMLP 显著优于现有的长/短期序列推荐系统，例如 NextItNet，表明它不仅是高效的选择，也是准确性方面的有力替代方案。

• 最后，我们可以观察到，AutoMLP 所消耗的可训练模型参数最少，这表明其在现实世界序列推荐系统中具有卓越的空间效率。

总而言之，在公开数据集上，AutoMLP 的性能可与流行的基线方法相媲美，且空间复杂度更低，这验证了其有效性。

3.7 效率分析

本节将研究 AutoMLP 的时间/空间效率，这些指标对于在工业推荐系统中部署推荐模型至关重要。

作为一项基本动机，我们认为，通过自动搜索短期兴趣长度，AutoMLP 比穷举搜索最优短期兴趣长度更高效。在图 6 中，我们将 AutoMLP 与 AutoMLP-s 进行比较，后者与 AutoMLP 的架构相同，但采用了不同的短期兴趣长度搜索算法。具体而言，AutoMLP-s 会穷举搜索能够取得最佳性能的短期兴趣长度……

性能与现有方法相当 [26, 33]。在图 6(a) 中，我们比较了候选搜索空间大小（即，候选短期长度）为 5 时 AutoMLP 和 AutoMLP-s 的搜索时间。在图 6(b) 中，我们展示了随着搜索空间增大，AutoMLP 的增幅比率要小得多，这表明 AutoMLP 具有更好的可扩展性。

此外，从图 7 中我们还可以观察到，与最先进的基于 Transformer 的方法相比，AutoMLP 在时间开销和内存消耗方面具有更高的训练效率。图 7(a) 展示了训练 AutoMLP、FDSA 和 BERT4Rec 直至收敛所需的总时间（秒）。我们可以观察到，由于 BERT4Rec 实现了完形填空目标和双向自注意力等复杂训练技术， [23], 其

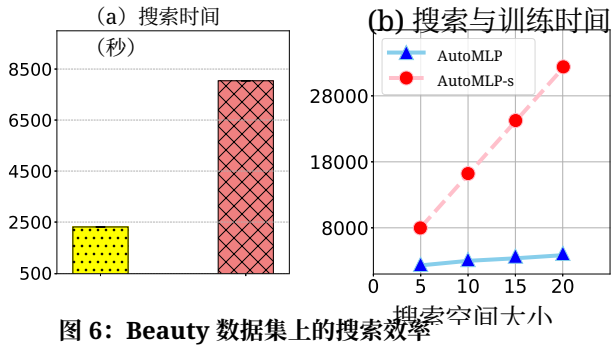

图 6：Beauty 数据集上的搜索效率

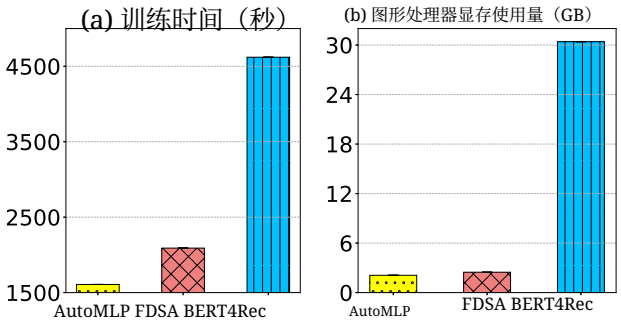

图 7：Beauty 数据集上的训练效率

训练效率相对较低。FDSA 通过结合自注意力和普通注意力，实现了更均衡的训练效率，但其训练成本仍高于 AutoMLP。

3.8 超参数分析

本节介绍 AutoMLP 中的关键超参数如何影响模型性能。与基于 Transformer 的方法 [10, 23, 36]和基于 CNN 的方法 [26, 35], 不同，后者拥有更多超参数和更大的搜索空间，我们提出的方法只有少数几个会显著影响性能的关键超参数。我们研究层数 L 和嵌入维度 D 如何影响 AutoMLP 的推荐性能。

如图 8 (a) 所示，我们可以观察到，最佳层数为 8。换句话说，较少的层（例如 $L = 4$）可能

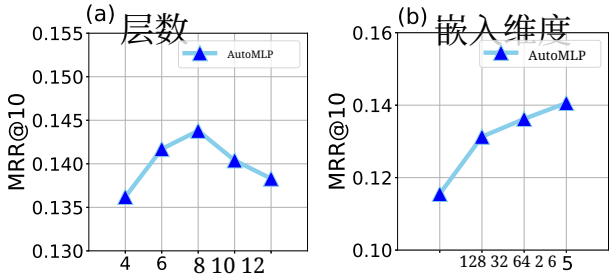

图 8：超参数对 Beauty 数据集的影响导致模型表征能力退化，而更多的层（例如 $L = 12$）可能会引发过拟合问题。从图 8 (b) 可以总结出，较小的嵌入维度会降低模型性能；随着嵌入维度的增大，AutoMLP 具备了更强的表征能力，能够更好地学习用户-物品交互表征。

3.9 消融研究

在前面的小节中，我们展示了 AutoMLP 的有效性和效率。关于这种有效性，一个直观的问题是：我们如何为所提出模型的各个模块分配贡献度？为了验证模型各部分的重要性，我们设计了简化的替代架构，每种架构都移除一个关键组件。性能结果见表3。其中，w/o Sequence Mixer 表示从 AutoMLP 中移除序列混合器，w/o Channel Mixer 表示从 AutoMLP 中移除通道混合器。

如表3所示，不使用序列混合器时，AutoMLP 的性能下降最为显著。正如我们在框架部分所述，序列混合器旨在捕捉输入序列的序列相关性；若没有它，模型只会假设输入序列彼此独立并据此进行预测。性能的大幅下降表明，序列信息在我们的推荐任务中至关重要，而序列混合器能够有效捕捉此类信息。

仅移除通道混合器时，我们也观察到性能有所下降，但与移除序列混合器相比，下降幅度较小。性能下降表明，在每一层之后交换跨通道信息，也能通过更好地学习物品不同隐式属性之间的联合相关性，提升预测质量。

4 相关工作

本节简要概述与我们研究相关的几个方向，包括序列推荐和MLP混合器。

4.1 序列推荐

序列推荐在推荐系统中发挥着至关重要的作用。它旨在根据用户与物品之间的序列交互，描绘用户连续变化的偏好。循环神经网络在建模序列依赖关系方面具有天然优势，因此在基于深度学习的序列推荐中占据主导地位。GRU4Rec [5]在基于会话的推荐中采用门控循环单元（GRU）。其改进版本 [4] 通过采用排序损失函数和改进的采样策略，进一步提升了性能。然而，循环神经网络存在

表3：Beauty 数据集上的消融研究对比

模型	MRR@10	NDCG@10	HR@10
不含序列混合器	0.1132	0.1315	0.2169
不含通道混合器	0.1408	0.1720	0.2700
AutoMLP	**0.1438**	**0.1754**	**0.2779**

较高的训练成本，尤其是在对长序列进行建模时。此外，由于梯度消失问题 [31]，这两种基于循环神经网络的模型都无法很好地捕捉长期依赖。

自从 [1],注意力成功应用以来，它已广泛应用于序列推荐。NARM [14] 通过编码器-解码器框架捕捉序列用户-物品交互以及用户在当前会话中的主要意图。Yu 等人 [34] 提出通过联合学习个体级和联合级物品交互来保持用户偏好。

近期，得益于相关领域的进展 [30], , Transformer 的表现已优于循环神经网络结构。为弥补基于循环神经网络的模型的不足，SASRec [10]提出使用注意力机制对长期序列依赖关系进行建模。Bert4Rec [23] 采用深层双向自注意力来学习用户偏好。FDSA [36] 通过以不同权重整合异构特征来建模转移模式。此外，基于 Transformer 的模型在没有启发式处理的情况下不会考虑输入序列的顺序[11]。

4.2 MLP混合器

近期关于MLP架构的进展 [17, 27–29] 表明，只需对设计进行简单调整，多层感知机便可成为基于 Transformer 的模型的有力替代方案，在与最先进的方法相比取得同等竞争力表现的同时，还能保持较低的时间和空间复杂度。其中，MLP-Mixer [28]通常被视为最具代表性的工作。它使用独立的多层感知机模块从图像中学习位置内相关性和跨位置相关性；在大型数据集上，MLP-Mixer展现出良好的可扩展性，性能可与基于 Transformer 和基于卷积神经网络（CNN）的方法相媲美。

值得注意的是，我们提出的方法并非首个尝试在推荐系统中采用纯多层感知机架构的工作。MOI-Mixer [13] 首次探索了将多层感知机架构用作基于 Transformer 的方法的替代方案的可能性。随后，MLP4Rec [16] 提出了一种三向信息融合方案，以便在丰富特征场景下连贯地捕捉不同属性层级上的高阶交互。

5 结论

本文提出了一种名为 AutoMLP 的长短期序列推荐系统。AutoMLP 仅采用多层感知机架构，在开源基准数据集和工业应用中的真实世界数据集上，表现均可与最先进的方法相媲美。此外，我们设计了一种自动短期兴趣长度搜索算法，能够高效地学习最优的短期兴趣长度。结合多层感知机架构的线性复杂度，与现有方法相比，我们的方法效率更高，且具有更大的提升潜力。

致谢

本研究部分得到了 APRC - CityU 新研究计划（编号：9610565，香港城市大学新任教师启动基金）、SIRG - City U 战略跨学科研究基金（编号：7020046、7020074）、HKIDS 青年学者研究资助（编号：9360163）、华为创新研究计划以及蚂蚁集团（CCF-蚂蚁研究基金）的支持。

参考文献

[1] Dzmitry Bahdanau、Kyunghyun Cho 和 Yoshua Bengio。2014年。通过联合学习对齐与翻译实现神经机器翻译。arXiv 预印本 arXiv:1409.0473（2014年）。[2] Benoît Colson、Patrice Marcotte 和 Gilles Savard。2007年。双层优化概述。《运筹学年鉴》153（2007年），235–256。[3] Jie Feng、Yong Li、Chao Zhang、Funing Sun、Fanchao Meng、Ang Guo 和 Depeng Jin。2018年。Deepmove：利用注意力循环网络预测人类移动。在《2018年万维网会议论文集》中。1459–1468。[4] Balázs Hidasi 和 Alexandros Karatzoglou。2018年。用于基于会话的推荐、具有 top-k 增益的循环神经网络。在《第27届 ACM 信息与知识管理国际会议论文集》中。843–852。[5] Balázs Hidasi、Alexandros Karatzoglou、Linas Baltrunas 和 Domonkos Tikk。2015年。基于会话的循环神经网络推荐。arXiv 预印本 arXiv:1511.06939（2015年）。[6] Balázs Hidasi、Massimo Quadrana、Alexandros Karatzoglou 和 Domonkos Tikk。2016年。用于富特征基于会话的推荐的并行循环神经网络架构。在《第10届 ACM 推荐系统会议论文集》中。241–248。[7] Linmei Hu、Chen Li、Chuan Shi、Cheng Yang 和 Chao Shao。2020年。结合长期与短期兴趣建模的图神经网络新闻推荐。《Information Processing & Management》57卷2期（2020年），102142。[8] Jin Huang、Wayne Xin Zhao、Hongjian Dou、Ji-Rong Wen 和 Edward Y Chang。2018年。利用知识增强记忆网络改进序列推荐。在《第41届 ACM SIGIR 信息检索研究与开发国际会议论文集》中。505–514。[9] Md Ashraful Islam、Mir Mahathir Mohammad、Sarkar Snigdha Sarathi Das 和 Mohammed Eunus Ali。2022年。基于深度学习的兴趣点（POI）推荐综述。《Neurocomputing》472（2022年），306–325。[10] Wang-Cheng Kang 和 Julian McAuley。2018年。自注意力序列推荐。在《2018年 IEEE 国际数据挖掘会议（ICDM）》中。IEEE，197–206。[11] Guolin Ke、Di He 和 Tie-Yan Liu。2020年。重新思考语言预训练中的位置编码。arXiv 预印本 arXiv:2006.15595（2020年）。[12] Diederik P Kingma 和 Jimmy Ba。2014年。Adam：一种随机优化方法。arXiv 预印本 arXiv:1412.6980（2014年）。[13] Hojoon Lee、Dongyoon Hwang、Sunghwan Hong、Changyeon Kim、Seungryong Kim 和 Jaegul Choo。2021年。MOI-Mixer：通过多阶交互改进多层感知机混合器，用于序列推荐。arXiv 预印本 arXiv:2108.07505（2021年）。[14] Jing Li、Pengjie Ren、Zhumin Chen、Zhaochun Ren、Tao Lian 和 Jun Ma。2017年。神经注意力会话推荐。在《2017年 ACM 信息与知识管理会议论文集》中。1419–1428。[15] Jiacheng Li、Yujie Wang 和 Julian McAuley。2020年。用于序列推荐的时间间隔感知自注意力。在《第13届网络搜索与数据挖掘国际会议论文集》中。322–330。[16] Muyang Li、Xiangyu Zhao、Chuan Lyu、Minghao Zhao、Runze Wu 和 Ruocheng Guo。2022年。MLP4Rec：用于序列推荐的纯多层感知机架构。在第31届国际人工智能联合会议和第25届欧洲人工智能会议（IJCAI-ECAI 2022）上。国际人工智能联合会议，2138–2144。[17] Hanxiao Liu、Zihang Dai、David So 和 Quoc V Le。2021年。关注多层感知机。《神经信息处理系统进展》34（2021年），9204–9215。[18] Hanxiao Liu、Karen Simonyan 和 Yiming Yang。2018年。DARTS：可微分架构搜索。arXiv 预印本 arXiv:1806.09055（2018年）。[19] Chen Ma、Peng Kang 和 Xue Liu。2019年。用于序列推荐的分层门控网络。在《第25届 ACM SIGKDD 知识发现与数据挖掘国际会议论文集》中。825–833。[20] Julian McAuley、Christopher Targett、Qinfeng Shi 和 Anton Van Den Hengel。2015年。基于图像的风格与替代品推荐。在《第38届 ACM SIGIR 信息检索研究与开发国际会议论文集》中。43–52。[21] Massimo Quadrana、Paolo Cremonesi 和 Dietmar Jannach。2018年。序列感知推荐系统。《ACM Computing Surveys》（CSUR）51卷4期（2018年），1–36。[22] Steffen Rendle、Christoph Freudenthaler 和 Lars Schmidt-Thieme。2010年。分解个性化马尔可夫链以实现下一篮推荐。在《第19届万维网国际会议论文集》中。811–820。[23] Fei Sun、Jun Liu、Jian Wu、Changhua Pei、Xiao Lin、Wenwu Ou 和 Peng Jiang。2019年。BERT4Rec：基于 Transformer 双向编码器表示的序列推荐。在《第28届 ACM 信息与知识管理国际会议论文集》中。1441–1450。[24] Ke Sun、Tieyun Qian、Tong Chen、Yile Liang、Quoc Viet Hung Nguyen 和 Hongzhi Yin。2020年。下一步去哪里：为兴趣点推荐建模长期与短期用户偏好。在《AAAI人工智能会议论文集》第34卷中。214–221。[25] Yong Kiam Tan、Xinxing Xu 和 Yong Liu。2016年。改进的基于会话推荐的循环神经网络。在《第1届推荐系统深度学习研讨会论文集》中。17–22。[26] Jiaxi Tang 和 Ke Wang。2018年。通过卷积序列嵌入实现个性化 top-n 序列推荐。在《第11届 ACM 网络搜索与数据挖掘国际会议论文集》中。565–573。[27] Yuki Tatsunami 和 Masato Taki。2021年。RaftMLP：基于多层感知机的模型是否梦想着在计算机视觉领域胜出？arXiv 预印本 arXiv:2108.04384 3（2021年）。[28] Ilya O Tolstikhin、Neil Houlsby、Alexander Kolesnikov、Lucas Beyer、Xiaohua Zhai、Thomas Unterthiner、Jessica Yung、Andreas Steiner、Daniel Keysers、Jakob Uszkoreit 等。2021年。MLP混合器：用于视觉的全多层感知机架构。《神经信息处理系统进展》34（2021年），24261–24272。[29] Hugo Touvron、Piotr Bojanowski、Mathilde Caron、Matthieu Cord、Alaaeldin El-Nouby、Edouard Grave、Gautier Izacard、Armand Joulin、Gabriel Synnaeve、Jakob Verbeek 等。2022年。ResMLP：采用数据高效训练进行图像分类的前馈网络。《IEEE模式分析与机器智能汇刊》（2022年）。[30] Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Łukasz Kaiser 和 Illia Polosukhin。2017年。Attention Is All You Need。《神经信息处理系统进展》30（2017年）。[31] Shoujin Wang、Liang Hu、Yan Wang、Longbing Cao、Quan Z Sheng 和 Mehmet Orgun。2019年。序列推荐系统：挑战、进展与前景。arXiv 预印本 arXiv:2001.04830（2019年）。[32] Chao-Yuan Wu、Amr Ahmed、Alex Beutel、Alexander J Smola 和 How Jing。2017年。循环推荐网络。在《第10届 ACM 网络搜索与数据挖掘国际会议论文集》中。495–503。[33] Haochao Ying、Fuzhen Zhuang、Fuzheng Zhang、Yanchi Liu、Guandong Xu、Xing Xie、Hui Xiong 和 Jian Wu。2018年。基于分层注意力网络的序列推荐系统。在 IJCAI 国际人工智能联合会议上。[34] Lu Yu、Chuxu Zhang、Shangsong Liang 和 Xiangliang Zhang。2019年。用于序列推荐的多阶注意力排序模型。在《AAAI人工智能会议论文集》第33卷中。5709–5716。[35] Fajie Yuan、Alexandros Karatzoglou、Ioannis Arapakis、Joemon M Jose 和 Xiangnan He。2019年。一种用于下一物品推荐的简单卷积生成网络。在《第12届 ACM 网络搜索与数据挖掘国际会议论文集》中。582–590。[36] Tingting Zhang、Pengpeng Zhao、Yanchi Liu、Victor S Sheng、Jiajie Xu、Deqing Wang、Guanfeng Liu、Xiaofang Zhou 等。2019年。用于序列推荐的特征级深层自注意力网络。在 IJCAI 上。4320–4326。[37] Kesen Zhao、Xiangyu Zhao、Zijian Zhang 和 Muyang Li。2022年。MAE4Rec：用于序列推荐的节省存储的变换器。在《第31届 ACM 信息与知识管理国际会议论文集》中。2681–2690。[38] Wayne Xin Zhao、Shanlei Mu、Yupeng Hou、Zihan Lin、Yushuo Chen、Xingyu Pan、Kaiyuan Li、Yujie Lu、Hui Wang、Changxin Tian 等。2021年。RecBole：面向推荐算法的统一、全面且高效的框架。在《第30届 ACM 信息与知识管理国际会议论文集》中。4653–4664。[39] Xiangyu Zhao、Haochen Liu、Wenqi Fan、Hui Liu、Jiliang Tang 和 Chong Wang。2021年。AutoLoss：推荐中的自动损失函数搜索。在《第27届 ACM SIGKDD 知识发现与数据挖掘会议论文集》中。3959–3967。[40] Xiangyu Zhao、Haochen Liu、Hui Liu、Jiliang Tang、Weiwei Guo、Jun Shi、Sida Wang、Huiji Gao 和 Bo Long。2021年。AutoDim：推荐系统中的字段感知嵌入维度搜索。在《2021年万维网会议论文集》中。3015–3022。[41] Guorui Zhou、Na Mou、Ying Fan、Qi Pi、Weijie Bian、Chang Zhou、Xiaoqiang Zhu 和 Kun Gai。2019年。用于点击率预测的深度兴趣演化网络。在《AAAI人工智能会议论文集》第33卷中。5941–5948。[42] Kun Zhou、Hui Wang、Wayne Xin Zhao、Yutao Zhu、Sirui Wang、Fuzheng Zhang、Zhongyuan Wang 和 Ji-Rong Wen。2020年。S3-Rec：通过互信息最大化进行序列推荐的自监督学习。在《第29届 ACM 信息与知识管理国际会议论文集》中。1893–1902。