

融合用户偏好和多交互网络的协同过滤算法^①



刘锦涛, 谢颖华

(东华大学 信息科学与技术学院, 上海 201620)

通讯作者: 刘锦涛, E-mail: 171633195@qq.com

摘要: 推荐系统中的辅助信息可以为推荐提供有用的帮助, 而传统的协同过滤算法在计算用户相似度时对辅助信息的利用率低, 数据稀疏性大, 导致推荐的精度偏低. 针对这一问题, 本文提出了一种融合用户偏好和多交互网络的协同过滤算法 (NIAP-CF). 该算法首先根据评分矩阵和项目属性特征矩阵挖掘出用户的项目属性偏好信息, 然后使用 *SBM* 方法计算用户间的项目属性偏好相似度, 并用其改进用户相似度计算公式. 在进行评分预测时, 构建融合用户-项目属性偏好信息的多交互神经网络预测模型, 使用动态权衡参数综合由用户相似度计算出的预测评分和模型的预测评分来进行项目推荐. 本文使用 MovieLens 数据集进行实验验证, 实验结果表明改进算法能够提高推荐的精度, 降低评分预测的 *MAE* 和 *RMSE* 值.

关键词: 协同过滤; 多交互神经网络; 项目属性偏好; 相似度; *SBM*

引用格式: 刘锦涛, 谢颖华. 融合用户偏好和多交互网络的协同过滤算法. 计算机系统应用, 2021, 30(9): 179-185. <http://www.c-s-a.org.cn/1003-3254/8134.html>

Collaborative Filtering Algorithm Combining User Preferences and Multi-Interaction Networks

LIU Jin-Tao, XIE Ying-Hua

(School of Information Science and Technology, Donghua University, Shanghai 201620, China)

Abstract: The auxiliary information in recommendation systems can provide real help for recommendation, while the traditional collaborative filtering algorithm has a low utilization rate of the auxiliary information and high data sparsity in the calculation of user similarity, which leads to low recommendation accuracy. In response to this problem, this study proposes an improved collaborative filtering algorithm that integrates user preferences and multi-interactive neural networks (NIAP-CF). Firstly, the information about the item attribute preferences of users is collected according to the rating matrix and the item attribute feature matrix. Then, the *SBM* method is used to calculate the similarity of item attribute preferences between users to improve the calculation formula for user similarity. In the process of score prediction, we build a multi-interactive neural network prediction model integrating user and item attribute preference. Dynamic trade-off parameters are introduced to integrate the predicted scores calculated by user similarity and by the model. Experimental verification based on the MovieLens data set shows that the improved algorithm can improve the recommendation accuracy and reduce the *MAE* and *RMSE* of score prediction.

Key words: collaborative filtering; multi-interactive neural network; item attribute preferences; similarity; *SBM*

随着信息和互联网技术的迅速发展, 海量的网络信息资源使得用户难以在短时间内找到自己感兴趣的信息资源, 这造成了信息过载问题^[1], 这种现象的出现严重影响了互联网的发展. 推荐系统通过分析用户的

历史行为来获得用户的兴趣偏好, 从而进行定向的信息资源推荐^[2,3], 减少用户因筛选信息而消耗的时间, 使得用户能够快速获取到于自己有用的信息资源的推荐. 但随着用户数和项目数的激增, 传统推荐算法面临着

① 收稿时间: 2020-12-15; 修改时间: 2021-01-11; 采用时间: 2021-02-23; csa 在线出版时间: 2021-09-02

严重的数据稀疏问题. 这一问题使得传统推荐算法计算出的用户相似度不够准确, 推荐精度不高. 因此, 改进相似度计算方法、缓解数据稀疏性一直以来是推荐算法研究的重点^[4-6].

王岩等将用户评分分值差异度和用户评分倾向相似性融入到传统的协同过滤算法中, 并用项目评分的标准差改进皮尔逊相似性, 综合用户兴趣和改进后的皮尔逊相似性得到更准确的用户相似度^[7]. Chen 等在余弦相似度和皮尔逊相似度的计算方法中引入平衡参数, 利用平衡参数解决不同用户间评级尺度不同的问题, 从而对这两种方法进行优化^[8]. 随着深度学习的发展, 使用深度学习缓解推荐系统的数据稀疏性问题得到了广泛地研究^[9]. 目前基于深度学习的推荐模型主要是结合神经网络和矩阵分解思想, 将用户和项目的独热编码 (one-hot) 作为模型的输入, 通过嵌入层将稀疏的输入向量映射为稠密向量, 然后利用深度神经网络结构去非线性地学习用户和项目潜向量之间的交互结果, 最后根据学习到的交互结果来生成推荐列表^[10-12].

1 相关知识

1.1 基于用户的协同过滤推荐算法

基于用户的协同过滤推荐算法是通过用户-项目评分矩阵, 计算出用户之间的评分相似度, 然后对评分相似度进行降序排序, 选出与目标用户相似度高的用户构成邻居用户集. 推荐时, 根据邻居用户计算预测评分, 并将预测评分高的项目推荐给目标用户^[13]. 这种推荐方法源于生活中的经验假设: 若两个用户之间的兴趣爱好相似, 则他们很可能对同一项目有着相似的偏好程度^[14].

(1) 用户相似度计算

相似度的计算方法主要有余弦相似性、改进的余弦相似性以及皮尔逊相关系数等^[15,16]. 余弦相似性是两个用户之间的评分向量的夹角余弦值, 余弦值越大, 相似度越高. 改进的余弦相似度考虑到不同用户的评分习惯对相似度计算准确性的影响, 在计算相似度时减去用户的平均评分. 皮尔逊相关系数在计算时要首先找出用户间的共同评分项目集, 然后基于该集合使用改进的余弦相似性进行用户相似度的计算, 计算公式如下:

$$Sim(u, v) = \frac{\sum_{i \in S_{uv}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in S_{uv}} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in S_{uv}} (r_{vi} - \bar{r}_v)^2}} \quad (1)$$

其中, S_{uv} 表示用户 u 和用户 v 的共同评分项目集合,

r_{ui} 和 r_{vi} 分别表示用户 u 和用户 v 对项目 i 的评分, \bar{r}_u 和 \bar{r}_v 分别表示用户 u 和 v 的平均评分.

(2) 评分预测

用户相似度计算完成后, 筛选出邻居用户集. 根据邻居用户对项目的评分和与目标用户的相似度来预测目标用户对项目的评分, 计算公式为:

$$P_{ui} = \bar{r}_u + \frac{\sum_{v \in NU(u)} Sim(u, v)(r_{vi} - \bar{r}_v)}{\sum_{v \in NU(u)} Sim(u, v)} \quad (2)$$

其中, P_{ui} 为目标用户 u 对项目 i 的预测评分值, $NU(u)$ 为目标用户 u 的邻居用户集, $Sim(u, v)$ 表示目标用户 u 与邻居用户 v 之间的相似度.

1.2 多交互神经网络

多层感知机 (Multi-Layer Perceptron, MLP) 指的是有多层隐藏层的神经网络. 相邻的神经网络层之间是全连接的. 单个神经元的计算是一个前向传播过程, 即通过一个线性函数关系和激活函数向前传播.

多交互神经网络通过构建并行的多层感知机来分别学习不同类型实体的隐特征表示. 考虑到隐藏层中每层网络的神经元数目和学习能力不同, 学习到的隐特征表示的维度和意义也不相同, 多交互神经网络充分利用每一层的隐表示结果, 对每层网络所学习到的不同类型实体的隐表示进行一次交互操作, 最后综合所有层的交互结果来获取模型最终的预测结果^[17], 网络模型如图 1 所示.

2 融合用户偏好和多交互网络的协同过滤算法

2.1 用户-项目属性偏好相似度

2.1.1 挖掘用户-项目属性偏好信息

表 1 显示出了项目的属性特征信息, 为便于后续计算, 做出如下定义: S 表示用户 u 的评分项目集, N 为用户 u 的评分项目数, q_s 表示项目 s 的流行度 (评分用户数), A_s 表示项目 s 包含的项目属性.

对于流行度高的项目, 用户选择它更可能是出于从众心理而不是因为项目本身的属性. 因此我们引入惩罚因子 γ 对流行度超过阈值 φ 的项目进行惩罚. 定义用户 u 对项目属性 a 的偏好度 I_{ua} 如下:

$$I_{ua} = \frac{1}{N} \sum_{s \in S} s_a q_s \quad (3)$$

$$q_s = \begin{cases} 1, p_s < \varphi \\ \gamma, p_s \geq \varphi \end{cases}, s_a = \begin{cases} 1, a \in A_s \\ 0, a \notin A_s \end{cases} \quad (4)$$

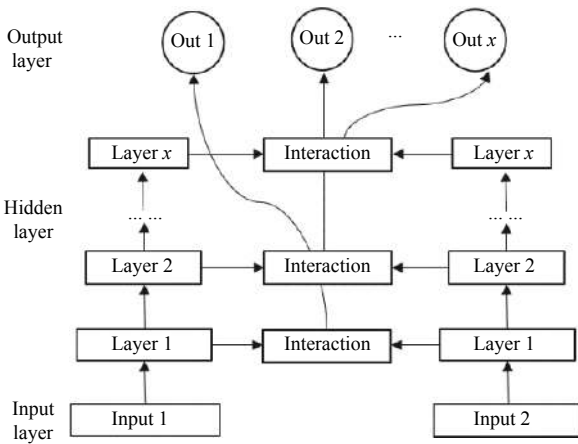


图1 多交互神经网络模型

表1 项目属性信息

项目	Attr ₁	...	Attr _i	...	Attr _k
Item ₁	0	...	0	...	1
⋮	⋮	⋮	⋮	⋮	⋮
Item _i	1	...	0	...	1
⋮	⋮	⋮	⋮	⋮	⋮
Item _m	1	...	1	...	0

2.1.2 计算用户-项目属性偏好相似度

对挖掘出的用户-项目属性偏好信息进行降序排序,得到用户的项目属性偏好排名,然后使用基于集合(Set Based Measure, SBM)的方法计算用户间的项目属性偏好相似度。

SBM是通过计算在不同深度时的两个排序列表的交集大小来衡量排序列表之间的相似度。假设用户 u 和用户 v 对项目属性 $\{a, b, c, d, e\}$ 的偏好排序如下:

$$u: [a, b, c, d, e], v: [b, a, c, d, e]$$

依次计算它们在深度 d 时,两个列表前 d 个元素构成的两个集合的交集,以及交集大小相对当前深度的比例,如表2所示。计算出不同深度的交集比例后,我们使用交集比例的平均值来度量用户之间的项目属性偏好相似度。计算公式如下:

$$SBM(u, v) = \frac{1}{k} \sum_{d=1}^k \frac{|u_{1:d} \cap v_{1:d}|}{d} \quad (5)$$

在推荐系统中,我们一般只关注用户最喜爱的那部分项目属性,并由此进行项目推荐。因此,我们希望在计算两个排序列表的相似度时,能够关注 top 元素的相

对位置权重。在前面例子中,项目属性数 k 为5, $SBM(u, v)$ 为0.8。假设另有用户 w 的项目属性偏好排序为: $[a, b, c, e, d]$,与列表 u 相比,列表 w 是通过交换列表 u 的最后两个元素的位置得到的,而列表 v 是通过交换列表 u 的前2个元素的位置得到的,计算出 $SBM(u, w)$ 为0.95。由此可以看出,SBM自身具有 top-weightiness属性。

表2 SBM方法

深度	列表 $u@d$	列表 $v@d$	交集	交集比例
1	a	b	$\{\}$	0/1=0
2	a, b	b, a	$\{a, b\}$	2/2=1
3	a, b, c	b, a, c	$\{a, b, c\}$	3/3=1
4	a, b, c, d	b, a, c, d	$\{a, b, c, d\}$	4/4=1
5	a, b, c, d, e	b, a, c, d, e	$\{a, b, c, d, e\}$	5/5=1

2.2 改进用户相似度计算公式

协同过滤算法通过计算用户偏好进行项目推荐,其中传统的用户协同过滤算法仅使用评分相似性来选择与目标用户具有相似偏好的邻居用户,没有考虑从辅助信息中提取对提升推荐性能有用的信息,数据稀疏性大,推荐精度不高。本文算法根据评分数据挖掘出用户的项目属性偏好信息,然后使用SBM方法计算出用户-项目属性偏好相似度,并用其改进用户相似度计算公式。

Jaccard系数是一种比较常用的用户相似性计算方法,它使用用户的共同评分项目在所有评分项目中所占的比例来衡量用户之间的相似性:

$$Jaccard(u, v) = \frac{|I_u \cap I_v|}{|I_u \cup I_v|} \quad (6)$$

对于共同评分项目,用户对其的评价可能截然相反的,然而Jaccard系数在计算用户相似性时没有考虑用户评分差异对用户偏好的影响,这会导致用户相似性计算不准确。式(1)的皮尔逊相关系数可以衡量用户间的评分相似性,但在数据稀疏,用户之间共同评分项目少的情况下,计算结果会不准确。因此可以使用皮尔逊相关系数来修正Jaccard系数,修正后的Jaccard系数如下:

$$JAC(u, v) = sim(u, v) \times Jaccard(u, v) \quad (7)$$

综合用户-项目属性偏好相似度和修正后的Jaccard系数, α 为调节参数,表示两种相似度之间的比重,最终的用户相似度计算公式如下:

$$Sim(u, v) = \alpha \times SBM(u, v) + (1 - \alpha) \times JAC(u, v) \quad (8)$$

2.3 评分预测

2.3.1 构建多交互神经网络模型

构建融合用户-项目属性偏好信息的多交互神经

网络预测模型, 如图 2 所示. 模型由输入层、嵌入层、隐藏层和输出层组成. 输入层输入用户和项目的 one-hot 编码向量, 嵌入层包含用户-项目属性偏好向量

(user attribute preference vector), 用户嵌入向量 (user latent vector), 项目属性特征向量 (item attribute feature vector), 项目嵌入向量 (item latent vector).

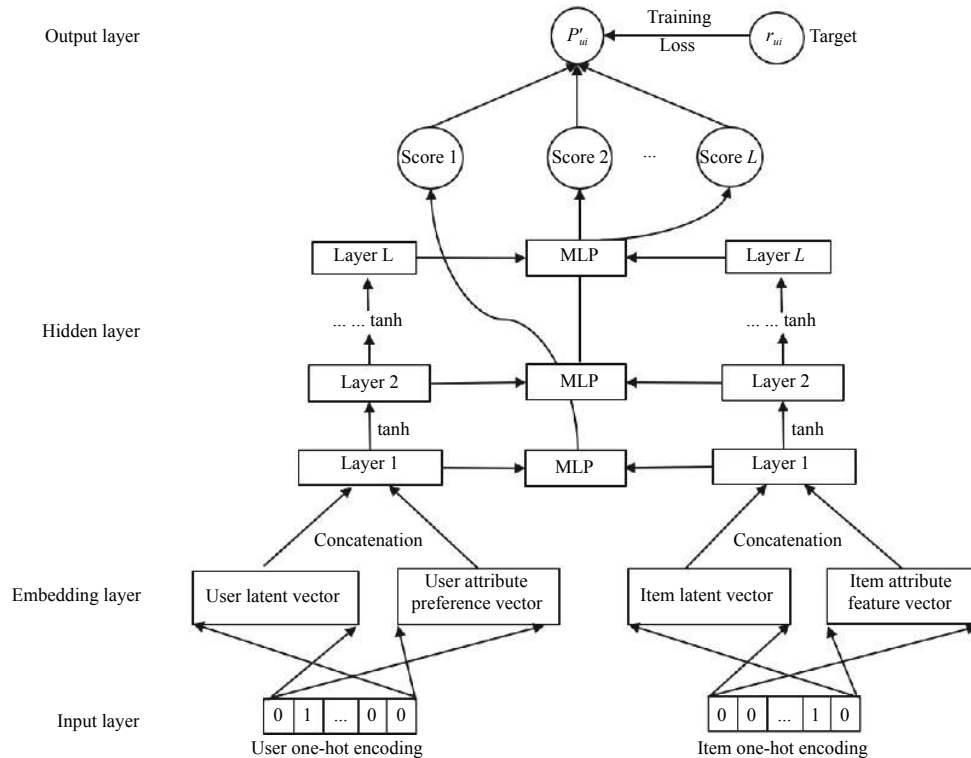


图 2 融合用户-项目属性偏好的多交互神经网络模型

模型采用多交互的网络结构, 通过两个并行的多层隐藏层网络去分别学习用户和项目在不同维度空间中的隐特征表示, 隐藏层的神经元数目表示要学习的隐特征表示的维度大小. 此外, 该模型融合了用户-项目属性偏好信息, 一定程度上缓解了单一评分数据的数据稀疏问题. 将用户-项目属性偏好向量和用户嵌入向量拼接后通过多层隐藏层进行非线性交互提取用户在不同维度空间中的隐特征表示. 同样将项目属性特征向量和项目嵌入向量拼接后通过多层隐藏层来提取项目在不同维度空间中的隐特征表示. 隐藏层采用金字塔型的网络结构, 后一层网络的神经元数目为前一层网络神经元数目的一半, 选择 tanh 函数作为模型的激活函数.

用 U_i 表示用户-项目属性偏好向量, U_h 表示用户嵌入向量, σ 表示激活函数, W_l 和 b_l 分别表示第 l 层隐藏层的权重矩阵和偏差向量, 则用户的各层隐特征表示如下:

$$z = \begin{bmatrix} U_i \\ U_h \end{bmatrix} \quad (9)$$

$$z_1 = \phi_1(z) = \sigma(W_1 z + b_1) \quad (10)$$

$$\dots$$

$$z_L = \phi_L(z_{L-1}) = \sigma(W_L z_{L-1} + b_L) \quad (11)$$

同理, 对于项目属性特征向量和项目嵌入向量, 通过多层隐藏层网络得到项目的各层隐特征表示.

在基于深度学习的推荐模型中, MLP 模型将用户和项目的 one-hot 编码向量通过嵌入层映射到新的维度空间中, 获得用户和项目的嵌入向量 (隐特征表示), 然后将用户和项目的嵌入向量拼接后通过多层感知机得到模型的预测评分. 然而特征维度大小选取对于预测的准确性有着重要影响, 一般需要进行多次实验来确定合适的特征维度, 并且在不同的维度空间中用户和项目的隐特征表示的意义不同, 只使用单一维度空间中的隐特征表示进行预测限制了模型的推荐性能. 在本文所使用的多交互神经网络模型中, 模型通过两个并行的多层隐藏层网络来分别学习用户和项目在不同维度空间中的隐特征表示, 后一层网络在前一层网络提取的隐特征表示的基础上进一步提取更深层次的隐特征表示. 隐藏层中每层网络的神经元数目和学习

能力不同,学习到的用户和项目的隐特征表示的维度和意义也不同.多交互神经网络综合每一层网络学习到的隐特征表示来提高模型的推荐性能,对每层用户和项目的隐特征表示进行一次交互操作,最后综合所有层的交互结果作为模型最终的预测结果.

我们使用3层感知机模型来获取每一层用户和项目的隐特征表示的交互结果,计算公式如下:

$$S_l = \phi_3(\phi_2(\phi_1(\begin{bmatrix} z_l \\ x_l \end{bmatrix}))) \quad (12)$$

其中, z_l 和 x_l 分别表示第 l 层的用户和项目的隐特征表示, S_l 表示 z_l 和 x_l 拼接后通过三层感知机模型的交互结果.

在输出层,将隐藏层每层用户和项目的隐特征表示的交互结果拼接起来进行一次全连接操作得到模型最终的预测评分:

$$P'_{ui} = h^T \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_L \end{bmatrix} \quad (13)$$

本文模型是用来进行评分预测任务的,因此选择基于点的方式来构建损失函数,使用均方差作为模型要优化的损失函数:

$$Loss = \sum_{u,i} (r_{ui} - P'_{ui})^2 \quad (14)$$

2.3.2 综合预测评分

对于评分预测系统,我们首先使用改进后的用户相似度筛选出邻居用户集,并根据评分预测公式计算出预测评分 P_{ui} .然后通过融合用户-项目属性偏好信息的多交互神经网络模型获得模型的预测评分 P'_{ui} .最终预测结果是这两种预测评分的动态加权:

$$\hat{r}_{ui} = \frac{C_{Nu}^i}{C_{Nu}^i + \lambda C_{Nu}} P_{ui} + \left(1 - \frac{C_{Nu}^i}{C_{Nu}^i + \lambda C_{Nu}}\right) P'_{ui} \quad (15)$$

其中, C_{Nu} 表示邻居集的大小, C_{Nu}^i 表示邻居集中对项目 i 评分的用户数, λ 为放缩因子. C_{Nu}^i 值越大,则由评分预测公式计算出的预测值在最终预测评分中的比重就越大.

在对用户进行推荐时,使用改进算法预测目标用户对项目的评分,选择预测评分高的项目推荐给目标用户.改进算法的推荐流程如图3所示.

3 实验评估

3.1 实验数据集及度量标准

MovieLens 数据集是推荐系统中常用的验证数据集,该数据集有多种规格,如:100k、1M和10M等.本文实验使用100k数据集,它包含943位用户对1682部电影的10万条评分信息,评分范围为1到5分.

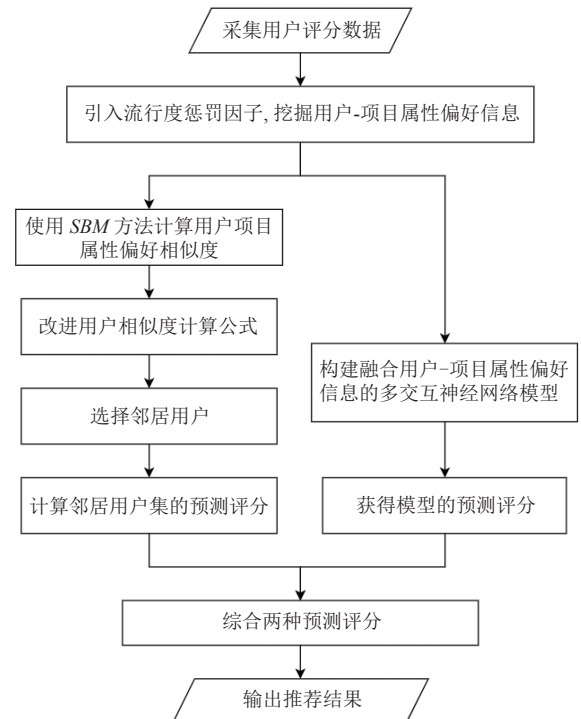


图3 改进算法推荐流程图

平均绝对误差(MAE)和均方根误差(RMSE)是推荐算法中使用最为广泛的两种性能评价指标,它们反映出预测评分的准确性,计算公式如下:

$$MAE = \frac{1}{N} \sum_{i=1}^N |\hat{r}_i - r_i| \quad (16)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{r}_i - r_i)^2} \quad (17)$$

F1-score是精确率(precision)和召回率(recall)的加权平均值,计算公式为:

$$F1-score = \frac{2 \times precision \times recall}{precision + recall} \quad (18)$$

Precision表示模型预测为正的样本中实际也为正的样本占被预测为正的样本的比例. Recall表示实际为正的样本中预测也为正的样本占实际为正的样本的比例. F1-score是两者的综合, F1-score越高,说明模型越稳健.

3.2 实验结果与分析

3.2.1 推荐性能分析

本文的相关参数设置:流行度阈值 $\varphi = 200$,惩罚因子 $\gamma = 0.5$,调节参数 $\alpha = 0.3$,放缩因子 $\lambda = 0.2$,batch-size为256,隐藏层的网络层数为3,优化器为Adam,学习率为0.001,正则化率为0.0001.

本文提出的综合预测模型记为NIAP-CF.实验将测试集中实际评分高于4分的样本标记为正样本,其

余标记成负样本. 在推荐时, 由预测评分高于 4 分的样本产生推荐列表.

对于基于用户相似度的推荐模型, 将本文算法与皮尔逊相似性计算方法、Jaccard 相似性计算方法以及文献 [6] 中相似度计算方法 (UII-CF) 进行 MAE 值、RMSE 值以及 F1-score 的比较来验证本文推荐算法的推荐效果.

图 4 和图 5 分别显示了在 MovieLens 数据集中本文提出的算法 NIAP-CF 和其它 3 种算法在不同邻居用户数下的 MAE 和 RMSE 值对比情况. 从图中可以看出, 4 种算法的 MAE 和 RMSE 值都随着邻居用户数的增加而逐渐减小, 并且在选取邻居用户数相同时, 本文算法的 MAE 和 RMSE 值小于其它 3 种算法, 预测精度最高. 此外, 本文算法综合了邻居集的预测评分和多交互神经网络模型的预测评分, 预测的精度较为稳定, 可以在邻居用户较少时获得较高的推荐精度.

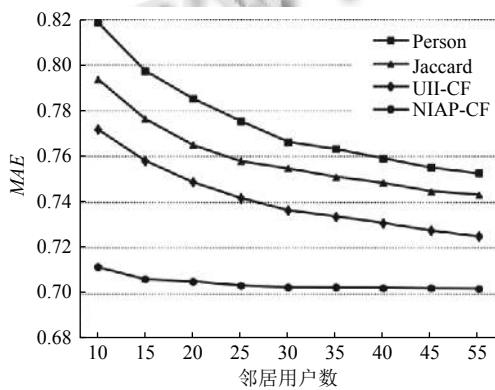


图 4 MAE 值对比结果

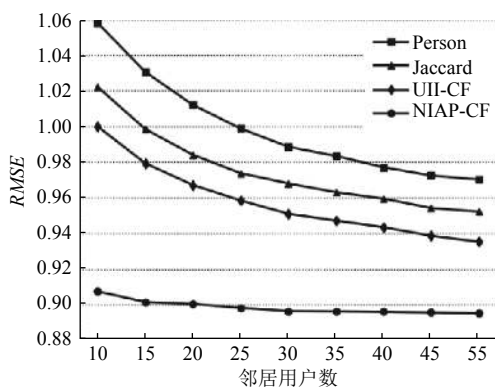


图 5 RMSE 值对比结果

图 6 显示了在 MovieLens 数据集中 4 种算法的 F1-score 值的对比结果. 由图可以看出, 在邻居用户数相同时, 本文算法的 F1-score 值最大, 模型的稳健度优于其它 3 种算法.

对于基于深度学习的推荐模型, 将本文提出的 NIAP-CF 模型与文献 [10] 中 He 等提出的 GMF 模型、MLP 模型以及 NeuCF 模型进行对比实验来验证本文推荐算法的有效性.

表 3 显示了在 MovieLens 数据集中本文模型与 GMF 模型、MLP 模型以及 NeuCF 模型的 MAE 值、RMSE 值以及 F1-score 的对比情况. 从表中可以看出, 本文模型的 MAE 和 RMSE 值低于其它 3 种模型, 预测的精度最高, 并且本文模型的 F1-score 最高, 模型的稳健度优于其它 3 种模型.

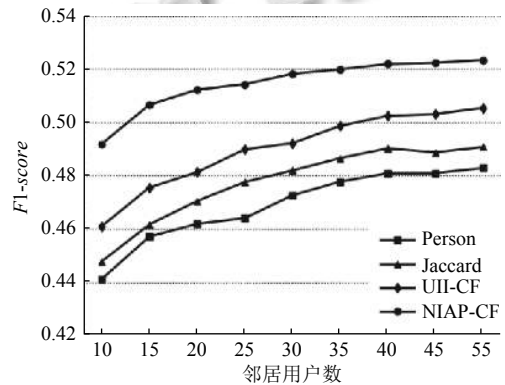


图 6 F1-score 对比

表 3 MovieLens 数据集对比实验结果

模型	MAE	RMSE	F1-score
MLP	0.721	0.917	0.505
GMF	0.717	0.913	0.510
NI-CF	0.715	0.910	0.513
NIAP-CF	0.701	0.894	0.522

3.2.2 参数分析

在用户相似度计算部分, 调节参数 α 表示用户-项目属性偏好相似度和修正后的 Jaccard 系数之间的比重, 对于计算用户相似度的准确性有着非常重要的影响. 因此, 需要确定 α 的值来消除其对推荐结果的影响. 选取邻居用户数为 30, 观察 MAE 值在不同 α 取值下的变化情况, 结果如图 7 所示. 从图中可以看出, 当 α 值为 0.3 时, 本文相似性计算方法达到最优.

在融合用户-项目属性偏好信息的多交互神经网络模型中, 用于学习用户和项目隐特征表示的隐藏层层数是神经网络中的一个非常重要的参数. 我们通过实验研究隐藏层的网络层数对模型预测精度的影响. 实验结果如表 4 所示. 由实验结果可以看出: 一定数目的隐藏层层数能够改善模型的推荐性能. 但更深层次的网络结构会耗费大量的计算资源, 在实际中, 隐藏层层数的选择可以根据具体需求来确定.

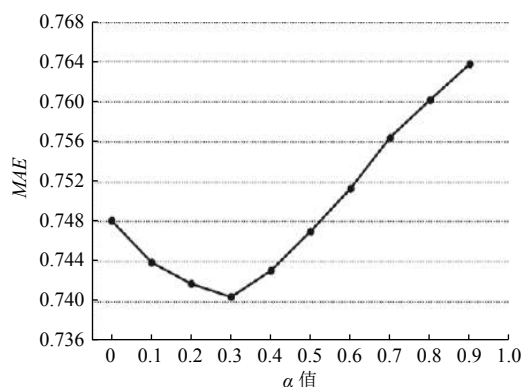
图7 MAE随 α 值的变化情况

表4 隐藏层网络层数对比结果

层	MAE	RMSE
Layer2	0.712	0.907
Layer3	0.709	0.903
Layer4	0.707	0.900
Layer5	0.706	0.899

4 总结与展望

传统的用户协同过滤算法仅通过用户间的评分相似性来预测评分,没有考虑从外部辅助信息中提取出对推荐有用的信息,数据稀疏性大,推荐精度不高。本文提出的NIAP-CF算法首先根据评分矩阵和项目属性特征矩阵挖掘出用户的项目属性偏好信息,然后使用SBM方法计算用户间的项目属性偏好相似度,并用其改进用户相似度计算公式。在进行评分预测时,建立融合用户-项目属性偏好信息的多交互神经网络预测模型,引入动态权衡参数,对由相似度计算出的预测评分和模型的预测评分进行动态加权,从而获得最终的预测评分。实验结果表明,与传统的协同过滤算法相比,本文提出的改进算法降低了评分预测的MAE和RMSE值,提高了推荐精度。

参考文献

- 代宝续, 杨晓雪, 罗蕊. 社交媒体用户信息过载的影响因素及其后果. 现代情报, 2020, 40(1): 152-158.
- 周万珍, 曹迪, 许云峰, 等. 推荐系统研究综述. 河北科技大学学报, 2020, 41(1): 76-87. [doi: 10.7535/hbkd.2020yx01009]
- 黄贤英, 龙姝言, 谢晋. 结合用户兴趣度聚类的协同过滤推荐算法. 计算机应用研究, 2019, 36(9): 2609-2612, 2617.
- 吕成成. 基于用户项目属性偏好的协同过滤推荐算法. 计

算机技术与发展, 2018, 28(4): 152-156, 160. [doi: 10.3969/j.issn.1673-629X.2018.04.032]

- Li R, Li MQ, Guo WQ. Research on collaborative filtering algorithm with improved similarity. Computer Science, 2016, 43(12): 206-208, 240.
- Zheng J, Qian YR, Yang XY, et al. Collaborative filtering algorithm based on trust and item preference. Journal of Computer Applications, 2016, 36(10): 2784-2788, 2798.
- 王岩, 张杰, 许合利. 结合用户兴趣和改进的协同过滤推荐算法. 小型微型计算机系统, 2020, 41(8): 1665-1669. [doi: 10.3969/j.issn.1000-1220.2020.08.016]
- Chen H, Li ZK, Hu W. An improved collaborative recommendation algorithm based on optimized user similarity. The Journal of Supercomputing, 2016, 72(7): 2565-2578. [doi: 10.1007/s11227-015-1518-5]
- Xue HJ, Dai XY, Zhang JB, et al. Deep matrix factorization models for recommender systems. Proceedings of the 26th International Joint Conference on Artificial Intelligence. Melbourne, VIC, Australia. 2017. 3203-3209.
- He XN, Liao LZ, Zhang HW, et al. Neural collaborative filtering. Proceedings of the 26th International Conference on World Wide Web. Perth, WA, Australia. 2017. 173-182.
- He XN, Du XY, Wang X, et al. Outer product-based neural collaborative filtering. Proceedings of the 27th International Joint Conference on Artificial Intelligence. Stockholm, Sweden. 2018. 2227-2233.
- 李博. 数据稀疏情况下的协同过滤推荐算法研究 [硕士学位论文]. 北京: 北京邮电大学, 2019.
- 王琰. 基于用户特征与时间权重的电影推荐算法研究 [硕士学位论文]. 淮南: 安徽理工大学, 2018.
- Wang HX. An improved collaborative filtering recommendation algorithm. Proceedings of the IEEE 4th International Conference on Big Data Analytics. Suzhou, China. 2019. 431-435.
- Zhang J, Peng QK, Sun SQ, et al. Collaborative filtering recommendation algorithm based on user preference derived from item domain features. Physica A: Statistical Mechanics and its Applications, 2014, 396: 66-76. [doi: 10.1016/j.physa.2013.11.013]
- Zhang J, Lin YJ, Lin ML, et al. An effective collaborative filtering algorithm based on user preference clustering. Applied Intelligence, 2016, 45(2): 230-240. [doi: 10.1007/s10489-015-0756-9]
- 李同欢. 基于深度学习的推荐模型研究 [硕士学位论文]. 重庆: 西南大学, 2019.