

# 时间上可持续的多兴趣偏好推荐系统<sup>①</sup>

尹 伟, 林钰斌, 张 威

(武汉科技大学 计算机科学与技术学院 智能信息处理与实时工业系统湖北省重点实验室, 武汉 430081)  
通信作者: 林钰斌, E-mail: 202113704125@wust.edu.cn



**摘 要:** 近年来, 推荐系统已经成为数据分析与挖掘、信息检索领域方向的热点. 然而, 部分推荐系统应用于用户多兴趣偏好还存在着问题: 首先用户的兴趣不唯一, 且对多个兴趣的偏好不等同; 其次用户当前的兴趣是否会在未来时间中持续. 基于此, 本文通过利用用户产生多兴趣并捕获其个性化需求的可持续性, 提出了一种 MIES 算法模型, 该模型能够很好捕捉到用户潜在的多种兴趣, 并注重用户兴趣的可持续, 提高了推荐的质量. 通过对比实验表明该模型有效解决了推荐系统对用户多维兴趣的捕捉和个性化兴趣的可持续性.

**关键词:** 数据分析与挖掘; 多兴趣偏好; 个性化兴趣; 可持续性; 推荐系统

引用格式: 尹伟, 林钰斌, 张威. 时间上可持续的多兴趣偏好推荐系统. 计算机系统应用, 2023, 32(11): 140-148. <http://www.c-s-a.org.cn/1003-3254/9295.html>

## Time-sustainable Multi-interest Preference Recommendation System

YIN Yi, LIN Yu-Bin, ZHANG Wei

(Hubei Provincial Key Laboratory of Intelligent Information Processing and Real-time Industrial System, College of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430081, China)

**Abstract:** In recent years, the recommendation system has become a hot spot in the field of data analysis and mining, as well as information retrieval. However, there are still problems in some recommendation systems serving the multi-interest preferences of users. Firstly, the users' interests are not single, and the preference for multiple interests is not equal. Secondly, it is not sure whether the users' current interests will continue in the future. Therefore, this study proposes a MIES algorithm model by utilizing the items that users participate in to generate multiple interests and capture the sustainability of their personalized interests. The model effectively captures users' diverse latent interests while emphasizing the sustainability of their interests, thus improving the quality of recommendations. Comparative experiments demonstrate that the model effectively addresses the challenges of capturing users' multidimensional interests in recommendation systems and ensuring the sustainability of personalized interests.

**Key words:** data analysis and mining; multi-interest preference; personalized interest; sustainability; recommendation system

近年来, 互联网尤其是在移动互联应用发展迅猛, 随之产生的信息数据也是呈爆炸性增长. 而推荐系统的出现提供了一种从庞杂的信中提取信息的手段, 在学术和工业界引起了极大的关注<sup>[1]</sup>.

推荐系统用来帮助用户发现与其兴趣相关项目并

提供不断提高用户体验. 在工业推荐系统中, 数据存储空间通常包含数十亿个项目, 这使得无法对每个项目与用户之间的相似度进行全面计算. 因此, 工业推荐系统的第 1 阶段是检索项目的候选生成. 在这个阶段, 系统会选择数百或数千个项目作为候选项, 然后将它们发送

① 基金项目: 国家自然科学基金面上项目 (62272355); 湖北省教育厅科学研究计划指导性项目 (B2022002)

收稿时间: 2023-04-24; 修改时间: 2023-05-23; 采用时间: 2023-06-06; csa 在线出版时间: 2023-09-19

CNKI 网络首发时间: 2023-09-21

到后续的评分和排名阶段进行更精细的处理。候选生成的目标是在不消耗过多计算资源的情况下,为用户 提供有吸引力的产品或服务。为了实现这个目标,候选生成器通常会采用一些技术,例如倒排索引和图上的随机游走,候选生成器的任务是在计算成本可控的前提下返回有限数量的候选项目。

顺序推荐系统捕获用户从过去到最近一段时间内的兴趣,进而根据用户顺序参与项目的历史信息力求准确的推荐对用户最具吸引力的项目;对于顺序推荐系统从不同角度利用用户的顺序参与记录,可分为两种方式:以用户为中心和以项目为中心的模型。

近来的研究中发现<sup>[2,3]</sup>对个人消费兴趣进行多重嵌入的重要性,并希望以此来获取到用户的多重兴趣,因为用户不会只对单一的主题感兴趣,可能同时对多个主题感兴趣;然而多个兴趣的折叠嵌入可能与其他主题最为接近。此外,在分类和链接预测任务中显示了多重嵌入的优势<sup>[4]</sup>。在多重嵌入中,每个嵌入的权重也是十分重要的,以往的多重嵌入方法,例如 Pinner-Sage<sup>[5]</sup>,假设嵌入的数量在所有用户中都是固定的,这样会导致超参数的查找成本很高,而且所有用户拥有相同数量兴趣的假设也是不可靠的。一些休眠用户则可以使用一个或两个向量来很好地表示,而其他用户则有更为多样化的兴趣集,需要有几个嵌入来表示他们。在最近的一项研究<sup>[6]</sup>中提出了用户对每个项目的兴趣是否会持续到训练时间结束后,这个概念被称为兴趣可持续;其明确预测用户每个兴趣项目是否会在最近一段时间内被消费。在顺序推荐模型研究中,Vaswani等<sup>[7]</sup>提出的 Transformer 网络为顺序建模带来了很大发展。对于以用户为中心的模型没有很好的考虑到用户的兴趣持续性是否会超过训练的时间;以项目为中心的模型虽然很好的考虑到了用户兴趣的持续性<sup>[8]</sup>,但是缺乏个性化的推荐。

对于上述问题,需要有一种方法来改进。本文提出的算法模型综合以商品和用户为中心的顺序推荐系统优势,同时引用了个性化和上下文感知的兴趣权重,在多个维度上学习用户兴趣。主要贡献如下。

(1) 利用用户其他的消费商品来补充每个项目用户的消费历史;通过有着相同兴趣的群体用户的消费历史来填充目标用户的稀疏的消费历史。

(2) 利用用户的连续购买(点击)历史为基点找到建立多种嵌入,每个嵌入都代表用户对某个类别的兴

趣,为每个嵌入分配权重,并通过从权重更高的嵌入中检索更多候选对象来改进候选生成的调用。

## 1 相关工作

### 1.1 顺序推荐模型

推荐系统中的一个基本的认知是,用户嵌入应该从用户的历史行为中推断出来<sup>[9]</sup>,因此顺序模型对于推荐模型是非常重要的。之前使用的经典序列模型是马尔可夫链,国内研究者 He 等<sup>[10]</sup>使用马尔可夫链很好地捕捉了一个序列的短期表现,但是其对长序列的记忆表现一般。后来出现的递归神经网络对长序列具有更强的表示能力,Xu 等<sup>[11]</sup>将其用于从任意长序列中学习用户嵌入。近几年,Li 等<sup>[12]</sup>把 Transformer 网络用在顺序推荐模型上取得了比较好的推荐效果。但是一般的模型并没有利用用户历史参与的项目来捕获他们的兴趣偏好。

以用户为中心的顺序模型主要利用点击或者购买商品信息来捕获用户的兴趣偏好。国外研究者提出的 SASRec 模型<sup>[13]</sup>将自注意力机制应用于推荐系统,以捕获与 RNN 和 CNN 相比消费项目之间的长期依赖性;以及 STAR-GCN 模型<sup>[14]</sup>利用图卷积网络(GCN)对用户行为序列进行建模,并通过多层堆叠和重构来提取更丰富的特征表示。近来,国内对于以用户为中心的顺序模型的研究有,LSAN 模型<sup>[15]</sup>基于 CNN 和自注意力机制捕获了消费品之间的局部和全局的交互联系;以及 RippleNet 模型<sup>[16]</sup>利用图神经网络和传播机制,将用户的偏好信号在知识图谱上传播,以进行个性化推荐。虽然以上这些模型都达到了比较好的效果,但并没有考虑到用户的兴趣是否会持续到训练时间结束之后。以项目为中心的模型通过利用所有用户对每个项目的消费历史来捕获用户对每个产品的一般兴趣偏好。根据国内以往的以项目为中心的研究来看,如 Wang 等<sup>[17]</sup>考虑了每个项目最后一次消费之后的时间段,以预测未来项目的重复消费;DIN 模型<sup>[18]</sup>利用注意力机制建模用户对项目的兴趣,同时考虑项目之间的关系,提高点击率预测的精度和个性化能力。同样,国外对于此的研究也取得了一些进展,如 CRIS 模型<sup>[7]</sup>通过预测每个项目是否在最近一段训练时间内消费,以此来预测用户对项目的普遍兴趣是否会持续到未来;NISAML 模型<sup>[19]</sup>用于下一个项目的推荐,利用自注意力机制和度量学习方法,捕捉用户行为序列中的关键信息,并进行个性

化推荐;近来以项目为中心的模型<sup>[7]</sup>比以用户为中心的模型显示出更好的推荐准确性.然而,这些模型侧重于非个性化的兴趣偏移,其倾向于推荐一般消费品,而没有考虑到每个用户的个性化需求,例如爱好素食或肉食.本文通过预测每个用户的消费来获取个性化兴趣可持续性和预测用户在最近一段训练时间内的消费量,来了解每个用户在训练时间之外可能消费哪些项目.

## 1.2 多兴趣用户表示

使用多种嵌入的方式表示用户大大提高了推荐质量,但并不是所有现有的推荐模型都能很容易地扩展到多兴趣框架.经典的协同过滤和矩阵分解方法不会自然产生多个用户嵌入,序列模型(如RNN和基于注意力的模型)也不会自然产生.为了从用户参与项目历史中获取其多兴趣偏好,国内外的研究方法大都是采用了启发式方法<sup>[20]</sup>和无监督学习方法,国外的研究方法主要有,如层次化聚类方法<sup>[5]</sup>和本体论社区挖掘方法<sup>[21]</sup>;国内关于这方面采用的方法主要有,混合推荐和基于图的推荐<sup>[22]</sup>.此外,科研人员还努力修改现有的神经网络以产生多种结果,如文献<sup>[23]</sup>使用的动态路由胶囊网络和文献<sup>[24]</sup>使用的多头自注意力模型.然而,以上的模型是将用户兴趣的数量作为超参数进行估计,并且不了解兴趣的权重.因此,它们为每个用户生成相同数量的聚类,并以相同的权重对待用户的每个兴趣.

## 2 MIES 算法

本文提出的算法模型,综合利用了顺序推荐系统的优势以及引用了个性化上下文感知兴趣权重,从而捕捉用户的多维兴趣以及其兴趣的持续性;该算法模型名为MIES(multi internal external sustainability).该算法定义 $D = \{u, i, t\}$ 为训练的数据,其中 $u$ 是用户, $i$ 为参与的项目, $t$ 为时间. $U$ 和 $I$ 表示用户和项目的集合,以 $h_u = (i, t)$ ,用户 $u$ 和项目 $i$ 的交互时间 $t$ 作为算法的输入.对于每个用户 $u \in U$ ,从历史参与顺序来看,有 $S = (P_{t_1}^u, P_{t_2}^u, \dots, P_{t_{|u|}}^u)$ 或 $(V_{t_1}^u, V_{t_2}^u, \dots, V_{t_{|u|}}^u)$ 以及时间戳向量 $T = (t_{t_1}^u, t_{t_2}^u, \dots, t_{t_{|u|}}^u)$ .该算法的目标是学习一组用户嵌入 $Z_i^u \in \mathbb{R}^d (i = 1, \dots, k)$ 和其权重 $W_i$ ,并预测每个用户在最近的训练时间内消费了哪些项目.

### 2.1 个性化兴趣的持续性

因为事先设置了预测用户在训练时间内消费项目的目标,因此该模型可以像以项目为中心的模型一样侧重于用户最近的偏好;由于测试时间和捕获用户最

新偏好的时间的间隔很短,可以提供准确的推荐.本文将最接近训练时间的一段时间认为是未来,并根据时间 $T$ 将用户的历史 $h_u$ 分为过去和最近两部分:

$$h_u = h_u^p \cup h_u^r \quad (1)$$

$$h_u^p = (i, t) | (i, t) \in h_u, t < T \quad (2)$$

$$h_u^r = (i, t) | (i, t) \in h_u, t \geq T \quad (3)$$

其中, $h_u^p$ 和 $h_u^r$ 是用户消费历史 $h_u$ 过去和最近的部分.

因为把用户的消费历史分为了两部分,所以算法要根据用户过去的消费历史 $h_u^p$ 预测每个用户在最近一段时间内的消费情况 $\{i | i \in h_u^r\}$ ;文中定义用户 $u$ 是否在训练时间内消费了项目 $i$ 的标签为:

$$y_{u,i} = \mathbb{1}[i \in h_u^r] \quad (4)$$

其中, $y_{u,i}$ 是二元标签, $\mathbb{1}$ 取值在 $\{0, 1\}$ 的指标函数.

### 2.2 以内在和外方案补充用户的消费历史

内在补充方案如图1所示,在用户消费的其他项目情况下,增加用户对一个项目的消费历史来缓解数据稀疏.其基本思想是,如果用户最近消费了类似的项目,则用户对类似项目的兴趣被认为是持续的.总体上来说,内在方案的目标是利用用户消费其他商品的特征值,基于一种项目相似度的计算来增强当前项目的特征值:

$$b_{u,i}^I = b_{(u,i)} + \sum_{j \in h_u^p \setminus \{i\}} a_{i,j} \cdot b_{u,j} \quad (5)$$

$$a_{i,j} = \text{sim}(v_i, v_j) \quad (6)$$

其中, $b_{u,i}^I \in \mathbb{R}^N$ 表示通过聚合用户 $u$ 的其他消费项目的特征来增强当前项目 $i$ 的特征值; $\text{sim}$ 是归一化余弦相似度 $\text{sim}(\cdot, \cdot) = (\cos(\cdot, \cdot) + 1/2 + \tau)$ 具有可调参数 $\tau \in \mathbb{R}$ 和项目 $i$ 的嵌入向量 $v_i \in \mathbb{R}^K$ .因此,该方案通过参考用户消费与当前项目高度相关的其他项目的消费历史来补充每个用户的目标项目的消费历史.进而可以做出以下预测:

$$\hat{y}_{u,i}^I = f_s(b_{u,i}^I + e_{u,i}) \quad (7)$$

其中, $\hat{y}_{u,i}^I$ 是基于内在方案的特征预测,即 $b_{u,i}^I$ .此外,还使用用户-项目联合表示 $(e_{u,i} = u_u + v_i)$ 作为用户 $u$ 和项目 $i$ 对模型的一个附加输入,其中 $u_u \in \mathbb{R}^K$ 是用户的嵌入向量.

外在补充方案同样在图1中给出了说明,通过参考有着相同兴趣爱好的用户的消费历史,来补充完善

每个用户的消费历史. 换言之, 就是利用有着相同爱好人群对指定项目的兴趣来预测目标用户对该项目的兴趣. 首先, 为指定项目*i*聚集有着相同爱好的用户群体的特征:  $b_{u,i}^E = \sum b_{u',i} \beta_{u',i} \cdot b_{u',i}$ , 其中 $b_{u,i}^E$ 是目标项目*i*所对应目标用户*u*的同样爱好人群的聚集特征, 其他用户为*u'*. 而 $B_{u,i}$ 是消费目标项目*i*中除对应的用户*u*和 $\beta_{u,u'}$ 之外其他用户的特征, 是目标用户*u*和其他用户*u'*之间的相似性, 如下所示:

$$B_{(u,i)} = \{b_{u',i} | u' \in U \setminus \{u\}, i \in h_{u'}^p\} \quad (8)$$

$$\beta_{u,u'} = sim(u_u, u_{u'}) \quad (9)$$

相似性函数 $sim$ 与用余弦相似性定义的式(1)相同. 此外, 基于式(1)还设置有着相同兴趣爱好的群体对目标项目*i*的消费标签, 使得:

$$y_{(u,i)}^E = \xi \left[ \sum_{u'} \beta_{u,u'} \cdot y_{u',i} \geq 1 \right] \quad (10)$$

其中, 聚合标签 $y_{u,i}^E$ 表示如果相同兴趣爱好的用户对项目*i*的加权消费次数大于一次, 即表示目标项目*i*被消费一次.

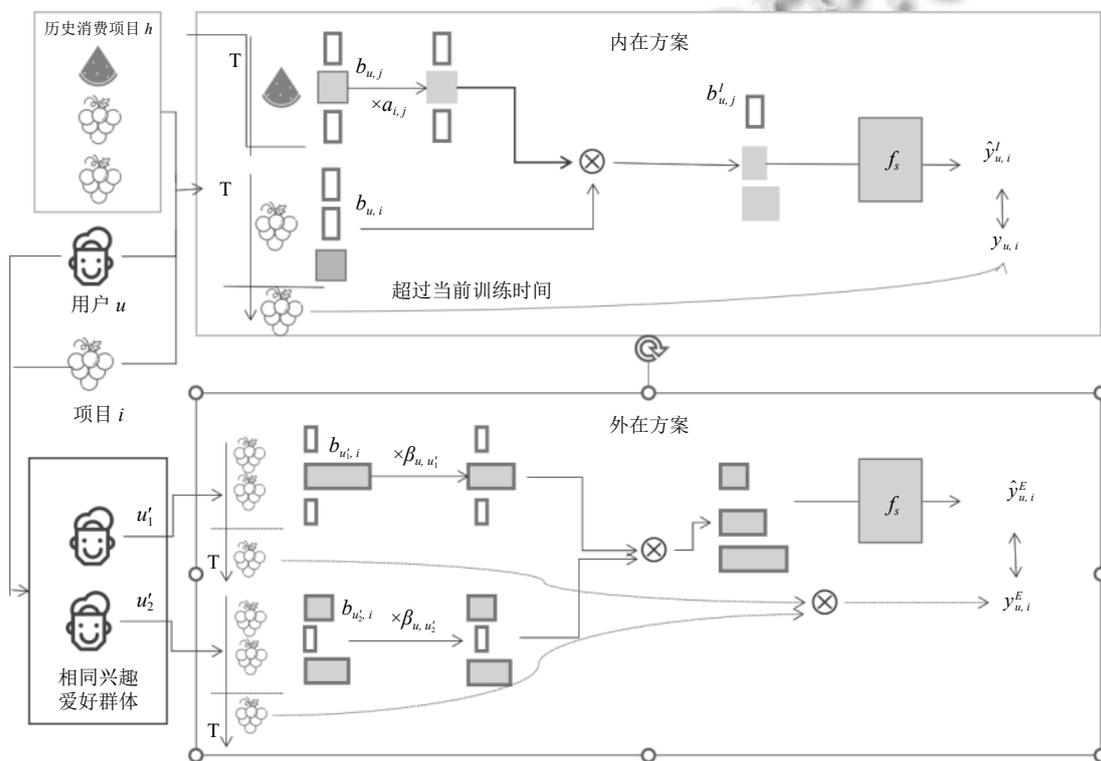


图1 用户消费历史的补充方案

### 2.3 算法模型

本文所提出的算法模型 MIES, 图2 给出了本文算法框架. 其由以下3部分构成, 分别为: 学习多兴趣项目的用户嵌入、兴趣聚类的选择以及兴趣聚类权重持续; 在第2.2.1-2.2.3节中分别介绍了构成算法框架的每个模块.

#### 2.3.1 多兴趣用户嵌入

用户的兴趣偏好可以通过其消费项目的顺序与时间来很好地表示出来, 因此, 可以将项目特征 $b_{u,i}$ 与其时间和位置信息连接起来以产生动作编码:

$$e_j = [p_j; \tau(t_j)] \quad (11)$$

其中,  $[\cdot]$  表示向量的串联, 时间戳的单位为天, 位置编码 $\rho$ 和时间编码 $\tau$ 由式(12)-式(15)可得:

$$\tau_{2j}(t_i) = \sin(t_i / (\tau_{\max})^{2j/m_t}) \quad (12)$$

$$\tau_{2j+1}(t_i) = \cos(t_i / (\tau_{\max})^{2j/m_t}) \quad (13)$$

$$\rho_{2j}(t_i) = \sin(t_i / (\rho_{\max})^{2j/m_p}) \quad (14)$$

$$\rho_{2j+1}(t_i) = \cos(t_i / (\rho_{\max})^{2j/m_p}) \quad (15)$$

为了使用向量来表示用户参与的整个项目, 算法

模型必须要考虑到项目之间的相似性,此过程是通过 self-attention 层实现的:

$$e_{(i,j)}^h = \frac{((W_q^h e_j + b_q^h)^T \cdot (W_r^h e_i + b_r^h))}{\sqrt{d_{\text{model}}}} \quad (16)$$

$$a_{i,j}^h = \text{Softmax}_i(e_{i,j}^h) \quad (17)$$

其中,  $W_i^h$  和  $b_i^h$  分别表示查询投影权重和偏差. 使用掩码矩阵  $M$  和约束注意力分数  $a$ , 把每个注意力头汇聚成:

$$z_j^h = \sum_i a_{i,j}^h M_{i,j} p_i \quad (18)$$

为了处理来所有注意力头的聚合向量,在自注意力层之后应用了 *Dropout* 层、*LayerNorm* 层和 *FCL* 层,并将输出向量计算为:

$$z_j = \text{FCL}(\text{LayerNorm}(\text{Dropout}(\{z_j^1; \dots; z_j^l\}))) \quad (19)$$

其中,  $j = 1, \dots, l$ , *FCL* 有两个全连接层,层之间有一个 ReLU 函数作为激活函数. 经过注意力层和全连接层的输出后,现在每个向量  $z_j$  不仅包含当前项目的特征  $b_{u,i}$ ,还包含与其他项目和相邻项目特征的邻接关系,那么接下来的工作就是要从这些向量  $\{z_j\}$  中,找出哪些是最具有代表性的.

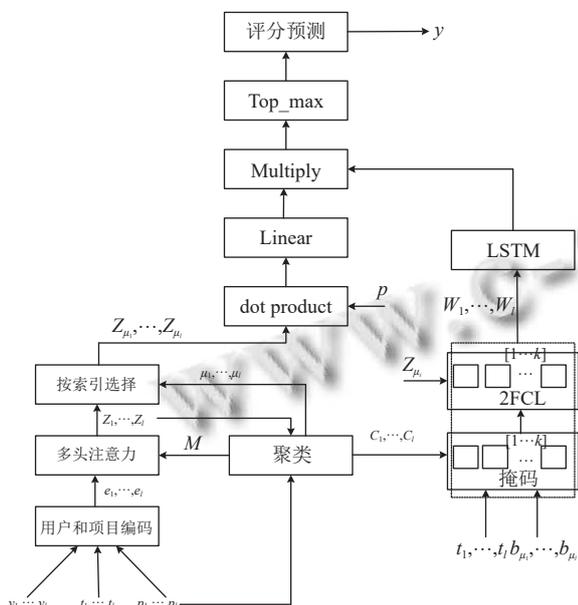


图2 MIES 算法框架

由于多头注意力模块已经产生了  $l$  个输出向量  $z_1, \dots, z_l$ , 每个  $z_i$  使用  $b_{u,i}$  作为查询. 如何从中找到更具代表性的向量, 使用聚类将每个  $b_{u,i}$  中的最后一项表示为

$b_{u,1}, \dots, b_{u,k}$ , 其作为查询的  $z_*$  表示相应的集群. 设用户表示为  $Z \in \mathcal{R}^{k \times d}$ , 则:

$$Z = [z_{\mu_1}^T; \dots; z_{\mu_k}^T] \quad (20)$$

### 2.3.2 兴趣聚类的选择

由于 Ward 聚类算法在 PinnerSage<sup>[5]</sup> 中的显示出了非常好的效果, 因此将其应用于 MIES 中, 有利于探索协同过滤数据集上聚类算法的选择和聚类数量. 为了证明 Ward 算法在 MIES 中的重要性, 将在实验部分比较各种聚类算法在模型上的表现.

### 2.3.3 兴趣聚类权重可持续模块

在用户存在多种兴趣之外, 对每种兴趣的爱好是不均等的, 因而可以根据用户对不同兴趣爱好的侧重进行排序进而选出其中比较关键的候选项目, 即用户对某一兴趣相关的项目的参与度更高, 则对这一兴趣项目聚类分配更高的权重. 为更好利用该聚类种的上下文信息和用户参与序列, 在聚类表示  $z_j$  和属于聚类  $j$  的项目  $\tau$  的时间编码之上构建了一个两层前馈网络; 在输入前馈网络之前, 会对不属于当前聚类的项目遮掩为 0, 以此保证输入的维度一致. 因而聚类权重可以表示为:

$$w_j = \text{FCL}(\{z_j; \zeta_{[c_1 \in c_2]} \cdot \tau_1; \dots; \zeta_{[c_1 \in c_2]} \cdot \tau_j\}) \quad (21)$$

具体而言, *FCL* 由两个全连接层和一个 *LayerNorm* 层组成, Sigmoid 在其中作为激活函数, 归一化层保证梯度不会消失, 使得模型收敛速度变快. 而多头注意力模块输出的是用户感兴趣的主体, 因此通过 *FCL* 输出的权重也应当是正的, 因此第 2 个全连接层的输出由 log 函数归一化到  $[0, +\infty)$  上.

对于用户来说, 只要当前显示的项目在属于其兴趣的一种, 用户就在参与互动. 即项目嵌入只需要非常接近于用户嵌入的其中一种, 而不是接近所有的用户嵌入. 因此, 用户-项目的匹配度由每个兴趣维度上项目嵌入和用户嵌入的最大值决定; 此外, 用户对每个兴趣聚类的匹配度也应该表现在, 当用户更多地参与数据某一兴趣聚类的项目时, 应该为这个兴趣聚类分配更高的权重. 因而, 把用户与项目匹配度表示为:

$$y = \max\{w_j \text{Linear}(z_j \cdot b_{u,i})\}_{j=1}^k \quad (22)$$

其中,  $Z = [z_{\mu_1}^T; \dots; z_{\mu_k}^T]$  是用户在  $k$  维度上的多兴趣嵌入,  $b_{u,i}$  为项目嵌入.

为了捕获到  $b_{u,i}$  和多兴趣嵌入  $Z$  的序列动态, 在

FCL 层之后使用了长短期记忆网络 LSTM, 根据来自 LSTM 最后隐藏表示  $h_{u,i} \in \mathbb{R}^k$  来预测当前项目的可持续性标签. 因此在这部分, 模型将  $h_{u,i}$  作为最后输出的特征分类标签. 由于随着处理的序列模型增加, 可能会出现梯度爆炸的问题, 模型通过梯度裁剪解决该问题.

### 2.3.4 模型训练

在训练 MIES 模型时, 本文将负似然对数 (NLL) 作为模型的损失函数:

$$\xi = - \frac{\sum_{u \in U} \left( \sum_{b_{u,i} \in \xi_+^u} \log(y_i^u) + \sum_{b_{u,i} \in \xi_-^u} \log(1 - y_i^u) \right)}{\sum_{u \in U} (|\xi_+^u| + |\xi_-^u|)} \quad (23)$$

其中,  $\xi_+$  和  $\xi_-$  为数据集中的带有正标签和负标签的数据.

## 3 实验

### 3.1 数据集

本次实验使用了 3 个具有代表性的数据集: Amazon-book, Advertisement 和 Miniprogram, 其中, Miniprogram 是团队开发的小程序在使用过程中收集的用户点击商品的相关信息, 这 3 个数据集主要包括用户 ID、项目 ID 和时间戳, 具体统计数据如表 1 所示.

表 1 实验数据集描述

数据集	Amazon-book	Advertisement	Miniprogram
用户数	8026324	1141729	3263
项目数	2330066	662061	10545
时间跨度 (ms)	573523200	50808600	15418494

### 3.2 评价标准

本文提出的算法是对用户潜在的兴趣偏好项目进行预测, 所以采用命中率 (HR)、ROC 曲线下面积 (AUC) 和归一化折损累计增益 (NDCG) 作为预测准确度指标. 计算公式如下:

$$HR = \frac{1}{N} \sum_{i=1}^N hit(i) \quad (24)$$

$$AUC = \frac{\sum_{i \in \text{positiveClass}} rank - \frac{M(1+M)}{2}}{M \times N} \quad (25)$$

$$NDCG = \frac{\sum_{i=1}^p 2^{rel_i-1} / \log_2(i+1)}{\sum_{i=1}^{|REL|} 2^{rel_i-1} / \log_2(i+1)} \quad (26)$$

### 3.3 聚类算法对比

为了验证 Ward 聚类算法在本文中的有效性, 选择

了 Spectral<sup>[25]</sup>、BIRCH、HOP-DBSCAN<sup>[26]</sup>、HDC<sup>[27]</sup> 与其进行对比, 簇数根据需要设置为 4、9、12.

不同的聚类算法对性能有边际影响, PinnerSage 提出了 Ward 算法优于 Spectral, 根据下表可以看出并不矛盾. 聚类方法适用于外生的项目嵌入, 因此聚类方法会受到非平面几何和异常值的影响. 对于协作过滤数据集, 聚类方法适用于由多头自注意力层产生的编码, 该层对项目和其他项目的嵌入进行平均. 经过多头注意力处理之后的编码应该平滑分布, 因此任何聚类方法在这方面几乎同样有效. 再而, 选择聚类的数量是一个不简单的权衡. 减少聚类数量的动机是存储和计算成本, 这些成本随着聚类数量的增加而线性增长. 具体聚类算法的表现如表 2 所示.

表 2 聚类算法在 AUC 上的表现

聚类算法	聚类簇数	Amazon-book	Advertisement
Ward	4	<b>0.8026</b>	0.8799
	9	0.8106	<b>0.8976</b>
	12	0.8231	<b>0.8991</b>
Spectral	4	0.7892	0.8816
	9	0.8156	0.8921
	12	<b>0.8286</b>	0.8906
BIRCH	4	0.7983	0.8837
	9	0.8052	0.8965
	12	0.8145	0.8987
HOP-DBSCAN	4	0.8052	0.8965
	9	0.8145	0.8987
	12	0.7996	0.8902
HDC	4	0.7996	0.8902
	9	<b>0.8162</b>	0.8972
	12	0.8223	0.8990
HDC	4	0.7864	0.8803
	9	0.8092	0.8935
	12	0.8257	0.8983

### 3.4 参数设置

选取 3 个数据集中的 60% 的最后一个月, 即 30 天作为训练集, 20% 的最后一个月作为验证集, 剩下的数据作为测试集. 本文将 Adam 优化器的学习率调整为 {0.01, 0.005, 0.001}, 多头注意力设置为 {8, 16, 32}, 批量大小调整为 {64, 128, 256, 512, 1024, 2048}, 嵌入的维度  $k$  在 {16, 32, 64, 128} 中, 使用 Xavier 初始化在模型中初始化用户和项目嵌入. 从图 3 可以看出当嵌入维度为 32 时 NLL 值最小, 效果最好.

为验证多兴趣的权重是随用户参与项目而不断变化的, 将 MIES 与聚类权重模块的两个变体进行比较, 即等权重聚类模块和指数衰减权重聚类模块. 等权重是将 1 作为聚类权重, 进而产生加权相等的用户多兴趣表示; 指数衰减权重由  $w_j = \sum_{C_i \in j} \exp(-\lambda(t_{\text{now}} - t_i))$  可得, 其中  $t_{\text{now}}$  为用户最后一次参与项目的时间. 另外设置  $\lambda=0.01$  和聚类的数量  $k=5$ .

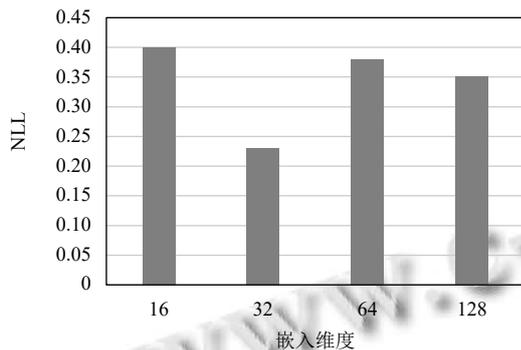


图3 Amazon-book 数据集下嵌入维度对 NLL 的影响

图 4 中分别展示了在 3 种不同权重的模型中, 3 个数据集在更新过程中不同时间点的性能变化曲线; A\_B 表示 Amazon-book 数据集, A\_D 表示 Advertisement 数据集, M\_P 表示 Miniprogram 数据集; 而 1, 2, 3 分别为不同的时间段, 即每次以  $2 \times 10^3$  叠加增长. 从图 4 可以看出等权重和指数衰减权重在 AUC 上表现较为平稳, 但是学习权重在总体上对模型的帮助是优于其他两个权重的, 因而, 基于个性化和上下文感知的聚类权重是用于推荐系统的更为合理的方案.

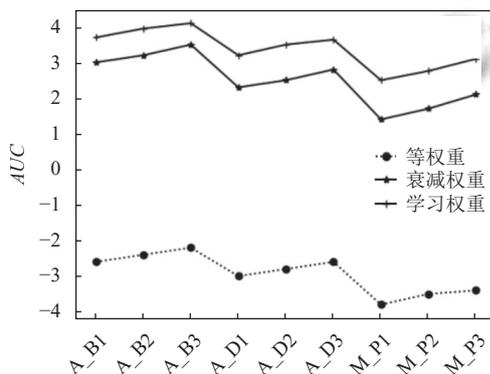


图4 3 类数据集下不同的聚类权重方式对模型的影响

图 5 展示了推荐的前 10 个项目的精确度随着迭代周期的变化, 每次迭代都会对当前推荐的项目进行一次评分. 其中, 迭代周期为 21 的时达到了最优值,

但是可以看出来当超过 21 时, 随着周期数的增加, 精确度的值在逐步下降, 即出现了过拟合的问题.

经过验证集的验证最终得到的参数是: 用户和项目嵌入维度为 32, 批量大小为 64, 多头注意力的注意力头的数量为 8, 学习率为 0.01, 迭代周期为 21.

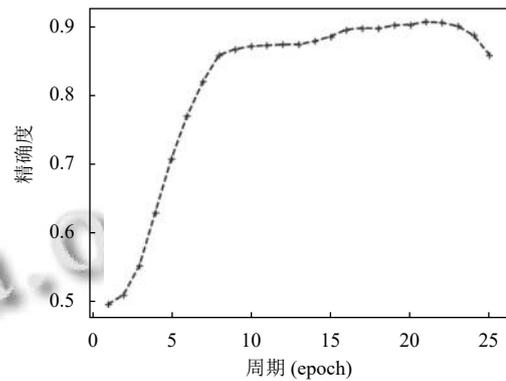


图5 Amazon-book 数据集下迭代周期数变化对 NDCG 的影响

### 3.5 算法性能对比

为验证本文算法的有效性选择了 BPR、Caser、LSAN、NeuMF、SimpleX 等经典推荐算法进行对比, 根据对比实验的结论进行了分析.

BPR<sup>[28]</sup>: 是基于贝叶斯后验优化的个性化排序算法, 本身不优化用户对物品的评分, 只是借由评分来优化用户对物品的排序.

Caser<sup>[29]</sup>: 采用卷积神经网络捕获用户最近活动的动态模式影响, 结合用户长期偏好以及短期兴趣, 更侧重于短期信息的影响.

LSAN<sup>[15]</sup>: 利用组成嵌入的概念并同时注意到了项目的内在动态性, 提出一种时态上下文感知的嵌入合成方案, 通过双注意力网络有效的学习偏好.

NeuMF<sup>[30]</sup>: 是一个典型的基于深度学习的推荐算法, 结合了传统矩阵和多层感知机, 同抽取高纬度和低纬度的特征.

SimpleX<sup>[31]</sup>: 提出了两段式的搜索架构, 在针对用户历史行为序列的数据时有着更好的扩展性和准确性.

在 Amazon-book 和 Advertisement 数据集的 60% 的最后一个月, 即 30 天作为训练集, 20% 的最后一个月作为验证集, 剩下的数据作为测试集. 并于上述模型进行了对比, 总体 NDCG 和 Recall 的比较结果如表 3 所示.

为了进一步验证 MIES 算法在一般数据集上的健

壮性,本文分别在 Amazon-book 和 Advertisement 数据集上训练出相关的推荐模型,进而在 Miniprogram 数

据集上测试算法的推荐性能,总体 *NDCG* 的比较结果如表 4 所示。

表 3 MIES 算法与其他模型对比结果

训练集 评价指标	Amazon-book			Advertisement		
	<i>NDCG@10</i>	<i>Precision@10</i>	<i>Recall@10</i>	<i>NDCG@10</i>	<i>Precision@10</i>	<i>Recall@10</i>
BPR	0.3582	0.3396	0.2536	0.3716	0.3483	0.2382
Caser	0.4231	0.4006	0.1632	0.4557	0.4361	0.1494
LSAN	0.4956	0.4839	0.1468	0.5371	0.5136	0.1106
NeuMF	0.4396	0.4086	0.1653	0.5552	0.5362	0.1352
SimpleX	0.4072	0.3865	0.1559	0.4838	0.4650	0.1436
MIES	0.6283	0.6174	0.0865	0.6842	0.6705	0.0634

表 4 Miniprogram 在其他数据集训练模型上的

*NDCG@10* 表现

训练模型	Amazon-book	Advertisement
Miniprogram	0.6743	0.7135

#### 4 结论与展望

在本文中提出了一个能获取到用户多个兴趣的偏好以及个性化兴趣的可持续性的推荐系统,首先设计了简单有效的方式补充用户过往的消费历史,使得用户的消费历史变得饱和而又不失合理,然后基于不同用户有着不同数量类型兴趣的前提下,用多个嵌入表示用户的兴趣,并基于最接近用户嵌入的项目嵌入来定义它们的关联性,进而更新相应聚类权重,此外还模型种还设计了多头注意力机制来处理来自多用户多兴趣的嵌入,更加深层次的凸显了用户对每个兴趣的不同偏好。最后,结合具有相应权重的多兴趣嵌入通过 LSTM 网络对用户最新的兴趣进行“记忆”,进而对用户可能潜在的兴趣项目进行评分,提高了推荐性能的准确度。未来的工作中需要考虑到用户所处特殊时间节点,如商家大促(双十一)等,可能兴趣波动会给推荐效果较大带来的影响。

#### 参考文献

- Kim D, Park C, Oh J, *et al.* Convolutional matrix factorization for document context-aware recommendation. Proceedings of the 10th ACM Conference on Recommender Systems. Boston: ACM, 2016. 233–240.
- Epasto A, Perozzi B. Is a single embedding enough? Learning node representations that capture multiple social contexts. Proceedings of the 2019 World Wide Web Conference. San Francisco: ACM, 2019. 394–404.
- Yu L, Liu L, Li XF. A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in e-commerce. Expert Systems With Applications, 2005, 28(1): 67–77. [doi: 10.1016/j.eswa.2004.08.013]
- Liu NH, Tan QY, Li YN, *et al.* Is a single vector enough?: Exploring node polysemy for network embedding. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage: ACM, 2019. 932–940.
- Pal A, Eksombatchai C, Zhou YT, *et al.* PinnerSage: Multimodal user embedding framework for recommendations at pinterest. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2020. 2311–2320.
- Ma C, Kang P, Liu X. Hierarchical gating networks for sequential recommendation. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage: ACM, 2019. 825–833.
- Vaswani A, Shazeer N, Parmar N, *et al.* Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000–6010.
- Hyun D, Cho J, Park C, *et al.* Interest sustainability-aware recommender system. Proceedings of the 2020 IEEE International Conference on Data Mining (ICDM). Sorrento: IEEE, 2020. 192–201.
- Wang JL, Ding KZ, Hong LJ, *et al.* Next-item recommendation with sequential hypergraphs. Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. Xi'an: ACM, 2020. 1101–1110.
- He RN, McAuley J. Fusing similarity models with Markov chains for sparse sequential recommendation. Proceedings of the 16th IEEE International Conference on Data Mining (ICDM). Barcelona: IEEE, 2016. 191–200.

- 11 Xu CF, Zhao PP, Liu YC, *et al.* Recurrent convolutional neural network for sequential recommendation. Proceedings of the 2019 World Wide Web Conference. San Francisco: ACM, 2019. 3398–3404.
- 12 Li JC, Wang YJ, McAuley J. Time interval aware self-attention for sequential recommendation. Proceedings of the 13th International Conference on Web Search and Data Mining. Houston: ACM, 2020. 322–330.
- 13 Kang WC, McAuley J. Self-attentive sequential recommendation. Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM). Singapore: IEEE, 2018. 197–206.
- 14 Zhang JN, Shi XJ, Zhao SL, *et al.* STAR-GCN: Stacked and reconstructed graph convolutional networks for recommender systems. Proceedings of the 28th International Joint Conference on Artificial Intelligence. Macao: AAAI Press, 2019. 4264–4270.
- 15 Li Y, Chen T, Zhang PF, *et al.* Lightweight self-attentive sequential recommendation. Proceedings of the 30th ACM International Conference on Information & Knowledge Management. Gold Coast: ACM, 2021. 967–977.
- 16 Wang HW, Zhang FZ, Wang JL, *et al.* RippleNet: Propagating user preferences on the knowledge graph for recommender systems. Proceedings of the 27th ACM International Conference on Information and Knowledge Management. Torino: ACM, 2018. 417–426.
- 17 Wang CY, Zhang M, Ma WZ, *et al.* Modeling item-specific temporal dynamics of repeat consumption for recommender systems. Proceedings of the 2019 World Wide Web Conference. San Francisco: ACM, 2019. 1977–1987.
- 18 Zhou GR, Mou N, Fan Y, *et al.* Deep interest evolution network for click-through rate prediction. Proceedings of the 33rd AAAI Conference on Artificial Intelligence. Honolulu: AAAI, 2019. 5941–5948.
- 19 Zhang S, Tay Y, Yao LN, *et al.* Next item recommendation with self-attentive metric learning. Proceedings of the 3rd AAAI Conference on Artificial Intelligence. Hawaii: AAAI, 2019.
- 20 Jiang H, Wang WJ, Wei YW, *et al.* What aspect do you like: Multi-scale time-aware user interest modeling for micro-video recommendation. Proceedings of the 28th ACM International Conference on Multimedia. Seattle: ACM, 2020. 3487–3495.
- 21 Arafteh M, Ceravolo P, Mourad A, *et al.* Ontology based recommender system using social network data. Future Generation Computer Systems, 2021, 115: 769–779. [doi: 10.1016/j.future.2020.09.030]
- 22 Darban ZZ, Valipour MH. GHRS: Graph-based hybrid recommendation system with application to movie recommendation. Expert Systems with Applications, 2022, 200: 116850. [doi: 10.1016/j.eswa.2022.116850]
- 23 Li C, Liu ZY, Wu MM, *et al.* Multi-interest network with dynamic routing for recommendation at Tmall. Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing: ACM, 2019. 2615–2623.
- 24 Liu J, Chen S, Wang B, *et al.* Attention as relation: Learning supervised multi-head self-attention for relation extraction. Proceedings of the 29th International Conference on International Joint Conferences on Artificial Intelligence. 2021. 3787–3793.
- 25 谢娟英, 丁丽娟, 王明钊. 基于谱聚类的无监督特征选择算法. 软件学报, 2020, 31(4): 1009–1024. [doi: 10.13328/j.cnki.jos.005927]
- 26 Lai WH, Zhou MR, Hu F, *et al.* A new DBSCAN parameters determination method based on improved MVO. IEEE Access, 2019, 7: 104085–104095. [doi: 10.1109/ACCESS.2019.2931334]
- 27 McInnes L, Healy J, Astels S. HDBSCAN: Hierarchical density based clustering. The Journal of Open Source Software, 2017, 2(11): 205. [doi: 10.21105/joss.00205]
- 28 Rendle S, Freudenthaler C, Gantner Z, *et al.* BPR: Bayesian personalized ranking from implicit feedback. Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence. Montreal: AUAI Press, 2012. 452–461.
- 29 Tang JX, Wang K. Personalized top-N sequential recommendation via convolutional sequence embedding. Proceedings of the 11th ACM International Conference on Web Search and Data Mining. Marina Del Rey: ACM, 2018. 565–573.
- 30 He XN, Liao LZ, Zhang HW, *et al.* Neural collaborative filtering. Proceedings of the 26th International Conference on World Wide Web. Perth: International World Wide Web Conferences Steering Committee, 2017. 173–182.
- 31 Mao KL, Zhu JM, Wang JP, *et al.* SimpleX: A simple and strong baseline for collaborative filtering. Proceedings of the 30th ACM International Conference on Information & Knowledge Management. Gold Coast: ACM, 2021. 1243–1252.

(校对责编: 牛欣悦)