一种基于 NeuMF 的推荐多样性提升方法

刘浩翰 曲昕彤 贺怀清
(中国民航大学计算机科学与技术学院 天津 300300)

摘 要 针对提升推荐系统中多样性的问题,提出基于 NeuMF 的 NDMF 模型。完善推荐多样性特征,定义复合用户活跃度和项目多样性推荐因子,并配合多层感知机挖掘用户-项目的深层交互;对推荐列表进行重排序,即通过多样性特征对项目的预测分数进行相应降权,进一步提高多样性。实验结果表明,在牺牲较少精确度(牺牲了 0.02 左右)的条件下,该模型可大幅提升推荐的多样性(提升了 0.09 左右),保证用户对推荐结果的满意度。

关键词 NDMF NeuMF 多样性 复合用户活跃度 项目多样性推荐因子 重排序

中图分类号 TP3 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2021.02.035

A RECOMMENDATION DIVERSITY IMPROVEMENT METHOD BASED ON NEUMF

Liu Haohan Qu Xintong He Huaiqing
(*School of Computer Science and Technology, Civil Aviation University of China, Tianjin 300300, China*)

Abstract In order to improve the diversity of recommendation systems, we propose an NDMF model based on NeuMF model. We improved the recommendation diversity characteristics, defined the composite user activity and item diversity recommendation factors, and cooperated with the MLP to mine the deep interaction of the user-project. We reranked the list of recommendation, that is, reduced the prediction scores of the project by the composite user activity and project popularity to further improve the diversity. The experimental results show that the method can greatly improve the recommendation diversity (improve about 0.09) under the condition of less sacrifice (reduce about 0.02), and the user's satisfaction with the recommendation results is guaranteed.

Keywords NDMF NeuMF Diversity Composite user activity Item diversity recommendation factor Reranking

0 引 言

在过去的 20 年,推荐系统的发展逐渐完善,它能将可能被用户喜欢的资讯或项目推荐给用户,从而帮助用户从海量的信息中筛选出需要的特定信息[1]。伴随着用户的个性化需求越来越高,在海量信息中找到并为用户匹配到满足其个性化需求,增强其满意度的信息,成为了专家学者和广大网络用户关心的核心问题。

随着深度学习在推荐领域取得的突破性进展,在整合的多源异构数据中构建贴合用户需求的用户模型,提高推荐系统的性能和精确度,成为解决上述问题的主流解决方案。矩阵分解[2](Matrix Factorization, MF)是最受欢迎的一种协同过滤技术,它使用潜在特征向量来表示用户或项目。但 MF 因为使用一个简单的固定内积,在估计低维潜在空间中用户-项目的复杂交互时会造成非线性建模能力较差的限制。He 等[3]针对这一问题提出了一种神经网络结构模拟用户和项目的潜在特征,设计了基于神经网络的协同过滤通用框架(Neural Collaborative Filtering, NCF),表明 MF 可以被解释为 NCF 的特例即广义矩阵分解(Generalized Matrix Factorization, GMF),并利用多层感知机(Multi-Layer Perceptron, MLP)来赋予 NCF 高水平的非线性建

收稿日期:2019-08-10。**刘浩翰**,副教授,主研领域:图形图像与可视化分析,推荐系统。**曲昕彤**,硕士生。**贺怀清**,教授。

模能力,由此提出神经矩阵分解模型 NeuMF。NeuMF 统一了 MF 在建模用户-项目潜在结构方面的线性建模优势和 MLP 的非线性优势,并且较一般方法提高了推荐的精确度。

然而在实际的推荐环境中,精确度并不是提高用户对推荐项目满意度的唯一标准,推荐列表的多样性也是一种重要指标。多样性反映的是推荐列表中项目种类的差异性,且提供更加多样化的推荐列表不仅可以帮助用户获取新颖的项目,开拓个人偏好空间,还有助于覆盖用户的大部分兴趣点,而且盲目崇拜精确度指标可能会伤害推荐系统,降低用户的满意度。因此,如何实现将多样性融入与深度学习结合的推荐系统中,在损失较少精确度的前提下大幅提高推荐的多样性就成为了可以尝试的研究目标。

目前提高推荐多样性的方法有多种,在典型的协同过滤算法中提高多样性的方法主要有两种:在推荐算法中提高多样性以及在推荐列表上提高多样性。具体代表性方法有:Zhang 等[4]首次将物质扩散理论应用在项目-项目(item-item)网络结构上,推荐方法每一步得分的传递都会除以自己的度,从而导致用户的视野汇聚在那些度较大的节点上,能极大程度地提高推荐的精确性,但在推荐列表多样性上则表现不佳。Premchaiswadi 等[5]基于每个项目的总体多样性效应,提出“总体多样性效应”的重排序推荐方法。Ren 等[6]结合基于用户和项目的协同过滤算法,并为其划分权重,使用不同的多样性方法生成推荐列表,最终达到提升多样性的目的。Ho 等[7]提出 5D 分数的概念,并把推荐分为资源分配和推荐两个阶段,资源分配阶段将推荐机会重新分配给项目,为长尾项目提供机会,并为具有良好口碑的项目留一些特权。

由于推荐多样性与推荐精确度存在着此长彼消的关系,目前提出的用来解决多样性问题的方法多以牺牲精确度来提升多样性为主,尤其是推荐列表排序法会损失较多的精确度,并且没有从根本的用户与项目交互过程中学习多样性特征,只将用户或是项目的某一属性特征与推荐算法结合或是只对推荐列表进行操作。为突破以往方法的限制,本文在 NeuMF 框架基础上提出了 NDMF 模型,利用神经网络的特质,在较少损失推荐精确度的同时提高推荐多样性。

1 NeuMF 框架

NeuMF 框架如图 1 所示。

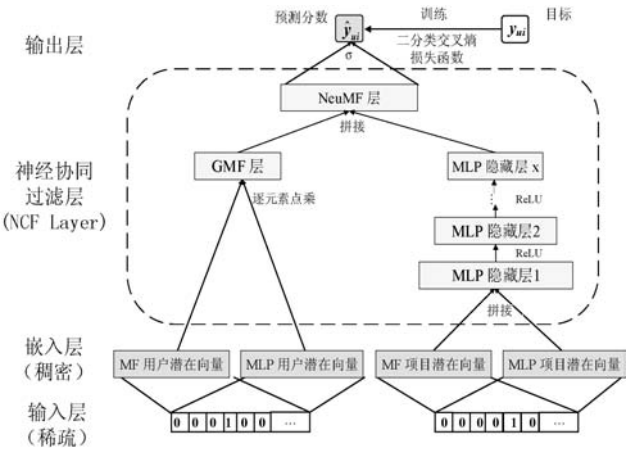

图 1 NeuMF 模型框架

$$\phi^{\mathrm{GMF}} = \hat{y}_{ui}^{G} = a_{\mathrm{out}}(\boldsymbol{h}^{\mathrm{T}}(\boldsymbol{p}_u \otimes \boldsymbol{q}_i)) \tag{1}$$

式中:$\boldsymbol{p}_u$ 和 $\boldsymbol{q}_i$ 分别表示用户和项目的潜在向量(使用隐式反馈得到的交互向量称潜在向量);$\otimes$ 表示向量的逐元素乘积即向量内积;a_{out} 和 $\boldsymbol{h}$ 分别表示输出层的激活函数和连接权,然后将向量映射到输出层得到 GMF 隐层输出的预测分数 $\hat{y}_{ui}^{G}$,这里用 ϕ^{GMF} 表示。右侧采用 MLP 来学习用户-项目的潜在交互,设 a、$\boldsymbol{W}$、$\boldsymbol{b}$ 分别表示激活函数、每层感知机的权重矩阵和偏置向量,σ 表示隐层输出的激活函数,然后得到 MLP 隐层输出的预测分数 $\hat{y}_{ui}^{\mathrm{M}}$:

$$\begin{aligned}
\boldsymbol{z}_1 &= \boldsymbol{\phi}_1(\boldsymbol{p}_u, \boldsymbol{q}_i) = \begin{bmatrix} \boldsymbol{p}_u \\ \boldsymbol{q}_i \end{bmatrix} \\
\boldsymbol{\phi}_2(\boldsymbol{z}_1) &= a_2(\boldsymbol{W}_2^{\mathrm{T}}\boldsymbol{z}_1 + \boldsymbol{b}_2) \\
&\vdots \\
\boldsymbol{\phi}_L(\boldsymbol{z}_{L-1}) &= a_L(\boldsymbol{W}_L^{\mathrm{T}}\boldsymbol{z}_{L-1} + \boldsymbol{b}_L) \\
\phi^{\mathrm{MLP}} &= \hat{y}_{ui}^{\mathrm{M}} = \sigma(\boldsymbol{h}^{\mathrm{T}}\boldsymbol{\phi}_L(\boldsymbol{z}_{L-1}))
\end{aligned} \tag{2}$$

最后在 NCF 框架下结合 GMF 和 MLP,并且为了使得融合模型具有更大的灵活性,GMF 和 MLP 独立学习嵌入向量,并通过隐层输出以结合两个模型得到最终输出 $\hat{y}_{ui}$:

$$\hat{y}_{ui} = \sigma\left(\boldsymbol{h}^{\mathrm{T}}\begin{bmatrix} \phi^{\mathrm{GMF}} \\ \varphi^{\mathrm{MLP}} \end{bmatrix}\right) \tag{3}$$

2 NDMF 模型

本文在 NeuMF 模型保证精确度的基础上在神经网络中融合多样性特征因子,并在推荐过程的最后进行推荐列表重排序以进一步提高列表多样性。保证精确度在可接受的损失范围内提高推荐结果的多样性和用户满意度。由此形成的模型称为多样神经矩阵分解模型(Neural Diversity Matrix Factorization,NDMF)。

2.1　多样性特征因子

首先介绍两个概念。用户活跃度:表示用户产生过交互行为的项目总数;项目流行度:表示对项目产生过交互行为的用户总数。二者都符合长尾分布[8]。

2.1.1　复合用户活跃度 $k(u)$

考虑到用户的活跃度不仅体现在交互过的项目数上,也体现在用户产生交互的项目类别上,为此本文综合考虑这两个影响因素,定义复合用户活跃度如下:

$$k(u) = \omega \times k_0(u) + (1-\omega) \times k_1(u) \quad (4)$$

式中:$k(u)$ 表示用户 u 的复合用户活跃度,由两部分组成:一部分为用户 u 的活跃度,将其简单地归一化处理后记为 $k_0(u)$;另一部分由 kmeans 聚类算法得到,记为 $k_1(u)$。项目集合通过聚类得到 R 个类簇,将用户 u 交互过的项目逐一与 R 个类簇对比得到 r 个子类簇[9],于是 $k_1(u)$ 表示如下:

$$k_1(u) = \frac{r}{R} \quad k_1(u) \in [0,1] \quad (5)$$

再以阈值 ω 调节二者比重,以达到最理想的实验效果。kmeans 聚类算法根据相似性原则将具有较高相似度的项目划分至同一类簇,并且以距离作为项目对间相似性度量的标准。本文利用项目的类别属性将项目 p、q 在 n 维空间上的特征分布表示为 $f_p = \{p1,p2,\cdots\}$,$f_q = \{q1,q2,\cdots\}$ 根据余弦夹角定理将项目 p、q 的相似性定义为:

$$Dis(p,q) = \frac{\sum_1^n (f_{pi} \times f_{qi})}{\sqrt{\sum_1^n f_{pi}^2} \times \sqrt{\sum_1^n f_{qi}^2}} \quad (6)$$

2.1.2　项目多样性推荐因子 $k(i)$

在电商系统中,冷门(长尾)商品的销售总额比实体零售店的商品多很多,甚至会超过热门商品的销售总额,所以长尾商品的销售总额不可忽视。因此,提高推荐的多样性、丰富用户的视野,可以通过挖掘长尾商品来实现。

研究表明长尾分布用单一的函数描述不足以反映其特征,但多个函数的叠加可以达到较好的效果。文献[10]提出一种由 n 个底为 e 的指数函数线性组合(Hyper-Exponential Function,HEF)描述长尾分布函数的方法:

$$k_1(i) = \sum_{j=1}^{n} p_j e^{-\lambda_j k_0(i)} + C \quad (7)$$

式中:$k_0(i)$ 为项目流行度。印桂生等[10]以韦伯分布为例,对 HEF 的效果做实例分析,得出当 $n=2$,即 $k_1(i) = p_1 e^{-\lambda_1 k_0(i)} + p_2 e^{-\lambda_2 k_0(i)} + C$ 时,对长尾分布的描述效果最优,此时 HEF 的系数分别为 $p_1 = 0.467\,1$、$p_2 = 0.346\,8$、$\lambda_1 = 1.546$、$\lambda_2 = 0.063\,98$、$C = 0.184\,9$。

因项目流行度符合长尾分布,本文使用 HEF 来描述项目的长尾分布情况。将项目流行度代入式(6)后得到项目 i 的多样性推荐因子 $k_1(i)$,为方便后续实验操作,使用函数 $f(x) = \log(x+1)$ 将 $k_1(i)$ 进行平滑处理,最后将结果归一化得到 $k(i)$。本文将 $k(i)$ 作为项目 i 的多样性推荐因子,并且与用户复合活跃度结合以提高推荐的多样性,应用在 NDMF 模型中。

2.2　模型结构

NDMF 由改进后的 GMF 与 MLP 两部分组成,结构如图 2 所示。MLP 不仅可以弥补 GMF 单独用向量内积描述用户与项目间的潜在交互特征带来的局限性,还提升了模型的非线性建模能力。

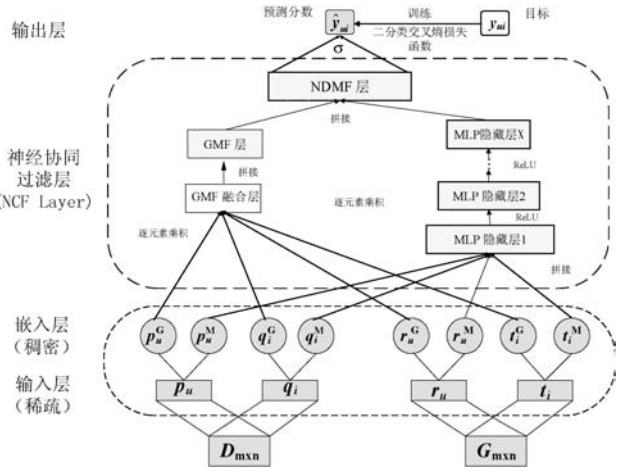

图 2　NDMF 模型结构图

输入层:仅使用一个用户和一个项目的特征作为输入。模型底部包括四个输入特征向量,$\boldsymbol{p}_u$ 和 $\boldsymbol{q}_i$ 为用户 u 和项目 i 的交互特征向量,由用户-项目交互矩阵 $\boldsymbol{D}_{m\times n}$ 得到,$\boldsymbol{r}_u$ 和 $\boldsymbol{t}_i$ 为用户 u 的复合活跃度特征向量和项目 i 的多样性特征向量,由 $k(u)$ 和 $k(i)$ 融合的矩阵 $\boldsymbol{G}_{m\times n}$ 而得。

嵌入层:它是一个全连接层,用来将输入层的四个高维稀疏特征向量映射成低维的稠密向量,为了使 NDMF 具有最大限度的灵活性,让 GMF 与 MLP 独立学习嵌入,并且以上角标 G 和 M 区分二者的输入。

NCF 层:将嵌入层得到的嵌入向量送入各自的 NCF 层,在 GMF 部分定义 NCF 层的映射函数为:

$$\begin{aligned} \boldsymbol{\phi}_1(\boldsymbol{p}_u^G, \boldsymbol{q}_i^G) &= \boldsymbol{p}_u^G \otimes \boldsymbol{q}_i^G \\ \boldsymbol{\phi}_1(\boldsymbol{r}_u^G, \boldsymbol{t}_i^G) &= \boldsymbol{r}_u^G \otimes \boldsymbol{t}_i^G \\ \boldsymbol{\phi}_2(\boldsymbol{p}_u^G, \boldsymbol{q}_i^G, \boldsymbol{r}_u^G, \boldsymbol{t}_i^G) &= (\boldsymbol{p}_u^G \otimes \boldsymbol{q}_i^G) \oplus (\boldsymbol{r}_u^G \otimes \boldsymbol{t}_i^G) \end{aligned} \quad (8)$$

式中:⊗代表向量的逐元素点乘;⊕代表将用户-项目点乘得到的交互特征向量与用户活跃度-项目长尾因子点乘得到的多样性特征向量进行拼接(concatenate)。两个特征向量融合后,交互特征与多样性特征

的联合即达到提高推荐多样性的目的。在 MLP 部分定义 NCF 层结构：

$$\boldsymbol{m}_1 = \boldsymbol{\phi}_1(u,i) = \begin{bmatrix} \boldsymbol{p}_u^{\mathrm{M}} \\ \boldsymbol{q}_i^{\mathrm{M}} \\ \boldsymbol{r}_u^{\mathrm{M}} \\ \boldsymbol{t}_i^{\mathrm{M}} \end{bmatrix} \tag{9}$$

$$\boldsymbol{\phi}_2(\boldsymbol{m}_1) = a_2(\boldsymbol{W}_2^{\mathrm{T}}\boldsymbol{m}_1 + \boldsymbol{b}_2)$$

$$\vdots$$

$$\boldsymbol{\phi}_L(\boldsymbol{m}_{L-1}) = a_L(\boldsymbol{W}_L^{\mathrm{T}}\boldsymbol{m}_{L-1} + \boldsymbol{b}_L)$$

式中：a、$\boldsymbol{b}$、$\boldsymbol{W}$ 分别表示激活函数、偏置向量和权重矩阵。本文中激活函数选择 ReLU(Rectified Linear Unit)，因为它被证明不会导致过饱和，且实验结果表明 ReLU 的效果较 Sigmoid 和双曲正切函数更好。最后将改进后的 GMF 与 MLP 进行融合得到 NDMF，两部分从嵌入层开始独立学习，在最后的隐藏输出层 NDMF 层进行融合，方案如下：

$$\boldsymbol{\phi}^{\mathrm{GMF}} = \begin{bmatrix} \boldsymbol{p}_u^{\mathrm{G}} \otimes \boldsymbol{q}_i^{\mathrm{G}} \\ \boldsymbol{r}_u^{\mathrm{G}} \otimes \boldsymbol{t}_i^{\mathrm{G}} \end{bmatrix}$$

$$\boldsymbol{\phi}^{\mathrm{MLP}} = a_L(\boldsymbol{W}_L^{\mathrm{T}}(\boldsymbol{m}_{L-1} + \boldsymbol{b}_L)) \tag{10}$$

$$\hat{y}_{ui} = a\left(\boldsymbol{h}^{\mathrm{T}}\begin{bmatrix} \boldsymbol{\phi}^{\mathrm{GMF}} \\ \boldsymbol{\phi}^{\mathrm{MLP}} \end{bmatrix}\right)$$

输出层：输出用户 u 对项目 i 的预测分数 $\hat{y}_{ui}$。$\hat{y}_{ui}$ 的训练过程是通过最小化 $\hat{y}_{ui}$ 和目标值 y_{ui} 间的逐点损失[11]。

NDMF 模型使用隐式反馈[12]，但 y_{ui} 为 1 并不代表用户 u 喜欢 i，同样 y_{ui} 为 0 也不意味着用户 u 不喜欢 i。

$$y_{ui} = \begin{cases} 1 & \text{用户 } u \text{ 和项目 } i \text{ 有交互} \\ 0 & \text{用户 } u \text{ 和项目 } i \text{ 没有交互} \end{cases} \tag{11}$$

隐式反馈上的推荐问题可以形式化为估计样本中未曾有过交互的项目的分数问题，并且可被抽象为学习函数 $\hat{y}_{ui} = f(u,i \mid \boldsymbol{\Theta})$。学习函数 f 用来得到模型预测分数 $\hat{y}_{ui}$，其中：$\boldsymbol{\Theta}$ 表示模型参数，一般由损失函数来估计；f 表示表示模型参数映射到预测分数的学习函数。本文提出的 NDMF 模型将利用神经网络，参数化学习函数 f，从而估计 $\hat{y}_{ui}$。

2.3 输入数据预处理

输入数据预处理分为两部分，采用两种自定义的 one-hot 编码方式得到用户-项目交互矩阵 $\boldsymbol{D}_{m\times n}$ 与复合用户活跃度-项目多样性推荐因子交互矩阵 $\boldsymbol{G}_{m\times n}$，然后根据 $\boldsymbol{D}_{m\times n}$ 与 $\boldsymbol{G}_{m\times n}$ 得到 4 个稀疏的底层输入向量。NDMF 模型输入包括 4 个特征向量：用户特征向量，项目特征向量，复合用户活跃度特征向量，项目多样性推荐因子特征向量。

首先为了得到 $\boldsymbol{D}_{m\times n}$，对用户和项目特征进行 one-hot 编码处理，即以用户数 M 和项目数 N 为横纵阶数生成矩阵，将用户与项目的交互结果(0 或 1)填充到矩阵中，例：若用户 u 对项目 i 有过交互则矩阵中对应位置为 1，否则为 0。然后为了得到 $\boldsymbol{G}_{m\times n}$，需要将复合用户活跃度 $k(u)$ 和项目多样性推荐因子 $k(i)$ 进行特征结合，将 $k(u)$ 和 $k(i)$ 保留相同小数位后进行等倍数扩大化为整数，将 $k(u)$ 化为二进制并生成 $m\times k$ 阶矩阵，对 $k(i)$ 进行相同处理生成 $k\times n$ 阶矩阵，其中 k 为 $k(u)$ 与 $k(i)$ 中最大数值所需二进制化的位数。最后将得到的两个矩阵相乘得到 $m\times n$ 阶矩阵 $\boldsymbol{G}_{m\times n}$，如图 3 所示。

$$\begin{matrix} \\ k(u_1) \\ k(u_2) \\ \vdots \\ k(u_m) \end{matrix}\begin{matrix} k \\ \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & \cdots \\ 0 & 0 & 0 & 0 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ 1 & 0 & 0 & 0 & 1 & \cdots \end{bmatrix} \end{matrix} \times \begin{matrix} \begin{matrix} k(i_1) & k(i_2) & \cdots & k(i_n) \end{matrix} \\ k \begin{bmatrix} 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & & \vdots \end{bmatrix} \end{matrix} = \boldsymbol{G}_{m\times n}$$

图 3 复合用户活跃度矩阵与项目多样性因子矩阵

$\boldsymbol{G}_{m\times n}$ 中的每一项都结合了用户 u 的复合活跃度以及项目 i 的多样性推荐因子，因此其表征的是用户-项目对的多样性特征。多样性推荐因子本身可提高冷门项目的推荐权重，而复合用户活跃度则体现用户接受冷门项目的能力，二者相乘起到调节推荐因子所占比重的作用，即复合活跃度较高的用户相比复合活跃度低的用户更能够承受多样性推荐因子带来的冷门项目所占比重的大幅度提升。

2.4 损失函数

在 2.2 节中利用 NDMF 模型得到学习函数 f 后，使用逐点学习方法中的均方差进行回归来学习模型参数 $\boldsymbol{\Theta}$，并且考虑到隐式反馈的性质即目标值 y_{ui} 是二进制值 1 或 0，本文将预测分数 $\hat{y}_{ui}$ 表征为项目 i 和用户 u 存在交互的可能性大小。为了测量预测结果较实际值的偏差并且实现上述表达，使用概率函数作为输出层的激活函数将输出结果 $\hat{y}_{ui}$ 控制在[0,1]之间，并定义 NDMF 的损失函数(目标函数)：

$$L = -\sum_{(u,i)\in y\cup y^-} y_{ui}\log\hat{y}_{ui} + (1-y_{ui})\log(1-\hat{y}_{ui}) \tag{12}$$

式(12)与二分类交叉熵损失函数[13]是相同的，原因在于我们把隐式反馈的输出当作一个二分类问题进行处理，并且使用随机梯度下降法最小化目标函数。其中 y^- 是消极实例(负反馈)，从与用户无交互的项目中进行均匀采样得到负反馈，并且可控制采样比。

2.5　候选推荐列表重排序

NDMF 模型最后为测试集中每个用户-项目对计算出预测分数,并且针对每个用户将其对应的所有项目的预测分数 $\hat{y}_{ui}$ 降序排列,得到该用户的 TopN 推荐列表。为了进一步提高最终推荐列表的多样性,本文对 NDMF 模型预测得到的项目候选推荐列表进行重排序,即把候选推荐列表中的某些项目根据给定的标准调整其得分值,从而有更大(或更小)的概率出现在最终的 TopN 列表中。

模型最后预测结果 $\hat{y}_{ui}$ 代表用户 u 与项目 i 存在交互的概率,那么在重排序阶段则利用 $k(u)$ 与 $k(i)$ 对该概率进行相应调整[14],方案如下:

$$Score_{\text{new}}(u,i) = \frac{Score(u,i)}{\log(k(i)+1)^{stand(\log(k(u)+1))}} \quad (13)$$

式中; $Score(u,i)$ 代表模型预测的分数 $\hat{y}_{ui}$; 用函数 $\log(x+1)$ 对 $k(u)$ 与 $k(i)$ 进行统一的平滑处理;*stand* 表示使用极大极小值方法进行标准化。这样当给定用户 u 时,其复合用户活跃度 $k(u)$ 不变,所以分母的指数项不变,预测分数 $Score(u,i)$ 根据项目多样性推荐因子 $k(i)$ 大小被相应的降权。由于 log 函数的特性,当候选推荐列表中 $k(i)$ 越小,降权作用越小,即 $Score_{\text{new}}(u,i)$ 降低得越少。同理 $k(i)$ 越大,降权作用越大,$Score_{\text{new}}(u,i)$ 降低得越多,导致该项目在列表中后移。而对于不同的用户时,其 $k(u)$ 越大,分母指数项越接近 1,提高了由 $k(i)$ 带来的降权幅度,反之亦然。由此,重排序后长尾(冷门)项目的预测分数受到的降权幅度小,在推荐列表中前移,提高了其入选 TopN 列表的概率,进而提高推荐结果的多样性。简言之,重排序方法综合复合用户活跃度与项目多样性推荐因子,对高活跃度的用户提高其对多样性的容忍度,对低活跃度用户基本保持不变,以免过多损失推荐的精确度。

3　算法描述

本文算法步骤描述如下:

输入: $\boldsymbol{D}_{m\times n}$ 和 $\boldsymbol{G}_{m\times n}$。

输出:评价指标 HR、NDCG 和 ILS、损失函数 loss。

Step1　开始。

Step2　训练特征并预测结果。

for 用户 u_1 to 用户 u_m

for 项目 i_1 to 项目 i_n

调用训练后的 NDMF 模型,得到当前用户-项目对的预测分数 $\hat{y}_{ui}$,将预测结果按照当前用户编号排列,得到当前用户的候选推荐列表 $result_{ux}$(m 个用户对应 m 个列表,分别存入 $result_{u1}$,$result_{u2}$,…,$result_{um}$)。

Step3　候选推荐列表重排序。

for 用户 u_1 to 用户 u_m

利用式(13)对当前用户候选推荐列表 $result_{ui}$ 中的预测分数进行调整,进而达到对 $result_{ui}$ 中的项目分数重排序的目的。

Step4　利用 $result_{ui}$ 计算并输出三个评估指标 HR、NDCG 和 ILS,并代入式(12)计算损失值 loss。

Step5　结束。

4　实　验

4.1　实验数据集

本文实验数据集使用 MovieLens 和 Pinterest 两个数据集:

(1) MovieLens 显式反馈数据集。该数据集广泛应用于评估协同过滤算法,虽然它是显式反馈数据集,但我们要从显式反馈中学习隐式信息。为此,将其转换为隐式数据,其中每条数据被标记为 0 或 1 表示用户是否对该项进行评级。

(2) Pinterest 隐式反馈数据集。该数据集用于评估基于内容的图像推荐算法,类似于朋友圈点赞,原始数据庞大且稀疏。例如,超过 20% 的用户只点赞过一次,难以用来评估协同过滤算法。因此,过滤数据集,仅保留赞过 20 次以上的用户。处理后得到了包含 55 187个用户和 1 445 621 个项目交互的数据的子集。两数据集数据数量如表 1 所示。

表 1　数据集

数据集	交互数量	项目数	用户数
MovieLens1M	1 00 209	3 706	6 040
Pinterest	1 445 621	9 911	55 187

4.2　评价方案

本文采用了与 NeuMF 相同的评估方法——留一法[15](leave-one-out):对于每个用户,使用其最近的一次交互作为测试集,并将其余数据作为训练集。由于在评估过程中为每个用户排列所有项目花费的时间太多,所以随机抽取 100 个与用户没有过交互的项目,将测试项目排列在这 100 个项目中。

为了衡量推荐结果的精确度与多样性,本文采用命中率(Hit Radio, HR)、折损累计增益(Normalized Discounted Cumulative Gain, NDCG)和列表内部多样性

(Intra-list Similarity,ILS)进行评估。HR 衡量测试项目是否存在于 TopN 列表中;NDCG 用来衡量测试项目在 TopN 列表中的位置,位置越靠前则增益越高,精确度越高;ILS 针对单个用户的推荐列表,通过计算项目之间的相似度进而衡量列表的多样性,推荐列表中项目越不相似,ILS 越小,推荐结果的多样性越好。为了便于观察实验结果,定义多样性评价指标 $Div = 1 - ILS$,即 Div 越大,多样性越好。

4.3 实验过程

4.3.1 类簇个数确定

在聚类算法中类簇个数 K 对聚类的结果有直接影响,本文使用轮廓系数法[16] 对不同类簇得到的聚类结果进行评估。项目 i 的轮廓系数 $s(i)$ 定义为:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{14}$$

式中:$b(i)$ 表示项目 i 的簇内相似度,即与簇内其他项目的距离的平均值,$b(i)$ 越小,说明项目 i 越应该被聚类到该簇;$a(i)$ 表示项目 i 的簇间相似度,即与不同簇项目的平均距离的平均值,$a(i)$ 越大,说明项目越不属于其他簇。由此可得结论,$s(i)$ 接近 1,则说明项目 i 聚类合理;$s(i)$ 接近 -1,则说明项目 i 更应该分类到另外的簇。为了度量整个聚类的质量,求得所有项目的平均轮廓系数:

$$\bar{s}(i) = \frac{1}{n}\sum_{i=1}^{n} s(i) \tag{15}$$

式中:n 为项目数;$\bar{s}(i)$ 值越大聚类效果越好。由 MovieLens 和 Pinterest 两个数据集规模及其中项目类别数目等特征将类簇个数分别限定在[10,80]和[40,110]之间,并且对于区间内的每一个 K 值重复执行 20 次来避免 k-means 算法的随机性,最后对于每个 K 得到的 20 个 $\bar{s}(i)$ 平均值 $avg(\bar{s}(i))$,在两个数据集上的结果如表 2、表 3 所示。

表 2 在 MovieLens 上不同类簇数的聚类效果

簇数	$avg(\bar{s}(i))$	簇数	$avg(\bar{s}(i))$
28	0.547 9	33	0.604 8
29	0.563 1	34	0.594 3
30	0.578 2	35	0.595 7
31	0.586 7	36	0.579 1
32	0.603 2	37	0.566 2

表 3 在 Pinterest 上不同类簇数的聚类效果

簇数	$avg(\bar{s}(i))$	簇数	$avg(\bar{s}(i))$
74	0.514 0	79	0.542 9
75	0.518 2	80	0.535 7

续表 3

簇数	$avg(\bar{s}(i))$	簇数	$avg(\bar{s}(i))$
76	0.525 4	81	0.520 6
77	0.529 2	82	0.507 9
79	0.534 6	83	0.493 1

表 2 与表 3 分别列出两个数据集聚类效果较好时的类簇情况,可以看出在 MovieLens 上类簇为 33 时 $avg(\bar{s}(i))$ 值最大,也就是聚类的质量最好,同理在 Pinterest 上类簇数为 79 时 $avg(\bar{s}(i))$ 值最大,所以在下述实验中将在两个数据集上分别使用类簇数 33 和 79 进行实验。

4.3.2 模型参数设置

模型 NDMF 是基于 NCF 框架提出的,因此需要在 NCF 上调整参数来达到更高的效果。首先使用式(12)二类交叉熵损失函数学习模型,取正负采样比 1:4,对于模型参数的初始化选择高斯随机分布,并用 Adam 作为学习率自适应优化算法,它通过对参数进行频繁和大幅度的更新来适应每个参数的学习速率,Adam 方法在 NeuMF 和 NDMF 模型上的收敛速度都比普通 SGD(Stochastic Gradient Descent)快,缓解了调整学习率的难度。训练批次大小和学习速率通过测试选择最优的 256 和 0.001,并且由于 NCF 的结构特性,其隐藏输出层即最后一层隐藏层决定了模型的性能,所以将其作为重要预测因素并使用[8,16,32,64]的因素大小作为模型的评估标准,若预测因素大小为 8,则 NCF 层即结构为 32 - 16 - 8,分别为输入层,嵌入层和隐层输出的大小。由于深层的网络结构对于推荐任务也存在影响,经过实验验证选择隐藏层数为 3 的 MLP。

4.3.3 不同方法性能对比

将 NDMF 与 GMF、MLP、NeuMF 三种以 NCF 框架为基础的方法在两个真实数据集上进行对比,分别在隐层输出大小为[8,16,32,64]上进行实验,结果如图 4 所示。在 MovieLens 数据集上,NDMF 的精确度指标 HR 和 NDCG 较 NeuMF 相差 0.02 左右,但与 GMF 和 MLP 相比相差不大,且上升趋势保持平稳,其中 NDCG 降低幅度大于 HR 的原因在于得到预测结果后对其进行了重排序,可能导致目标项目的位置后移。如果目标项目被移出 TopN 列表,那么 HR 和 NDCG 都会降低,如果没有移出 TopN 列表那么只有 NDCG 会降低。在多样性评价指标 ILS 上可以明显看出,NDMF 的列表内部多样性明显高于其他三种方法,其中高于 NeuMF 方法 0.09 左右,并且随着精确度的上升会出现下降趋势。对于 Pinterest 数据集而言,整体趋势同上,

可看出 NDMF 方法精确度较 NeuMF 方法降低幅度稍小,并且趋势平稳,且在多样性上的优势突出。由此,本文提出的 NDMF 方法在 NCF 框架中使用神经网络学习了构成的多样性特征,得到的实验结果显示,在精确度的损失在可接受范围内,并且不低于一般协同过滤推荐算法的前提下,换来了推荐列表多样性的大幅提升。

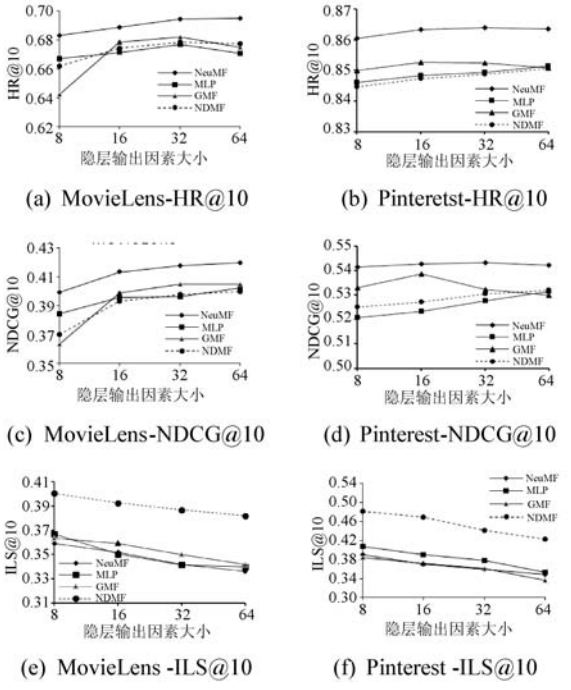

图 4　在两个数据集上三个评价指标的性能比较

4.3.4　对比实验

以隐层输出大小为 8,从不同角度进行以下对比实验。在 2.1.1 节中,以两种不同的方法计算了用户的活跃度,并以 ω 控制二者的贡献度,图 4 是以 $\omega = 0.5$ 得出的实验结果。改变两种用户活跃度所占比重并观察实验结果的变化,如表 4 所示,随着 ω 的增大精确度有较小幅度的波动而多样性则小幅度下降,ω 增大表示聚类得到的用户活跃度 $k_1(u)$ 占比逐渐增多,相比于单纯由项目被评价的次数得到的活跃度 $k_0(u)$,$k_1(u)$ 携带的多样性特征更加明显,带来的增益更多。

表 4　ω 不同时 NDMF 方法在两个数据集上的对比实验结果

ω	MovieLens			Pinterest		
	HR	NDCG	ILS	HR	NDCG	ILS
0.3	**0.658 3**	**0.358 1**	**0.479 5**	**0.843 6**	**0.519 8**	**0.409 9**
0.4	0.659 1	0.362 7	0.474 6	0.842 8	0.502 9	0.406 7
0.5	0.663 9	0.369 7	0.468 6	0.844 6	0.524 9	0.401 3
0.6	0.665 9	0.370 3	0.466 2	0.845 0	0.525 0	0.398 4
0.7	0.666 4	0.371 6	0.458 7	0.845 4	0.525 1	0.391 5

最后选取 TopN 列表的长度为 10 和 15 在两个数据集进行对比,随着推荐列表长度增加,目标项目存在于列表的可能性或是其在推荐列表的排名得到提升,且由于 Pinterest 数据集的稀疏度较大,其精确度与多样性的变化比 MovieLens 要小。可以看出,在 NeuMF 中 ILS 的涨幅远小于 NDMF,这是由于 NDMF 不仅新增了用户和项目的多样性特征,还在 MLP 的非线性内核从数据中学习交互函数时添加了复合用户活跃度与项目多样性推荐因子,具体实验结果见表 5 和表 6。

表 5　Top10 下 NeuMF 与 NDMF 实验对比

数据集	模型	Top10		
		HR	NDCG	ILS
MovieLens	NeuMF	0.681 6	0.396 2	0.389 6
	NDMF	0.663 9	0.369 7	0.468 6
Pinterest	NeuMF	0.860 5	0.541 3	0.359 2
	NDMF	0.844 6	0.524 9	0.401 3

表 6　Top15 下 NeuMF 与 NDMF 实验对比

数据集	模型	Top15		
		HR	NDCG	ILS
MovieLens	NeuMF	**0.737 9**	**0.435 5**	**0.386 5**
	NDMF	**0.721 8**	**0.421 7**	**0.508 1**
Pinterest	NeuMF	**0.882 5**	**0.567 1**	**0.358 8**
	NDMF	**0.878 6**	**0.543 4**	**0.469 7**

5　结　语

针对如何提高多样性,本文提出 NDMF 模型。其在 GMF 中以用户-项目对为单位进一步学习复合用户活跃度与多样性推荐因子结合得到的特征,同时用 MLP 学习用户-项目对间的潜在交互关系。NDMF 不仅通过新特征的学习提高了推荐的多样性,而且在统一了用户-项目潜在结构方面 MF 的线性建模优势以及 MLP 的非线性建模优势的基础上保证了推荐的精确度。最后在得到的预测结果上进行重排序,更加确保了多样性的提升。实验结果证明,精确度的损失在可接受的范围内,且与精确度的损失相比多样性得到更大幅度的提升。

近两年,阿里巴巴提出了基于 DNN 模型的深度兴趣网络和它的进化版,基于用户多样性以及用户历史数据的部分有效性,在其中设计了“兴趣层”充分挖掘用户历史数据中的信息来提升 CTR 预估的性能。基于该研究,未来我们将尝试通过在 NDMF 模型中的

MLP 部分添加由注意力机制构成的“兴趣层”,来探索用户和项目的深度交互,进而保证推荐的精确性和多样性的同时提升。

参 考 文 献

[1] Ricci F, Rokach L, Shapira B. Recommender systems handbook[M]. Springer, 2011.

[2] Wang Y, Wu L, Lin X, et al. Multiview spectral clustering via structured Low-Rank matrix factorization[J]. IEEE Transactions on Neural Networks and Learning Systems, 2018, 29(10): 1 - 11.

[3] He X, Liao L, Zhang H, et al. Neural collaborative filtering[C]//26th International Conference on World Wide Web, 2017: 173 - 182.

[4] Zhang Y C, Medo M, Ren J, et al. Recommendation model based on opinion diffusion[J]. Europhysics Letters, 2007, 80(6): 417 - 429.

[5] Premchaiswadi W, Poompuang P, Jongswat N, et al. Enhancing Diversity-Accuracy technique on User-Based Top-N recommendation algorithms[C]//37th IEEE Annual Computer Software and Applications Conference Workshops, 2013: 403 - 408.

[6] Ren C, Zhu P, Zhang H. A new collaborative filtering technique to improve recommendation diversity[C]//IEEE International Conference on Computer, 2017.

[7] Ho Y C, Chiang Y T, Hsu J Y J. Who likes it more?: mining worth-recommending items from long tails by modeling relative preference[C]//7th ACM international conference on Web search and data mining, 2014: 253 - 262.

[8] Krishnan A, Sharma A, Sankar A, et al. An adversarial approach to improve Long-Tail performance in neural collaborative filtering[C]//27th ACM International Conference on Information and Knowledge Management, 2018: 1491 - 1494.

[9] 邓明通,刘学军,李斌. 基于用户偏好和动态兴趣的多样性推荐方法[J]. 小型微型计算机系统,2018,39(9):2029 - 2034.

[10] 印桂生,张亚楠,董红斌,等. 一种由长尾分布约束的推荐方法[J]. 计算机研究与发展,2013,50(9):1814 - 1824.

[11] He X, Zhang H, Kan M Y, et al. Fast matrix factorization for online recommendation with implicit feedback[C]//39th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2016: 549 - 558.

[12] Ostuni V C, Noia T D, Sciascio E D, et al. Top-N recommendations from implicit feedback leveraging linked open data[C]//7th ACM Conference on Recommender Systems, 2013: 85 - 92.

[13] Xue H J, Dai X, et al. Deep matrix factorization models for recommender systems[C]//26th International Joint Conference on Artificial Intelligence, 2017: 3203 - 3209.

[14] 师扬波. 面向多样性的推荐算法研究[D]. 成都:电子科技大学,2018.

[15] Koren Y. Factorization meets the neighborhood: A multifaceted collaborative filtering model[C]//14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2008: 426 - 434.

[16] Jain R, Koronios A. Innovation in the cluster validating techniques[J]. Fuzzy Optimization and Decision Making, 2008, 7(3): 233 - 241.

(上接第 157 页)

[12] 张春祥,栾博,高雪瑶,等. 基于句法分析的汉语词义消歧[J]. 计算机应用研究,2014,31(1):40 - 42.

[13] Han L, Deng X J, Wu G H. A knowledge-based word sense disambiguation algorithm utilizing syntactic dependency relation[C]//2017 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). IEEE, 2017: 58 - 90.

[14] 王少楠,宗成庆. 一种基于双通道 LDA 模型的汉语词义表示与归纳方法[J]. 计算机学报,2016,39(8):1652 - 1666.

[15] Bordogna G, Pasi G. Hierarchical-hyperspherical divisive fuzzy C-means(H2D-FCM) clustering for information retrieval[C]//2009 IEEE/WIC/ACM International Joint Conferences on Web Intelligence & Intelligent Agent Technologies. IEEE, 2009: 614 - 621.

[16] 哈工大社会计算与信息检索研究中心. 语言技术平台云: LTP[EB/OL]. [2014 - 11 - 30]. http://www.ltp-cloud.com/.

[17] Che W X, Li Z H, Liu T. LTP: A chinese language technology platform[C]//Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations. ACM, 2010: 13 - 16.

[18] 施聪莺,徐朝军,杨晓江. TFIDF 算法研究综述[J]. 计算机应用,2009,29(S1):167 - 170,180.

[19] 余冲,李晶,孙旭东,等. 基于词嵌入与概率主题模型的社会媒体话题识别[J]. 计算机工程,2017,43(12):184 - 191.

[20] 刘赞,陈西宏,刘进,等. 基于模糊 C 类均值聚类的信源数估计方法[J]. 系统工程与电子技术,2019,41(2):244 - 248.

[21] 王展,杜平安,李杨,等. 基于 FCM 聚类的示温漆图像分割算法[J]. 航空动力学报,2018,33(3):604 - 610.

[22] Bezdek J C, Ehrlich R, Full W. FCM: The fuzzy C-means clustering algorithm[J]. Computers & Geosciences, 1984, 10(2/3): 191 - 203.

[23] 何径舟,王厚峰. 基于特征选择和最大熵模型的汉语词义消歧[J]. 软件学报,2010,21(6):1287 - 1295.