**文章编号**:1671-9352(2026)03-0075-11 DOI:10.6040/j.issn.1671-9352.2.2024.089

# 带有均匀性约束和重要性均衡的基于原型的推荐方法

曹玉祥,廉涛*,王龙,荆星博,窦浩铖
(太原理工大学人工智能学院, 山西 晋中 030600)

**摘要**:基于原型的推荐算法可以通过学习一组代表典型偏好或共性特点的用户原型(或物品原型)表示,以及用户(或物品)与原型之间的关联强度,实现可解释的推荐。然而,现有算法忽视了原型表示之间的差异性以及不同原型之间的负载均衡,不能充分释放模型的表达能力。因此,以 ProtoMF 为基础,提出一种带有均匀性约束和重要性均衡的基于原型的推荐方法 ProtoMF++。该方法在用户原型(或物品原型)之间添加均匀性约束,通过最小化原型表示之间的平均成对高斯势的对数,提升原型表示之间的差异性;另外,将每个原型与所有用户(或物品)的累计关联强度视作其负载重要性,通过最小化各个原型的负载重要性的变异系数,实现不同原型之间的重要性均衡。在 3 个基准数据集上进行实验,结果表明 ProtoMF++的推荐效果优于现有基于原型的推荐方法,在 Baby 数据集上,HitRatio@ 10 和 NDCG@ 10 指标值分别提升 4.74%和 10.64%。

**关键词**:推荐系统;原型表示;均匀性约束;重要性均衡
**中图分类号**:TP391 **文献标志码**:A
**引用格式**:曹玉祥,廉涛,王龙,等. 带有均匀性约束和重要性均衡的基于原型的推荐方法[J]. 山东大学学报(理学版),2026,61(3):75-85.

# Prototype-based recommendation method with uniformity constraints and importance balancing

CAO Yuxiang, LIAN Tao*, WANG Long, JING Xingbo, DOU Haocheng
(College of Artificial Intelligence, Taiyuan University of Technology, Jinzhong 030600, Shanxi, China)

**Abstract**: Prototype-based recommendation algorithms can achieve explainable recommendations by learning a set of user prototypes (or item prototypes) that represent typical preferences or common characteristics, as well as the association strength between users (or items) and prototypes. However, existing algorithms overlook the differences between prototypes and the load balancing among them, and hence cannot fully release the expressive power of the model. Therefore, a prototype-based recommendation method ProtoMF++ with uniformity constraints and importance balancing is developed on top of ProtoMF. This method added uniformity constraints between user prototypes (or item prototypes) and enhanced the differences between prototypes by minimizing the logarithm of the average pairwise Gaussian potential between prototype representations. In addition, the load importance of each prototype is defined as the total association strength between it and all users (or items), and the coefficient of variation of their load importance is minimized to realize importance balancing across different prototypes. Experiments are conducted on three benchmark datasets, and the results show that ProtoMF++ outperforms existing prototype-based recommendation methods. For example, on the Baby dataset, the values of HitRatio@ 10 and NDCG@ 10 increase by 4.74% and 10.64%, respectively.

**Key words**: recommender system; prototype representation; uniformity constraint; importance balancing

## 0 引言

推荐系统与身处人工智能时代的人们息息相关,如流媒体中的视频推送、购物平台的商品推荐等。推荐算法的一个重要分支是协同过滤[1],其核心在于如何建模用户和物品之间的历史交互,进而预测用户可能

**收稿日期**:2024-10-09; **网络出版时间**:2025-05-12
**基金项目**:国家自然科学基金资助项目(62102279)
**第一作者**:曹玉祥(2000— ),男,硕士研究生,研究方向为推荐系统. E-mail:yxc705841@ gmail.com
*通信作者:廉涛(1988— ),男,副教授,硕士生导师,博士,研究方向为推荐系统、信息检索、大数据挖掘与分析. E-mail:liantao@ tyut.edu.cn

感兴趣的物品。近年来,一些研究探索了原型在推荐系统中的应用,提升了推荐结果的准确性和可解释性[2-3]。例如,ProtoMF[4]学习一组用户/物品原型,每个原型可表示若干相似实体(用户或者物品)的共性特点,并且根据用户/物品与这些原型之间的关联强度,通过线性组合的方式计算用户与物品之间的匹配分数,提升推荐的可解释性和透明性。但是,现有基于原型的推荐方法忽视了原型表示之间的差异性,没有关注它们在隐含空间中的分布情况。此外,建模用户/物品与原型之间的关系时忽视了负载均衡问题,可能导致一些原型与较多用户/物品之间存在紧密关系,而另一些原型仅与个别用户/物品存在较强关联。这些问题均不利于充分发挥模型的表达能力,可能限制模型的推荐性能。一些研究指出在协同过滤中设计复杂的编码器可能仅会带来微弱的性能提升[5],部分研究人员尝试设计除常见的成对 BPR(Bayesian personalized ranking)损失[6]之外的新型优化目标[7-8],并证明这些目标函数能够更加稳定地提升性能。

因此本文的研究目标是:以基于原型的推荐方法为基础,通过引入额外的损失函数,一方面优化原型表示在隐含空间中的分布,增强它们之间的差异性,另一方面平衡各个原型与全量用户/物品之间的累计关联强度,从而进一步提升此类方法的推荐性能。协同过滤中向量表示的质量直接影响到推荐系统的性能。Wang 等[9]研究发现直接优化用户和物品表示的均匀性以及二者之间的对齐性可以有效提升推荐性能。受此启发,本文认为在基于原型的推荐方法中原型表示向量应当尽可能均匀地分布于整个空间中。因此,本文在用户原型(或物品原型)之间施加均匀性约束,增大原型表示之间的差异性,以便它们各自分散辐射不同特点的用户/物品。用户/物品与原型之间存在一种多对多的关系,尽管现有方法要求每个用户/物品至少与一个原型具有紧密关联,并且每个原型至少与一个用户/物品具有紧密关联,但是这无法保证不同原型的负载均衡。近年来,大语言模型采用混合专家网络(mixture of experts, MoE)来处理文本序列中的大量词例(token)。在将词例路由到各个专家子网络时,通过添加额外约束实现负载均衡[10-11],避免有的专家子网络需要处理大量词例,而有的专家子网络接收的词例却寥寥无几。受此启发,本文将每个原型与所有用户/物品的关联强度之和视作该原型的负载重要性,并施加重要性均衡损失,在每个用户/物品仅与少量原型保持较强关联的同时,确保各个原型在全量用户/物品中发挥的整体作用基本相当。

本文的主要贡献可概括为以下 3 点。

1) 指出现有基于原型的推荐方法忽视了原型表示之间的差异性,原型与用户/物品之间的关系建模忽视了负载均衡,不利于充分释放模型的表达能力。

2) 提出 ProtoMF++模型,在用户原型(及物品原型)之间施加均匀性损失,提升原型表示之间的差异性;对各个原型与全量用户/物品之间的累计关联强度施加重要性均衡损失,促进各个原型之间的负载均衡。

3) 在 3 个基准数据集上进行的实验表明 ProtoMF++的推荐效果优于其他基于原型的推荐方法;消融实验证明了均匀性损失和重要性均衡损失均有助于提高模型的推荐性能。

## 1 相关工作

### 1.1 基于原型的推荐方法

大多数协同过滤方法会为用户或物品学习唯一的表示向量,并利用用户和物品的表示向量之间的关系进行推荐。基于原型的推荐方法则学习一组原型表示向量,每个原型代表若干相似用户或物品的共性特点,随后利用用户/物品与原型之间的关系进行推荐。

基于代表的矩阵分解模型(representative-based matrix factorization, RBMF)[12]是一个比较早期的方法,核心思想是通过优化的方法从原始用户—物品交互矩阵中选择具有代表性的若干行(即用户代表)或若干列(即物品代表),并基于其他用户或物品与代表之间的关系进行评分预测,这些代表可视作数据集中的原型。Barkan 等[13]提出一种基于锚点的协同过滤方法(anchor-based collaborative filtering, ACF),首先定义一组反映典型偏好或共性特点的锚点向量,然后将用户和物品表示为锚点向量的凸组合,进而计算用户和物品之间的匹配分数。Du 等[14]提出将物品映射到品味簇(taste clusters)集合中,并通过一些代表性标签来区分不同的品味簇,进而实现可解释的推荐。这种方法仅在用户或物品一侧学习原型表示,或者两端共用一组原型表示,从而限制了模型的表达能力。与之不同,Melchiorre 等[4]提出的 ProtoMF 模型分别在用户和物品两端学习原型表示,并构建两套推荐子模型,最后结合两端模型计算最终的推荐分数。这样做不仅保证了模型

的可分离性,最终的推荐分数可以分解为两端子模型的预测分数,同时也能够通过用户和物品与两端原型的关联强度实现很好的可解释性。

以上基于原型的推荐方法忽视了原型表示之间的差异性,不利于学习更多样化的原型向量;在建模用户/物品与原型的关系时,尽管部分工作添加了包容性和排他性约束,但是没有着重关注各个原型之间的负载均衡问题,无法充分发挥模型的表达能力。

### 1.2　表示均匀性与重要性均衡

近年来,一些研究发现对比学习得到的表示向量的质量与对齐性和均匀性密切相关[15]。其中对齐性要求正例对具有相似的表示;均匀性要求样本表示向量在空间中的分布尽可能均匀。Wang 等[9]将其引入到协同过滤推荐中,从理论层面证明完美的对齐性和均匀性可以达到最优化 BPR 损失的目的。Yan 等[16]在图表示学习中施加均匀性约束将不相关节点的表示分开,极大地提高了模型的节点分类能力。这些研究证实了表示均匀性对于提高模型表达能力的重要性。

在基于大语言模型的自然语言处理任务中,混合专家网络凭借其独特设计,已经展现出优异性能。Shazeer 等[10]通过构建稀疏混合专家网络,并在不同专家之间添加负载均衡损失,极大地提高了模型的效能。本文将这种思想应用到基于原型的推荐方法中,在建模用户/物品与原型之间的关系时引入重要性均衡损失。

## 2　模型

首先简要描述所解决的任务,然后阐述基于原型的推荐方法,包括用户端和物品端原型建模、均匀性约束和重要性均衡的具体实现以及最终的优化目标。

### 2.1　任务描述

本文关注推荐中的隐式反馈[6],假设只能基于隐式交互数据来推断用户的喜好。$\boldsymbol{Y} \in \mathbf{R}^{|U|\times|I|}$ 为用户—物品交互矩阵,其中 $y_{ui}=1$ 表示用户和物品之间存在交互,$y_{ui}=0$ 表示用户和物品之间不存在交互。给定用户和物品之间的交互矩阵 $\boldsymbol{Y}$,本文采用基于原型的方法预测用户 $u$ 对物品 $i$ 的偏好程度。为便于后续章节的叙述,表 1 列出了本文使用的主要符号及其说明。

表 1　符号列表

Table 1　List of symbols

| 符号 | 说明 |
|---|---|
| $U$, $I$ | 用户/物品集合 |
| $u$, $i$ | 某个用户/物品 |
| $\boldsymbol{e}_u$, $\boldsymbol{e}_i$ | 用户/物品表示向量 |
| $d_1$, $d_2$ | 用户/物品表示维度 |
| $K$, $L$ | 用户/物品原型数量 |
| $\boldsymbol{p}_k$, $\boldsymbol{q}_l$ | 用户/物品原型表示向量 |
| $\boldsymbol{c}_u$, $\boldsymbol{c}_i$ | 用户/物品与原型之间的关联强度向量 |

### 2.2　基于原型的推荐过程

本文方法的基本思想如图 1 所示。图 1(a)为基础的 ProtoMF 模型框架,在用户端和物品端分别学习用户原型和物品原型;图 1(b)说明添加原型均匀性约束前后原型在空间中分布的变化;图 1(c)说明在用户/物品与原型之间的关联强度矩阵上如何现各个原型之间的重要性均衡。

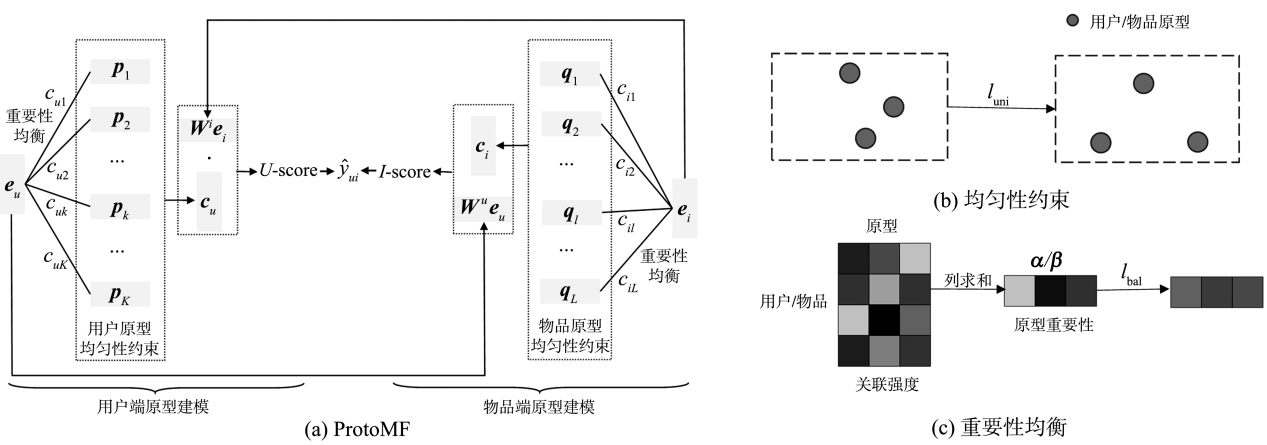


图 1　ProtoMF++模型

Fig.1　ProtoMF++ model

2.2.1 用户端原型建模

假设人群的喜好可以用 $K$ 个用户原型表示,每个原型向量 $\boldsymbol{p}_k \in \mathbf{R}^{d_1}$ 代表了某类相似用户的共同偏好。例如在商品推荐中,某个用户原型代表青年大学生的整体偏好。用户 $u$ 和第 $k$ 个用户原型之间的关联强度可通过如下形式计算:

$$c_{uk} = \mathrm{sim}(\boldsymbol{e}_u, \boldsymbol{p}_k) = 1 + \frac{\boldsymbol{e}_u^{\mathrm{T}} \boldsymbol{p}_k}{\| \boldsymbol{e}_u \| \cdot \| \boldsymbol{p}_k \|}, \tag{1}$$

其中,相似度函数使用平移余弦相似度,取值范围为[0,2];$\| \cdot \|$ 为 $L_2$ 范数。用户 $u$ 与全部 $K$ 个用户原型的关联强度组成的向量,记作 $\boldsymbol{c}_u \in \mathbf{R}^K$

$$\boldsymbol{c}_u = [c_{u1}, c_{u2}, \cdots, c_{uK}]。 \tag{2}$$

各个用户和不同用户原型的关联强度可组成一个 $|U| \times K$ 的矩阵。

用户 $u$ 对物品 $i$ 的预测分数可通过如下双线性函数计算:

$$U\text{-score}(u,i) = \boldsymbol{c}_u^{\mathrm{T}} \boldsymbol{W}^i \boldsymbol{e}_i, \tag{3}$$

其中,$\boldsymbol{W}^i \in \mathbf{R}^{K \times d_2}$、$\boldsymbol{W}^i \boldsymbol{e}_i \in \mathbf{R}^K$ 可视作物品 $i$ 在 $K$ 维用户原型空间中的表示。

2.2.2 物品端原型建模

物品端原型的建模过程与用户端类似。假设物品库可以被 $L$ 个物品原型所代表,每个原型向量 $\boldsymbol{q}_l \in \mathbf{R}^{d_2}$ 代表了某些相似物品的共同特点,一件物品可能具有不止一个物品原型所表示的关键特点。物品 $i$ 和第 $l$ 个物品原型之间的关联强度可定义为

$$c_{il} = \mathrm{sim}(\boldsymbol{e}_i, \boldsymbol{q}_1) = 1 + \frac{\boldsymbol{e}_i^{\mathrm{T}} \boldsymbol{q}_l}{\| \boldsymbol{e}_i \| \cdot \| \boldsymbol{q}_l \|}, \tag{4}$$

其中相似度函数仍使用平移余弦相似度。物品 $i$ 与全部 $L$ 个物品原型之间的关联强度组成的向量,记作 $\boldsymbol{c}_i \in \mathbf{R}^L$

$$\boldsymbol{c}_i = [c_{i1}, c_{i2}, \cdots, c_{iL}]。 \tag{5}$$

各个物品与不同物品原型的关联强度可组成一个 $|I| \times L$ 的矩阵。

用户 $u$ 对物品 $i$ 的预测分数也可通过如下双线性函数计算:

$$I\text{-score}(u,i) = \boldsymbol{c}_i^{\mathrm{T}} \boldsymbol{W}^u \boldsymbol{e}_u, \tag{6}$$

其中,$\boldsymbol{W}^u \in \mathbf{R}^{L \times d_1}$、$\boldsymbol{W}^u \boldsymbol{e}_u \in \mathbf{R}^L$ 可视作用户 $u$ 在 $L$ 维物品原型空间中的表示。

2.2.3 推荐评分

将式(3)、(6)分别计算的用户和物品两端的预测分数相加作为最终的推荐分数,即

$$\hat{y}_{ui} = U\text{-score}(u,i) + I\text{-score}(u,i), \tag{7}$$

这样便于分离用户端和物品端模型对于最终推荐的贡献。

使用采样 softmax 损失(sampled softmax loss)[17] 作为推荐任务损失

$$l_{\mathrm{rec}} = \sum_{(u,i^+) \in Y^+} \left\{ -\hat{y}_{ui^+} + \log\left( e^{\hat{y}_{ui^+}} + \sum_{(u,i^-) \in Y^-} e^{\hat{y}_{ui^-}} \right) \right\}。 \tag{8}$$

其中,$Y^+ = \{(u,i^+) \mid u \in U,\ i^+ \in I,\ y_{ui^+} = 1\}$ 为训练集中已存在交互的用户—物品正例对,$Y^- = \{(u,i^-) \mid u \in U,\ i^- \in I,\ y_{ui^-} = 0\}$ 为未发生交互的用户—物品负例对,$\hat{y}_{ui^+}$ 为正样本 $(u,i^+)$ 的预测分数;$\hat{y}_{ui^-}$ 为负样本 $(u,i^-)$ 的预测分数。对于每个正样本 $(u,i^+)$,训练时采样 $n^-$ 个负样本。

**2.3 原型均匀性约束与重要性均衡**

本节详细介绍添加的 2 种额外损失:均匀性约束和重要性均衡。

2.3.1 原型均匀性约束

在不添加额外约束的情况下,不同原型表示之间的差异性难以保证,不利于刻画整个空间中的全量用户/物品。因此,在用户原型(及物品原型)之间添加均匀性损失,使得不同原型尽量均匀分散到整个空间中,这一点对于模型区分不同用户或物品群体的能力至关重要。

具体来说,本文将用户原型表示向量之间的均匀性损失定义为它们之间的平均成对高斯势(average pairwise Gaussian potential)的对数

$$l_{\text{uni}}^{\text{U}}=\log \underset{k_1\neq k_2}{E} \mathrm{e}^{-2\|\boldsymbol{p}_{k_1}-\boldsymbol{p}_{k_2}\|^2}, \tag{9}$$

其中 $\|\cdot\|^2$ 为 $L_2$ 距离的平方。理论上最小化式(9)会渐近收敛到均匀分布,直观上最小化式(9)可增大任何 2 个原型表示之间的距离。同理,物品原型之间的均匀性约束定义如下:

$$l_{\text{uni}}^{\text{I}}=\log \underset{l_1\neq l_2}{E} \mathrm{e}^{-2\|\boldsymbol{q}_{l_1}-\boldsymbol{q}_{l_2}\|^2}。 \tag{10}$$

本文同时最小化两端原型的均匀性损失:

$$l_{\text{uni}}=l_{\text{uni}}^{\text{U}}/2+l_{\text{uni}}^{\text{I}}/2。 \tag{11}$$

优化以上损失能够增大各个原型与最近原型之间的距离,提升原型表示的差异性。

2.3.2　原型重要性均衡

在用户/物品与原型表示之间的关联强度矩阵之上施加额外约束,在每个用户/物品仅与少量原型保持较强关联的同时,促进各个原型与全量用户/物品的累计关联强度更加均衡,避免浪费模型的表达能力。

定义一个用户原型的重要性为该原型与所有用户的关联强度之和

$$\alpha_k=\sum_{u=1}^{|U|} c_{uk}, \tag{12}$$

各个用户原型的重要性组成的向量记为

$$\boldsymbol{\alpha}=[\alpha_1,\alpha_2,\cdots,\alpha_K], \tag{13}$$

类似地,各个物品原型的重要性可记为

$$\beta_l=\sum_{i=1}^{|I|} c_{il},\quad \boldsymbol{\beta}=[\beta_1,\beta_2,\cdots,\beta_L]。 \tag{14}$$

为了实现重要性均衡,本文借助数学中的变异系数(coefficient of variation)这一概念,其计算方式如下:

$$\text{cv}(\boldsymbol{x})=\frac{\sigma(\boldsymbol{x})}{\bar{\boldsymbol{x}}}, \tag{15}$$

其中,$\sigma(\boldsymbol{x})$ 为 $\boldsymbol{x}$ 的标准差,$\bar{\boldsymbol{x}}$ 为 $\boldsymbol{x}$ 的均值。

于是,用户端原型和物品端原型的重要性均衡损失可分别定义为

$$l_{\text{bal}}^{\text{U}}(\boldsymbol{\alpha})=\text{cv}\ (\boldsymbol{\alpha})^2,\quad l_{\text{bal}}^{\text{I}}(\boldsymbol{\beta})=\text{cv}(\boldsymbol{\beta})^2。 \tag{16}$$

整个模型中原型的重要性均衡损失为二者之和

$$l_{\text{bal}}=l_{\text{bal}}^{\text{U}}(\boldsymbol{\alpha})+l_{\text{bal}}^{l}(\boldsymbol{\beta})。 \tag{17}$$

优化以上损失可促进不同原型之间的负载均衡,使得各个原型在整个推荐系统中的重要性基本相当,避免模型过度依赖某些原型,有利于充分发挥模型的表达能力并提高模型的鲁棒性。

### 2.4　优化目标

本文所提出的 ProtoMF++模型的优化目标包括 3 部分:式(8)中的推荐损失、式(11)中的均匀性损失以及式(17)中的重要性均衡损失:

$$l_{\text{ours}}=l_{\text{rec}}+\lambda_{\text{uni}}l_{\text{uni}}+\lambda_{\text{bal}}l_{\text{bal}}+\lambda_{L_2}\|\boldsymbol{\Theta}\|^2, \tag{18}$$

其中,$\|\boldsymbol{\Theta}\|^2$ 表示所有嵌入向量以及变换矩阵的 $L_2$ 范数,$\lambda_{L_2}$ 调节正则化强度;$\lambda_{\text{uni}}$ 和 $\lambda_{\text{bal}}$ 分别控制均匀性损失和重要性均衡损失的相对强度。

## 3　实验

本章通过实验回答以下研究问题。

**问题 1**　与其他基于原型的推荐方法相比,ProtoMF++是否具有更好的推荐效果?

**问题 2**　本文所添加的 2 个损失是否均有助于提升模型的推荐性能?

**问题 3**　ProtoMF++中的关键超参数如何影响其推荐性能?

**问题 4**　ProtoMF++学习的物品原型的可解释性如何?

### 3.1　实验设置

3.1.1　数据集

Amazon 数据集[18]作为推荐系统中广泛使用的电子商务数据集,涵盖了大量的用户评论、商品介绍以及

二者之间的交互信息。本文选取了该数据集中的 3 个子集 Beauty、Baby、Toys,这些数据集的评分为介于 0 到 5 之间的整数。本文首先对这些数据集进行了以下预处理:将评分高于 3.5 的样本视为正例,低于 3.5 的视为负例,并进行了 5 核过滤(5-core filtering)以去除交互次数较少的部分用户或物品。随后将每个用户已交互的所有物品按照时间戳进行排序,并采用留一法[19]将最后一个交互作为测试集,倒数第二个交互作为验证集,其余交互作为训练集。最终的数据统计汇总如表 2 所示。

表 2 预处理后的数据集统计信息

Table 2 Statistics of the preprocessed dataset

| 数据集 | 用户数/个 | 物品数/个 | 交互数/对 | 稀疏度/% |
|---|---|---|---|---|
| Beauty | 10 553 | 6 086 | 94 148 | 99.85 |
| Baby | 11 761 | 4 731 | 92 823 | 99.83 |
| Toys | 11 268 | 7 309 | 95 420 | 99.88 |

3.1.2 评价指标

本文将测试集中每个用户预留的一个相关物品以及随机采样的 99 个未交互物品按照预测分数降序排列,取前 10 个或 20 个物品推荐给用户。实验时采用命中率(hit ratio)和归一化折损累计增益(normalized discounted cumulative gain, NDCG)2 种指标来评估推荐列表的性能。

3.1.3 基线模型

本文选择以下与原型方法最相关的几个模型进行实验比较。

RBMF[12]是早期引入代表性用户概念的方法。这些代表性用户具有群体中比较典型的偏好行为,通过将任一用户与这些代表性用户进行比较可以推测出该用户的偏好。

ACF[13]首先定义一组锚向量(即原型表示),然后使用同一组锚向量的加权组合来表示用户和物品,并基于此进行推荐。

ECF[14]首先从用户—物品交互信息中挖掘出品味簇,并学习每个用户/物品与各个品味簇的隶属程度,然后根据用户和物品的品味簇隶属程度向量的内积进行推荐。

ProtoMF[4]分别为用户和物品学习两组不同的原型,然后基于用户/物品与对应原型之间的关联强度进行推荐。

3.1.4 实现细节

为了实现公平的比较,本文统一采用 Adam 算法[20]学习模型参数,学习率和权重衰减分别固定为 0.003 和 0.000 3,训练批次大小为 128。为方便起见,实验中设置用户和物品的嵌入维度相等,即 $d_1=d_2=64$;用户原型和物品原型数量相同,即 $K=L=50$。设置迭代轮数为 100 轮,如果连续 10 轮评价指标没有较大变化,就停止训练,防止模型过拟合。对于各个模型的超参数,本文在验证集上通过网格搜索进行调参,最终报告各个模型在测试集上的性能。

### 3.2 整体性能比较(问题 1)

表 3 报告了本文模型与其他基线模型的推荐性能。所有模型中的最优性能和次优性能分别用粗体和下划线突出显示,最后一列表示本文模型相较于最优基线方法的性能提升。分析表中结果可以得到以下重要结论:(1) 在基线模型中,ProtoMF 的效果最佳。该模型在用户和物品两端分别学习原型表示,相较于 ACF 仅在物品一端学习原型表示,这种做法可以丰富模型的表达能力。(2) 本文提出的 ProtoMF++模型在 3 个数据集上的推荐性能均优于所有基线模型。相比于次优模型 ProtoMF,在 Baby 数据集上,HitRatio@10 提升了 4.74%,NDCG@10 提升了 10.64%。这说明了本文所增加的均匀性损失和重要性均衡损失能进一步激发原型的表达能力,从而提供更精准的个性化推荐。

表 3 与基线方法的推荐性能比较

Table 3 Comparison of the recommendation performance with baseline methods

| 数据集 | 评价指标 | RBMF | ACF | ECF | ProtoMF | ProtoMF++ | 相对提升/% |
|---|---|---|---|---|---|---|---|
| Beauty | HitRatio@10 | 0.305 4 | 0.407 4 | 0.385 4 | <u>0.457 0</u> | **0.477 1** | 4.40*** |
| | HitRatio@20 | 0.418 7 | 0.558 1 | 0.461 4 | <u>0.595 8</u> | **0.613 3** | 2.94*** |
| | NDCG@10 | 0.174 9 | 0.231 6 | 0.196 6 | <u>0.281 5</u> | **0.288 8** | 2.59** |
| | NDCG@20 | 0.203 6 | 0.269 6 | 0.273 5 | <u>0.315 6</u> | **0.319 4** | 1.20* |

表 3(续)

| 数据集 | 评价指标 | RBMF | ACF | ECF | ProtoMF | ProtoMF++ | 相对提升/% |
|---|---|---|---|---|---|---|---|
| Baby | HitRatio@ 10 | 0.314 7 | 0.321 5 | 0.316 4 | 0.388 1 | **0.406 5** | 4.74*** |
| | HitRatio@ 20 | 0.446 5 | 0.481 0 | 0.472 6 | 0.534 5 | **0.546 2** | 2.19*** |
| | NDCG@ 10 | 0.177 1 | 0.164 2 | 0.180 6 | 0.205 9 | **0.227 8** | 10.64*** |
| | NDCG@ 20 | 0.210 4 | 0.204 3 | 0.220 6 | 0.252 5 | **0.265 6** | 5.19*** |
| Toys | HitRatio@ 10 | 0.228 6 | 0.325 5 | 0.300 4 | 0.385 9 | **0.414 0** | 7.29*** |
| | HitRatio@ 20 | 0.349 0 | 0.455 4 | 0.401 0 | 0.552 7 | **0.570 7** | 3.26* |
| | NDCG@ 10 | 0.120 9 | 0.208 1 | 0.188 0 | 0.233 1 | **0.245 1** | 5.14** |
| | NDCG@ 20 | 0.151 3 | 0.229 3 | 0.190 8 | 0.256 8 | **0.282 9** | 10.15*** |

注:对 ProtoMF++和 ProtoMF 进行 10 次重复试验和配对 $t$ 检验,$^{*}p<0.05$,$^{**}p<0.01$,$^{***}p<0.001$。

### 3.3　消融研究(问题 2)

#### 3.3.1　消融实验结果

本小节对均匀性损失和重要性均衡损失进行消融实验,结果如表 4 所示,可以得到以下结论:(1)如果去除重要性均衡损失,仅保留原型的均匀性损失,模型的性能有所降低,这证明了重要性均衡损失对于提升推荐效果具有不可忽视的贡献;(2)如果去除原型的均匀性损失,仅保留重要性均衡损失,模型性能下降更明显,这揭示了原型之间的差异性对于保障推荐效果至关重要。

表 4　ProtoMF++额外约束消融实验

Table 4　Ablation study on additional constraints of ProtoMF++

| 方法变种 | 评价指标 | Beauty | Baby | Toys |
|---|---|---|---|---|
| 去除均衡性 | HitRatio@ 10 | 0.456 8 | 0.393 7 | 0.405 6 |
| | NDCG@ 10 | 0.281 6 | 0.220 7 | 0.241 3 |
| 去除均匀性 | HitRatio@ 10 | 0.450 0 | 0.378 3 | 0.393 6 |
| | NDCG@ 10 | 0.270 2 | 0.212 0 | 0.233 0 |
| ProtoMF++ | HitRatio@ 10 | 0.477 1 | 0.406 5 | 0.414 0 |
| | NDCG@ 10 | 0.288 8 | 0.227 8 | 0.245 1 |

#### 3.3.2　均匀性约束消融分析

本小节在 Toys 数据集上进行了实验,结果如图 2 所示。分析 ProtoMF 以及对物品原型施加均匀性约束后 ProtoMF++所学习的物品原型之间的距离分布。具体实验过程如下:首先设置物品原型数量为 50,分别训练两种模型得到各自的物品原型表示向量;然后对于每个物品原型,计算它与其他物品原型之间欧氏距离的最小值;最后绘制如图 2 所示的箱线图,分析各个物品原型与最近物品原型的距离分布。由图 2 可知,ProtoMF++所学的物品原型集合中,每个原型与最近的另一个原型之间的距离都比较大。具体而言,ProtoMF 的物品原型之间最小距离的平均值为 1.398 7,方差为 0.004 2;而 ProtoMF++的物品原型之间最小距离的平均值为 1.479 5,方差为 0.002 6。因此,添加均匀性约束有效地改善了原型在整个空间中分布的均匀程度,增大了原型表示之间的差异性。

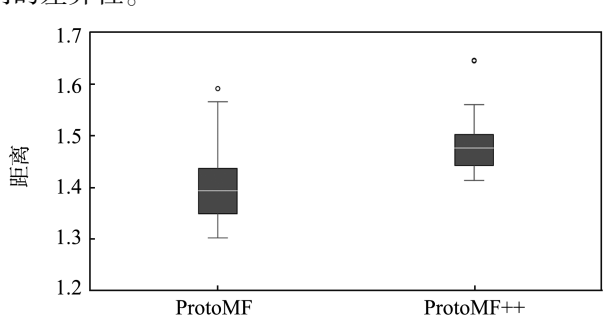


图 2　各个物品原型与最近物品原型之间的距离分布

Fig.2　The distribution of distance between each item prototype and the nearest item prototype

#### 3.3.3　重要性均衡消融分析

本小节在 Toys 数据集上进行了实验,结果如图 3 所示。图 3(a)展示了 ProtoMF 得到的 50 个物品原

型的负载重要性分布,图 3(b)展示了施加重要性均衡损失后 ProtoMF++得到的 50 个物品原型的负载重要性分布。可以观察到,相比于图 3(a),图 3(b)中各个原型的负载更加均衡,即各个原型与全量物品的累计关联强度基本相当。

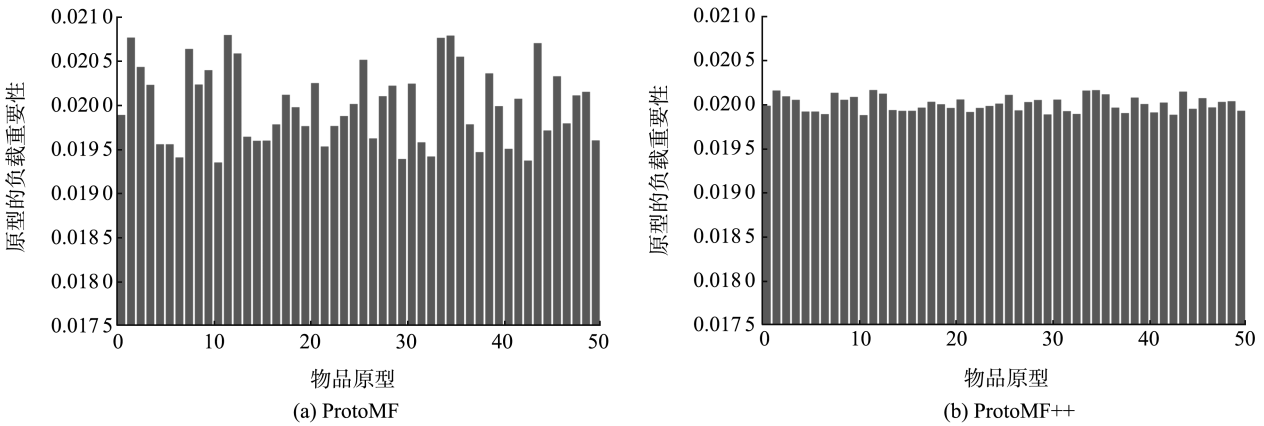


图 3 各个物品原型的负载重要性分布

Fig.3 The distribution of load importance of each item prototype

### 3.4 参数敏感性分析(问题 3)

#### 3.4.1 原型均匀性损失系数

本小节在[0.001,10]范围内调整优化目标式(18)中的超参数 $\lambda_{uni}$,观察 ProtoMF++的性能变化,结果如图 4 所示。可以看出,在不同数据集上,随着超参数 $\lambda_{uni}$的增大,推荐性能均呈现出先上升后下降的趋势。在 Beauty、Baby、Toys 数据集上,当 $\lambda_{uni}$分别取 0.1、1、0.1 时,模型达到了最佳性能。

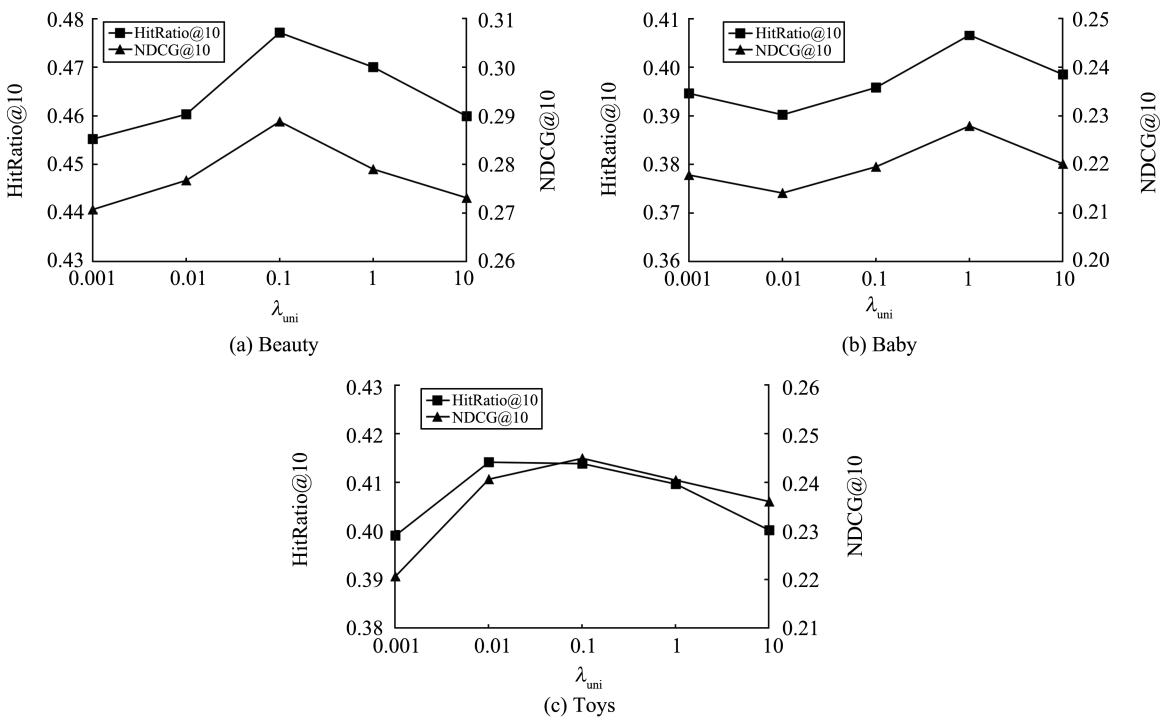


图 4 推荐性能随均匀性损失系数 $\lambda_{uni}$的变化趋势

Fig.4 Variation of the recommendation performance with respect to the uniformity loss coefficient $\lambda_{uni}$

#### 3.4.2 重要性均衡损失系数

本小节在[0.001,10]范围内调整优化目标式(18)中的超参数 $\lambda_{bal}$,观察 ProtoMF++的性能变化,结果如图 5 所示。虽然在以上参数范围内,不同数据集上的变化趋势不尽相同,但控制重要性均衡损失的超参数 $\lambda_{bal}$对推荐性能确有重要影响。以图 5(c)为例,在 Toys 数据集上,随着 $\lambda_{bal}$不断增加,模型性能先升后降,当 $\lambda_{bal}=0.01$ 时 NDCG@10 取得最高值。

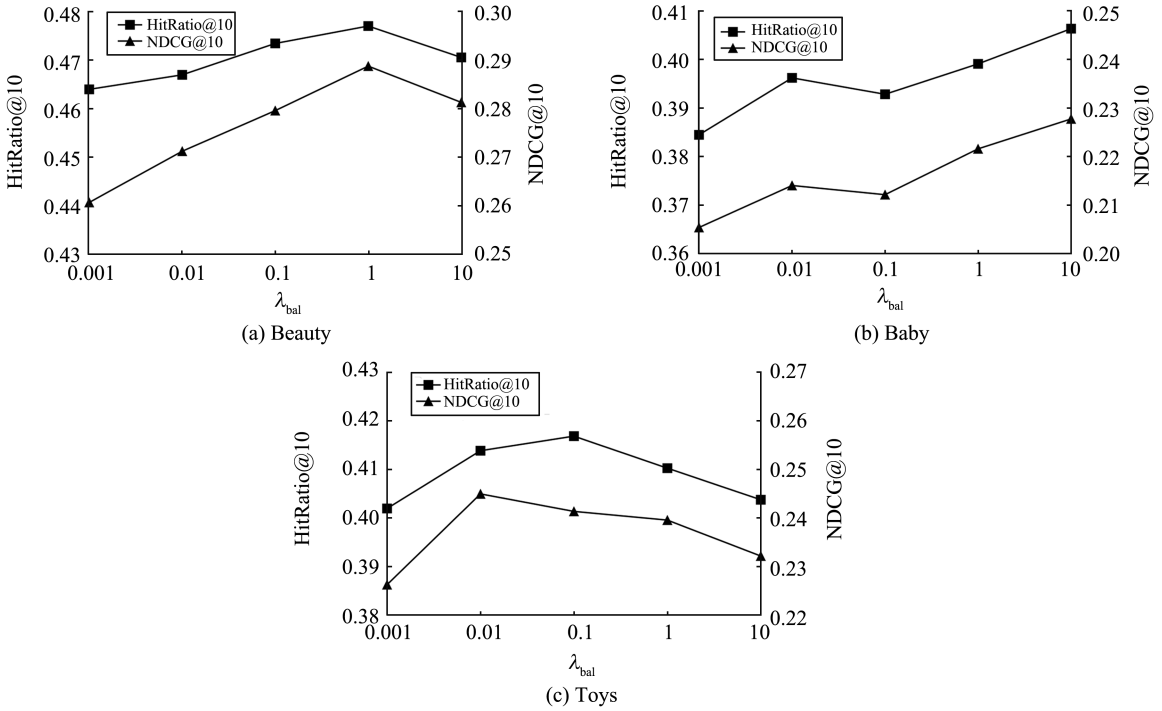


图 5　推荐性能随重要性均衡损失系数 $\lambda_{bal}$ 的变化趋势

Fig.5　Variation of the recommendation performance with respect to the importance balancing coefficient $\lambda_{bal}$

### 3.4.3　负采样数量

本小节在[9,99]范围内调整模型训练时的负采样数量 $n^-$,观察其对推荐性能的影响。ProtoMF++和 ProtoMF 模型的推荐性能变化趋势如图 6 所示。可以发现在 3 个数据集上,随着训练时负样本数量的增多,二者的性能均先上升,然后趋于平稳。当负采样数目相同时,ProtoMF++在绝大部分情况下的推荐性能优于 ProtoMF。为简便起见,遵循 ProtoMF[4] 的实验设置,统一将训练时的负样本数量 $n^-$ 设置为 99。

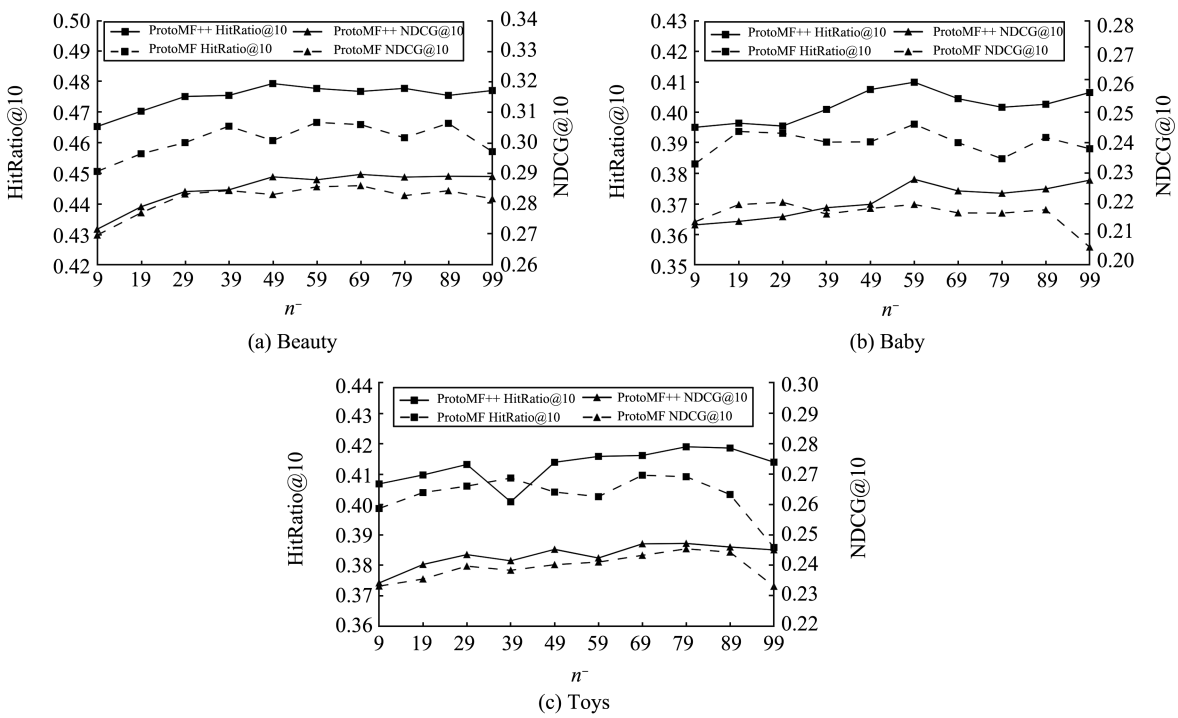


图 6　推荐性能随训练时负样本个数的变化趋势

Fig.6　Variation of the recommendation performance with respect to the number of negative samples during training

### 3.5 案例研究(问题 4)

本节选取在 Toys 数据集上 ProtoMF++所学习的第 10 号和第 20 号物品原型,分别找出与其关联强度最大的 5 个物品,并分析物品所属的类别信息(每个物品包含 2 到 3 个类别),结果如表 5 所示。可以看出,第 10 号物品原型主要代表了与“教育”“实验器材”等相关的物品;第 20 号物品原型则是与“毛绒玩具”“动物图案枕头”等相关的物品。这说明所学习到的物品原型具有一定的可解释性,代表了若干相似物品的共同特点。

表 5 Toys 数据集上与 10 号或 20 号物品原型最相关的物品信息

Table 5 The most relevant items for the 10th or 20th item prototype in the Toys dataset

| 物品原型 | 物品类别 | 最相关的 5 个物品 |
|---|---|---|
| 第 10 号物品原型 | Education/ Science/ Chemistry/ Mathematics | TY Pillow Pal-SPOTTY the Dalmatian<br>Nintendo Wario Plush 12 Inch<br>Maileg Pig Cuddle Pillow, Large<br>TY Classic Plush-MISTY the Seal<br>TY Pillow Pal-WOOF the Dog (Brown Version) |
| 第 20 号物品原型 | Stuffed Animals/ Plush/ Animals & Figures | Molymod Organic Molecular Model Teacher Set<br>Mini Pack Secret Formula Lab Kit<br>Distilled Water, Laboratory Grade, 3.8 L<br>Set of 12 Assorted Color Moon and Stars<br>Faith Mat: St. Francis (Wee Believers W2011-13) |

## 4 总结及展望

本文首先指出现有基于原型的推荐方法忽视了原型表示之间的差异性以及不同原型之间的负载均衡。然后提出了 ProtoMF++模型,通过在原型表示之间添加均匀性损失,提升了原型表示之间的差异性;通过对各个原型与全量用户/物品之间的累计关联强度施加重要性均衡损失,促进了各个原型之间的负载均衡。在 3 个公开数据集上的实验结果证明 ProtoMF++可以取得更优的推荐效果,均匀性损失和重要性均衡损失均有助于提升推荐性能。在未来的研究中,可以在基于原型的推荐方法中引入物品的多模态信息,进一步提高模型的表达能力,同时增强原型的可解释性。

**参考文献:**

[1] ABDOLLAHI B, NASRAOUI O. Explainable matrix factorization for collaborative filtering[C]//Proceedings of the 25th International Conference Companion on World Wide Web. Montréal: ACM, 2016:5-6.

[2] HASE P, CHEN CF, LI O, et al. Interpretable image recognition with hierarchical prototypes[C]//Proceedings of the AAAI Conference on Human Computation & Crowdsourcing. Skamania Lodge: AAAI Press, 2019:32-40.

[3] LI O, LIU H, CHEN C F, et al. Deep learning for case-based reasoning through prototypes: a neural network that explains its predictions[C]//Proceedings of the AAAI Conference on Artificial Intelligence. New Orleans: AAAI Press, 2018:3530-3537.

[4] MELCHIORRE A B, REKABSAZ N, GANHÖR C, et al. ProtoMF: prototype-based matrix factorization for effective and explainable recommendations[C]//Proceedings of 16th ACM Conference on Recommender Systems. Seattle: ACM, 2022: 246-256.

[5] MAO Kelong, ZHU Jieming, WANG Jinpeng, et al. SimpleX: a simple and strong baseline for collaborative filtering[C]// Proceedings of the 30th ACM International Conference on Information & Knowledge Management. New York: ACM, 2021: 1243-1252.

[6] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. BPR: Bayesian personalized ranking from implicit feedback[C]// Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence. Montreal: AUAI Press, 2009:452-461.

[7] ZHOU Chang, MA Jianxin, ZHANG Jianwei, et al. Contrastive learning for debiased candidate generation in large-scale recommender systems[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. New York: ACM, 2021:3985-3995.

[8] PARK S, YOON M, LEE J, et al. Toward a better understanding of loss functions for collaborative filtering[C]//Proceedings

of the 32nd ACM International Conference on Information & Knowledge Management. Birmingham: ACM, 2023:2034-2043.
[9] WANG Chenyang, YU Yuanqing, MA Weizhi, et al. Towards representation alignment and uniformity in collaborative filtering[C]//Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. Washington: ACM, 2022:1816-1825.
[10] SHAZEER N, MIRHOSEINI A, MAZIARZ K, et al. Outrageously large neural networks: the sparsely-gated mixture-of-experts layer[C]//International Conference on Learning Representations. Toulon: OpenReview, 2017.
[11] SAGTANI H, JHAWAR M G, MEHROTRA R, et al. Ad-load balancing via off-policy learning in a content marketplace[C]//Proceedings of the 17th ACM International Conference on Web Search & Data Mining. Merida: ACM, 2024:586-595.
[12] LIU N, MENG X R, LIU C, et al. Wisdom of the better few: cold start recommendation via representative based rating elicitation[C]//Proceedings of the 5th ACM conference on Recommender systems. Chicago: ACM, 2011:37-44.
[13] BARKAN O, HIRSCH R, KATZ O, et al. Anchor-based collaborative filtering[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management. New York: ACM, 2021:2877-2881.
[14] DU Yuntao, LIAN Jianxun, YAO Jing, et al. Towards explainable collaborative filtering with taste clusters learning[C]//Proceedings of the ACM Web Conference 2023. Austin: ACM, 2023:3712-3722.
[15] WANG T Z, ISOLA P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere [C/OL]//Proceedings of the 37th International Conference on Machine Learning, 2020:9929-9939. http://proceedings.mlr.press/v119/wang20k/wang20k.pdf.
[16] YAN Rong, BAO Peng, ZHANG Xiao, et al. Towards alignment-uniformity aware representation in graph contrastive learning[C]//Proceedings of the 17th ACM International Conference on Web Search & Data Mining. Merida: ACM, 2024: 873-881.
[17] JEAN S, CHO K, MEMISEVIC R, et al. On using very large target vocabulary for neural machine translation[C]//Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. Beijing: ACL, 2015:1-10.
[18] MCAULEY J, TARGETT C, SHI J, et al. Image-based recommendations on styles and substitutes[C]//Proceedings of the 38th International ACM SIGIR Conference on Research & Development in Information Retrieval. Santiago: ACM, 2015:43-52.
[19] CREMONESI P, TURRIN R, LENTINI E, et al. An evaluation methodology for collaborative recommender systems[C]//2008 International Conference on Automated Solutions for Cross Media Content & Multi-Channel Distribution. Florence: IEEE, 2008:224-231.
[20] KINGMA D P, BA J L. Adam: a method for stochastic optimization[EB/OL]. (2015-07-23)[2024-10-09].https://arxiv.org/abs/1412.6980.

(编辑:李艺)

(上接第 74 页)

[15] ZHAI C, LAFFERTY J. Model-based feedback in the language modeling approach to information retrieval[C]//Proceedings of the Tenth International Conference on Information and Knowledge Management. Atlanta: ACM, 2001:403-410.
[16] YU H C, XIONG C, CALLAN J. Improving query representations for dense retrieval with pseudo relevance feedback[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management. Seattle: ACM, 2021: 3592-3596.
[17] KINGMA D P, BA J. Adam: a method for stochastic optimization[EB/OL]. (2015-07-23)[2024-09-15]. https://arxiv.org/abs/1412.6980.
[18] JIN Xiaobo, GENG Guanggang, XIE Guosen, et al. Approximately optimizing NDCG using pair-wise loss[J]. Information Sciences, 2018, 453:50-65.
[19] GROS D, HABERMANN T, KIRSTEIN G, et al. Anaphora resolution: analysing the impact on mean average precision and detecting limitations of automated approaches[J]. International Journal of Information Retrieval Research, 2018, 8(3):33-45.

(编辑:李艺)