

基于用户坐标距离与误差修正的推荐算法

艾均, 孙逸帆, 苏湛, 罗黔琴

上海理工大学光电信息与计算机工程学院, 上海

收稿日期: 2024年3月29日; 录用日期: 2024年5月2日; 发布日期: 2024年5月8日

摘要

推荐系统是一种根据用户历史行为、社交关系等信息去判断用户感兴趣的物品的智能系统, 它可以有效的解决互联网信息过载的问题, 因此具有重要的研究意义和广泛应用价值。为了解决推荐算法存在准确性有待提升和可扩展性不足的问题, 本文提出了一种带有相似性选择和误差修正的用户坐标与评分距离推荐算法。该算法首先基于用户间相似性筛选出正相关的邻居, 针对筛选后的邻居集计算评分距离。然后建立用户间评分距离的合成坐标模型, 与得到的用户坐标间距离进行评分预测。最后基于训练题设计误差修正算法, 进一步提升预测准确度。通过本文算法与其他推荐算法在MovieLens数据集上进行对比, 试验结果表明, 该算法可以有效提高推荐算法的预测准确性和可扩展性。研究揭示了利用用户坐标距离进行预测的可行性, 为进一步研究推荐系统的性能提升和运行机理提供了有效的依据。

关键词

推荐系统, 相似性网络, 评分距离, 合成坐标, 误差修正

Recommendation Algorithm Based on User Coordinate Distance and Error Correction

Jun Ai, Yifan Sun, Zhan Su, Qianqin Luo

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Mar. 29th, 2024; accepted: May. 2nd, 2024; published: May. 8th, 2024

Abstract

The recommendation system is an intelligent system that determines the items that the user is in-

interested in based on the user's historical behavior, social relationships and other information. It can effectively solve the problem of Internet information overload, so it has important research significance and wide application value. In order to solve the problems of recommendation algorithm that needs to be improved in accuracy and insufficient scalability, this paper proposes a user coordinate and rating distance recommendation algorithm with similarity selection and error correction. The algorithm first selects positively correlated neighbors based on the similarity between users, and calculates the scoring distance for the filtered neighbor set. Then a synthetic coordinate model of the rating distance between users is established, and the rating prediction is performed with the obtained distance between user coordinates. Finally, an error correction algorithm is designed based on the training questions to further improve the prediction accuracy. By comparing the algorithm in this article with other recommendation algorithms on the MovieLens data set, the experimental results show that the algorithm can effectively improve the prediction accuracy and scalability of the recommendation algorithm. The research reveals the feasibility of using user coordinate distance for prediction, and provides an effective basis for further research on the performance improvement and operating mechanism of the recommendation system.

Keywords

Recommender System, Similarity Network, Rating Distance, Synthetic Coordinates, Error Correction

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

推荐系统目前已经成为各类在线平台或在线应用的不可或缺的一部分。随着互联网接入在全球范围内的普及,消费者的选择呈指数级增长。尽管这允许多种选择和更广泛的选择,但将消费者的偏好与最合适的产品相匹配变得越来越困难[1]。系统试图通过分析消费者偏好并尝试预测用户对新商品的偏好来改变这种情况。推荐系统有着广泛的应用,如:在拥有大量产品的消费者网站中,为消费者提供可能感兴趣的产品的有针对性的信息;或者设计营销策略,用于预测产品的受欢迎程度[2]。

目前推荐系统包括基于内容的推荐、基于协同过滤的推荐、基于知识的推荐和混合推荐[3]。

基于内容的推荐算法[4]主要根据物品的相关信息、用户的相关信息和用户对物品的行为来构建推荐算法。基于内容的推荐算法一般只依赖于用户自身的行为为用户提供推荐,不涉及到其他用户的行为。

基于协同过滤的推荐算法是一种广泛应用的推荐算法,它主要依赖用户的历史行为数据来进行推荐。这种算法可以进一步分为基于记忆的协同过滤算法(Memory-based Collaborative Filtering)和基于模型的协同过滤算法(Model-based Collaborative Filtering) [5]。基于记忆的协同过滤算法无论是基于物品[6]还是基于用户[7],核心都是计算相似性,基于相似用户更容易做出相似的评分行为,相似的物品更容易获得相似的评分[8]。基于模型的协同过滤算法包括聚类模型[9]、矩阵分解模型[10]、神经网络[11]等,核心是建立物品和用户的模型表示,并利用训练好的模型来进行预测和推荐。

基于知识的推荐算法是将已知的数据进行专家知识进行建模和抽象,结果作为知识系统存储,并在用户提出需求时通过约束条件加以筛选,进行推荐[12]。

混合推荐算法是综合上述各种类型的推荐算法,用以提高推荐性能,或是满足不同推荐需求,其混合方案包括:加权型、交叉型、切换型、瀑布型、特征组合型、特征增强型等[13]。

推荐系统在面对多任务时，仍出现一些挑战。比如：预测准确性、算法可扩展性、冷启动等[14]。预测准确性用于评估推荐系统预测用户对物品的评分能力，准确性越高，说明推荐系统对用户的喜好理解得越准确。可扩展性是指推荐系统能够有效地处理大规模的用户和物品数据。随着用户数量和物品数量的增加，推荐系统需要能够在有限的计算资源和时间内，依然能够提供高质量的推荐结果。冷启动指的是加入系统的新用户和新物品在没有评价数据等关键信息条件下的推荐问题。

本文针对推荐系统准确性有待提升、可扩展性不足的问题，研究了推荐系统中用户相似性、合成坐标和误差修正的相关技术，提出了基于误差修正与用户坐标距离的推荐算法(Recommendation Algorithm Based on User Coordinate Distance and Error Correction)。

2. 相关工作

基于用户共同评分物品集合计算的用户相似性，在推荐系统研究中一般被称为用户间的相似性(User Similarity)。以基于记忆的协同过滤推荐算法为例，其核心思想就是通过利用与目标用户兴趣偏好相似的用户群体的喜好程度来预测和推荐。具体来说就是计算目标用户与其他所有用户两两之间的相似性，再通过相似性从高到低来选择必要的 k 个邻居，最后根据邻居的评分来对目标用户进行预测[15]。推荐系统领域提出了许多方法来计算用户间相似性，例如皮尔逊相似系数[16]、余弦相似系数[17]、Jaccard 相似系数[18]等。这些相似性算法考虑不同的因素和计算方式来度量用户间的相似性。

合成坐标算法(Vivaldi) [19]是一种轻量级、自适应地合成网络坐标算法，它使用欧几里得坐标系及其关联的距离系数，每个节点与其他节点的一部分进行交互，将自身定位在空间中，以便节点距离和测量的延迟相匹配。当系统收敛时，意味着每个节点都已获得所需的位置，任何一对节点之间的距离都可以准确预测其延迟。

Harris 等提出了 SCoR [20]算法，将坐标分配给物品和用户，以便用户和物品之间的距离可以准确预测用户对物品的偏好。由于 SCoR 是无参数的，所以不需要微调就可以获得高性能。同时，与其他算法相比更能抵抗冷启动问题。但是，由于需要对所有的用户和物品数据进行建模，SCoR 的执行时间非常缓慢，且算法的可扩展性较低。

误差修正是推荐系统中的一种常见方法，主要目标是 minimized 预测误差，通过调整模型的参数，使得预测评分和实际评分的误差尽可能小。

梯度下降[21]是一种迭代的误差修正算法，它的工作原理是在每一步中，根据当前的误差，对模型的参数进行微调。如果预测评分高于实际评分，那么降低预测评分；反之，则提高预测评分。从一个随机的初始点开始，沿着函数的梯度的负方向，逐步更新参数，直到找到一个局部最小值。

Harris 等提出了 DTEC [22]算法，使用误差修正用于提高推荐系统的预测性能。DTEC 在推荐系统初始执行之后引入第二阶段，该阶段考虑用户和物品之间训练集中的误差及相似性来改进其预测。这种方法适用于任何具有正训练误差的基于模型的推荐算法，实验结果有效提高了推荐的准确性。

结合用户相似性、合成坐标和误差修正技术，本文将针对推荐系统准确性与可扩展性问题展开研究。

3. 方法

为了解决推荐系统准确性有待提升、可扩展性不足的问题，本文提出一种基于用户坐标距离与误差修正的推荐算法。如图 1 所示，算法包含了四个关键的步骤：第一、采用皮尔逊相似系数筛选用户的相似性，将相似性大于阈值的用户作为邻居，再计算用户之间的评分距离；第二、将用户和评分距离带入合成坐标算法建立模型，计算出用户间的坐标距离；第三、选择待测用户在合成坐标中所对应的邻居，预测用户对物品的评分；第四、使用误差修正算法对预测准确性进行更好地提升。

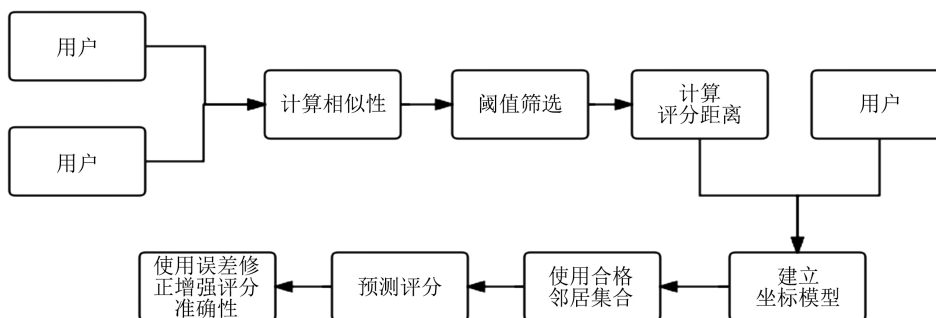


Figure 1. Algorithm structure diagram
图 1. 算法结构图

3.1. 相似性，评分距离计算及选择

基于距离模型的推荐算法在计算上相对简单且运行时间较短，但是因为没有考虑用户间的相似性，所以精度不高。本文使用皮尔逊相似系数计算用户之间的相似性。一般来说，两个用户相似性越大，评分相似性就越强，该邻居在预测的重要性就越高。

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

其中，集合 I_{uv} 代表用户 u, v 共同评分的物品集合， r_{ui} 和 r_{vi} 代表用户 u, v 对物品 i 的评分， \bar{r}_u 和 \bar{r}_v 代表用户 u, v 对所有物品的平均评分。

用户之间的评分距离计算公式定义如式(2)所示：

$$d_{uv} = \sum_{i \in I_{uv}} \frac{r_{vi} - r_{ui}}{|I_{uv}|} \quad (2)$$

其中， d_{uv} 是用户 u 和用户 v 的评分之间的评分距离， r_{ui} 和 r_{vi} 代表用户 u, v 对物品 i 的评分，集合 I_{uv} 代表用户 u, v 共同评分的物品集合， $|I_{uv}|$ 代表集合的大小。

UCD-EC 根据公式(1)计算用户之间的相似性，通过公式(3)对用户进行过滤，选择符合条件的用户作为预测目标的邻居用户。

$$\begin{cases} v \in N_u & \text{if } s_{uv} \geq T_h \\ v \notin N_u & \text{if } s_{uv} < T_h \end{cases} \quad (3)$$

其中， N_u 代表用户 u 选择的邻居集， s_{uv} 代表用户 u, v 的相似性， T_h 是 UCD-EC 算法设定的阈值。

3.2. 合成坐标计算

为了能更好地提高精度，将每个用户以及 3.1 中求出的评分距离进行合成坐标布局。本文使用修改后的合成坐标算法，将用户节点及 3.2 中计算、筛选后的评分距离作为网络节点，而所有用户的评分距离作为延迟，形成一个二分图，由用户节点和评分距离作为节点组成。该算法的输入是 (u, v, d_{uv}) 的三元组，其中 u, v 代表不同的用户节点， d_{uv} 是用户 u 和用户 v 的评分之间的评分距离。

对于训练集中的每个 (u, v, d_{uv}) 三元组，在节点 u 和 v 之间添加一条边。基于评分距离 d_{uv} ，为每条边分配一个权重 d'_{uv} ，该权重反映了边 (u, v) 的距离。如果用户 u 和用户 v 的评分距离更相近，那么它们 d'_{uv} 值会更高，反映出它们的 (u, v) 对对应于距离更小的边，反之亦然。最小评分 $\min R$ 的距离指定为 100，而最高评分的距离指定为 0。给定这些值，距离 d'_{uv} 计算如式(4)所示：

$$d'_{uv} = 100 \cdot \left(\frac{\max R - d_{uv}}{\max R - \min R} \right) \quad (4)$$

其中，节点 u 和节点 v 之间的用户坐标距离即为 d'_{uv} ， $\min R$ ， $\max R$ 表示最小偏好和最大偏好评级。因此，把 Vivaldi 中网络节点之间的延迟替换为上述等式后，可以根据用户和用户之间的评分距离算出用户间坐标距离。

3.3. 评分预测

通过合成坐标计算出所有用户的坐标距离后，针对目标用户，UCD-EC 会对他的邻居进行选择。对于用户 u ，如果用户 v 与他有足够的共同评分且存在空间中的坐标，则会计算 u 和 v 的评分差值，将结果放入用户 u 的邻居集合中。计算公式如式(5)所示：

$$r_{uv} = \max R - \frac{d'_{uv} \cdot (\max R - \min R)}{100} \quad (5)$$

其中， r_{uv} 是用户 u 和用户 v 的评分差值， d'_{uv} 是 u 和用户 v 的坐标距离， $\min R$ ， $\max R$ 表示最小偏好和最大偏好评级。

然后通过邻居集合中的评分差来预测目标用户的评分。如果有邻居，该用户的预测评分即为它对所有物品评分的平均值。基于邻居评分预测用户评分，预测公式如式(6)所示：

$$r_{ui} = \begin{cases} \frac{1}{|N_u|} \sum_{v \in N_u} (r_{uv} + r_{vi}), & |N_u| > 0 \\ \bar{r}_u, & |N_u| \leq 0 \end{cases} \quad (6)$$

其中， r_{ui} 是用户 u 对物品 i 评分的预测值， $|N_u|$ 是用户 u 的邻居数量， r_{uv} 是用户 u 和用户 v 的评分差值， r_{vi} 是用户 v 对物品 i 的评分， \bar{r}_u 是用户 u 对所有其他物品的评分均值。

3.4. 误差修正

为了更好地提高 UCD-EC 的预测准确性，本文使用了一种误差修正(Error Correction, EC)方法。该方法作用于训练集，对于每个用户，计算其预测值和实际值的平均绝对误差(MAE)，找出令 MAE 最小的最佳误差修正值 E 。算法 1 展示了单个用户误差修正的计算方法。

算法 1. 误差修正算法

输入：最大误差 $\max E$ ，误差步长 E_p ，实际值序列 R ，预测值向量 P

输出：使预测值和实际值之间 MAE 最小的误差修正值 E

1: 初始化最佳误差 E

2: 初始化 $\min MAE$ 为最大浮点数

3: 初始化最小误差 $\min E$ 为最大误差 $\max E$ 的相反数

4: FOR $error \in (\min E, \max E)$ 以 E_p 为步长 DO

5: 将 $error$ 加到 P 上，得到修正后的预测值向量 P'

6: 计算序列 P' 与 R 之间的 MAE 值

7: IF 计算出的 MAE 小于 $\min MAE$ THEN

8: $\min MAE = MAE$

9: $E = error$

10: END IF

11: END FOR

12: RETURN 经过计算得出的最佳误差修正值 E ，使得预测值和实际值的 MAE 最小

在此基础上, 对 3.3 中求出的预测评分进行误差修正, 公式如式(7)所示:

$$r'_{ui} = r_{ui} + E_u \quad (7)$$

其中, r'_{ui} 是进行误差修正后用户 u 对物品 i 评分的预测值, r_{ui} 是公式(6)得出的评分预测值, E_u 为该用户通过算法 1 出的误差修正值。

4. 实验

4.1. 实验设计

本实验使用 MovieLens 25M [23]数据集, 此数据集从电影推荐网站中收集的, 包含了由 162,000 名用户对 62,000 部电影的 25,000,000 条评分信息。本实验中采用了 100 KB 的数据集, 包括了 610 个用户对 9742 部物品的 100,836 条评分数据。

实验在处理器为 12th Gen Intel(R) Core(TM) i7-12700H 2.30 GHz, 32 GB 内存, 操作系统为 64 位的 Windows 11 的环境下运行, 采用了相同的编程语言进行算法实现和结果的可视化。

为了更好地评估实验的性能, 本文采用十折交叉验证, 每次使用 1 份作为测试集, 剩余 9 份作为训练集, 每次选取训练集的数据进行训练, 测试集的数据进行验证, 最后的结果为 10 次实验的平均值。

4.2. 基准对比算法

为了验证算法准确性, 本文将 UCD-EC 算法与基于合成坐标算法(Synthetic Coordinate, SCoR, 2017)、双重误差校正算法(Dual Training Error based Correction approach, DTEC, 2021)、用户观点传播算法(User Opinion Spreading, UOS, 2015)、基于信息熵的协同过滤算法(Entropy, 2020)、基于相似性网络资源分配算法(SRA, 2022)、基于向量相似性的算法(VS, 2020)等几种算法进行比较, 所有算法用到的参数都采用其论文中最优的指标。

1) 基于合成坐标的算法(SCoR) [20]把物品和用户当成节点, 将合成坐标分配给节点以使用户和物品间的距离可以准确预测用户对某个物品的偏好;

2) 双重误差校正算法(DTEC) [22]是一种基于双重(物品和用户训练误差的校正方法), 可以对一些基于模型的推荐算法进行误差校正, 提高其准确率;

3) 用户观点传播算法(UOS) [24]核心思想是将用户的观点分为正面和负面, 从而判断用户之间评价的相似程度;

4) 基于信息熵的协同过滤算法(Entropy) [25]将信息熵引入相似性计算, 通过计算用户之间的信息熵, 改进用户的相似性;

5) 基于相似性网络资源分配(SRA) [26]设计了一种基于用户相似性网络内资源分配的用户 - 物品推荐算法, 以提高预测精度, 同时保持推荐算法的多样性;

6) 基于向量相似性的算法(VS) [27]定义了全局相似性、局部相似性和元相似性作为用户间相似性指标, 使得针对不同物品可以用不同方式测量用户相似性。

4.3. 评价指标

本文主要考查了 UCD-EC 算法与基准算法间在评分预测误差、推荐分类质量、算法可扩展性三个方面的性能差异。其中, 使用平均绝对误差(MAE)、来评估算法预测的误差水平, 使用 Accuracy、Precision 来评估算法的推荐分类质量。

4.3.1. 预测准确性指标

平均绝对误差(MAE) [28]是模型评价中使用的标准度量。在推荐系统中, MAE 表示预测值与实际值

的平均误差, MAE 越小, 说明预测越准确。

$$MAE = \frac{1}{n} \sum_{i=1}^n |r'_{u,i} - r_{u,i}| \quad (8)$$

其中, n 表示实验中预测集的预测目标数量, $r'_{u,i}$ 表示用户的预测得分, $r_{u,i}$ 表示用户的实际评分。

4.3.2. 分类准确性指标

准确率(Accuracy) [29]是指推荐系统正确推荐的电影数占全部电影数量的比例, 它反映了推荐系统的整体效果。准确率的计算公式是:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

其中, TP (True Positive)表示用户实际感兴趣且被推荐的电影数, TN (True Negative)表示用户实际不感兴趣且未被推荐的电影数, FP (False Positive)表示用户实际不感兴趣但被推荐的电影数, FN (False Negative)表示用户实际感兴趣但未被推荐的电影数。

精确率(Precision) [30]是指推荐系统推荐给用户的电影中用户实际感兴趣的电影的比例, 它反映了推荐系统的用户满意度。精确率的计算公式是:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (10)$$

4.3.3. 可扩展性指标

在评估推荐系统的可扩展性时, 本文主要关注两个关键指标: 算法的执行时间和所需邻居的数量 (Neighbor-Used)。算法执行时间越短, 意味着推荐系统能够在实际应用中更迅速地为目标用户提供推荐。Neighbor-Used 则反映了为进行预测而需要保存到数据库中的邻居数量。所需的邻居数量越少, 占用的系统资源也就越少, 从而表明推荐算法具有更好的可扩展性。

4.4. 实验结果

4.4.1. 预测准确性指标对比

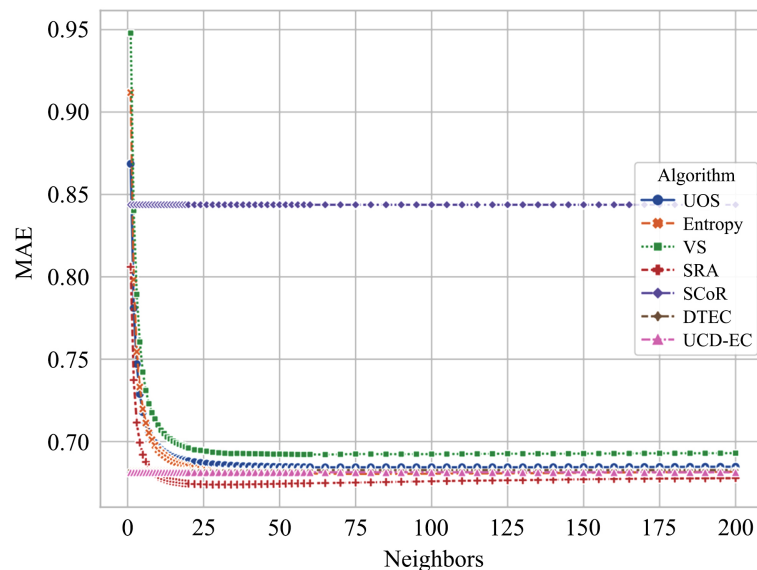


Figure 2. MAE comparison result of UCD-EC and other algorithms

图 2. UCD-EC 与其他算法的 MAE 对比结果

图 2 表示为各个算法的 MAE 值。MovieLens 25M 数据集内 UCD-EC 取得 MAE 最优值为 0.683。SRA 是表现最好的算法，但 SRA 牺牲了空间换取预测准确性，而 UCD-EC 算法提供较好准确性的同时，还有着良好的可扩展性。

4.4.2. 分类准确性指标对比

图 3、图 4 分别展示了不同算法的分类准确性评价结果，其中图 2 表示的是 Accuracy 指标，图 3 表示的是 Precision 指标。UCD-EC 的 Accuracy 值为 0.65，在所有算法中表现最优，Precision 值为 0.67，仅次于 SCoR 算法。基于 UCD-EC 在 Accuracy、Precision 的表现，UCD-EC 对比除了 SCoR 以外的其他算法能够更好分辨用户喜爱的电影，但是 SCoR 在准确度、空间和时间的性能指标表现均落后于 UCD-EC。

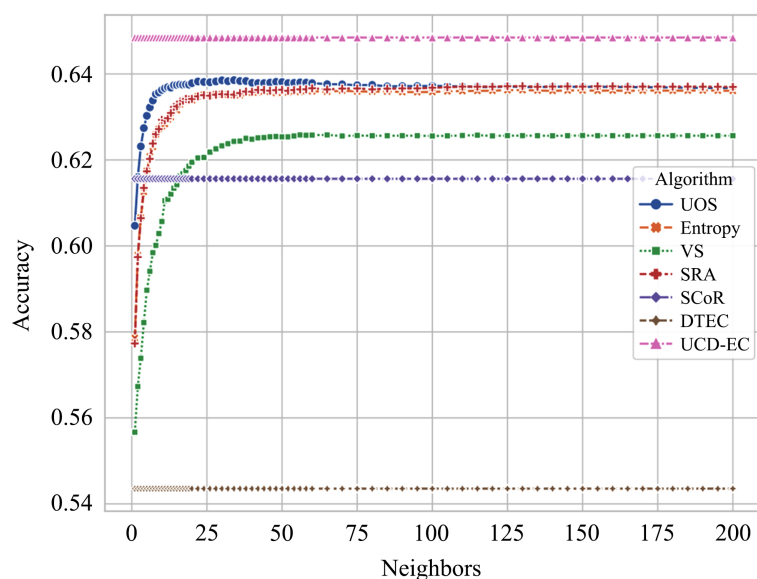


Figure 3. Accuracy comparison result of UCD-EC and other algorithms

图 3. UCD-EC 与其他算法的 Accuracy 对比结果

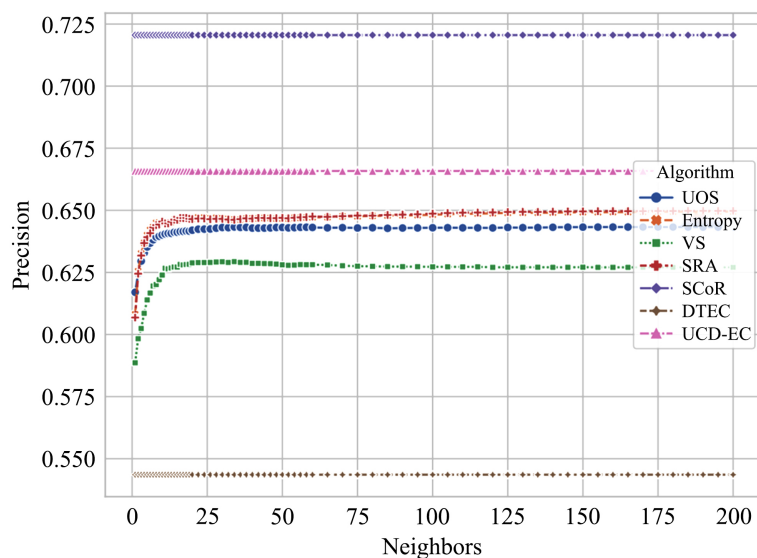


Figure 4. Precision comparison result of UCD-EC and other algorithms

图 4. UCD-EC 与其他算法的 Precision 对比结果

4.4.3. 可扩展性指标对比

图 5 表示每个算法在预测过程中对测试集中所有目标使用的平均邻居数。其中 SCoR、DTEC 和 UCD-EC 算法使用固定的用户数量进行预测，所以不会随着数据集中邻居用户数量的增加而变化。其余算法的曲线接近重合，这些算法选择的邻居数量相似，不同的是相似性的度量值。相比于其他算法，UCD-EC 平均只需要 15 个用户，可以选取更少的用户对目标用户的喜好物品进行预测与推荐，这加快了距离部分的计算速度，有效降低了算法的空间复杂度，进而提升了算法的可扩展性。

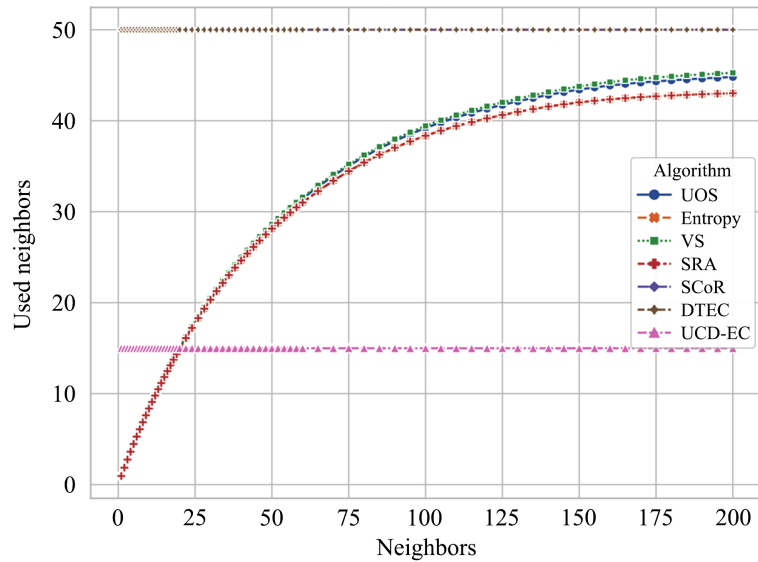


Figure 5. The actual number of neighbors for prediction and the number of neighbors the algorithm plans to choose

图 5. 用于预测的实际邻居数与算法计划选择的邻居数

图 6 表示每个算法在数据集中运行所需要的时间。由图中可知，UCD-EC 算法比 SCoR、DTEC 算法所需要的时间少，而比 UOS、Entropy、VS、SRA 算法所需的时间稍长。整体而言，UCD-EC 在运行时间指标的表现位于中等，可以相对快速地把物品推荐给目标用户，使得用户反馈变快。

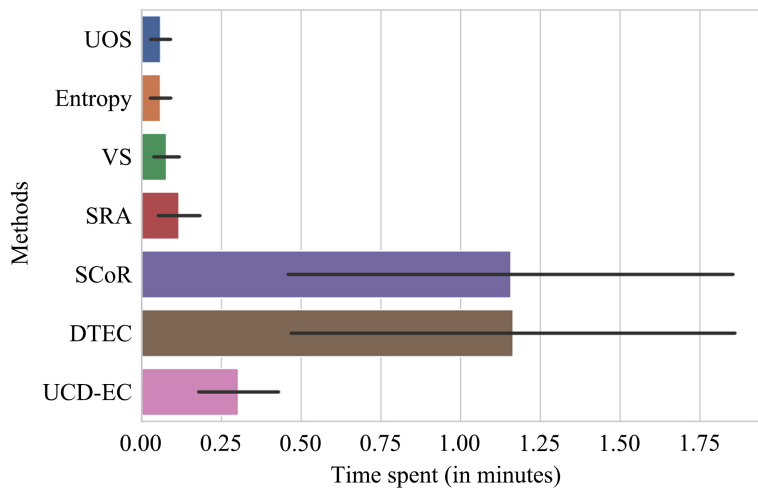


Figure 6. Time Spent comparison result of UCD-EC and other algorithms

图 6. UCD-EC 与其他算法的执行时间对比结果

5. 总结

针对推荐系统准确性不足和可扩展性低的问题, 本文提出了空间排布的方法, 针对用户和评分距离建立空间模型进行了探索, 将空间模型和推荐系统进行有效结合, 同时研究了推荐算法在邻居选择时的特殊情况, 结合相似性网络建模来改善推荐性能, 最后还设计了误差修正方法用以提高推荐算法的准确性。通过实验, 发现本文所提出的 UCD-EC 算法在预测准确性、分类准确性、可扩展性方面都位于前列。这说明利用用户相似性、坐标距离和误差修正的手段, 进一步挖掘推荐系统的潜在信息, 在仅增加少量时间的情况下, 提高预测的准确度, 更精确的找到用户喜欢的类别。

在未来研究中, 本文将专注于优化 UCD-EC 的运行时间。尽管 UCD-EC 优于同类型的 SCoR 和 DTEC 算法, 但仍有可以提高的空间。在未来的工作中, 研究人员会选择其他方式筛选邻居, 或是对空间分布模型进行改进以提高 UCD-EC 的运行效率。

参考文献

- [1] Jalili, M., Ahmadian, S., Izadi, M., *et al.* (2018) Evaluating Collaborative Filtering Recommender Algorithms: A Survey. *IEEE Access*, **6**, 74003-74024. <https://doi.org/10.1109/ACCESS.2018.2883742>
- [2] Yu, J., Yin, H., Xia, X., *et al.* (2023) Self-Supervised Learning for Recommender Systems: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, **36**, 335-355. <https://doi.org/10.1109/TKDE.2023.3282907>
- [3] Li, D., Lv, Q., Shang, L., *et al.* (2017) Efficient Privacy-Preserving Content Recommendation for Online Social Communities. *Neurocomputing*, **219**, 440-454. <https://doi.org/10.1016/j.neucom.2016.09.059>
- [4] Kim, J.C., Jeong, H.G. and Lee, S.W. (2021) Simultaneous Target Classification and Moving Direction Estimation in Millimeterwave Radar System. *Sensors*, **21**, Article 5228. <https://doi.org/10.3390/s21155228>
- [5] Xiong, R., Wang, J., Zhang, N., *et al.* (2018) Deep Hybrid Collaborative Filtering for Web Service Recommendation. *Expert Systems with Applications*, **110**, 191-205. <https://doi.org/10.1016/j.eswa.2018.05.039>
- [6] 赵永生, 祁云嵩. 基于改进相似度计算方法的协同过滤算法研究[J]. 计算机与数字工程, 2021, 49(3): 447-450, 541.
- [7] 江水. 基于协同过滤技术推荐系统的探究[J]. 计算机技术与发展, 2021, 31(11): 1-7.
- [8] Robillard, M., Walker, R. and Zimmermann, T. (2010) Recommendation Systems for Software Engineering. *IEEE Software*, **27**, 80-86. <https://doi.org/10.1109/MS.2009.161>
- [9] Li, N. and Li, C. (2009) Accumulative Influence Weight Collaborative Filtering Recommendation Approach. In: Chien, B.C. and Hong, T.P., Eds., *Opportunities and Challenges for Next-Generation Applied Intelligence*, Springer, Berlin, 73-78. https://doi.org/10.1007/978-3-540-92814-0_12
- [10] Hu, P., Yang, E., Pan, W., *et al.* (2022) Federated One-Class Collaborative Filtering via Privacy-Aware Non-Sampling Matrix Factorization. *Knowledge-Based Systems*, **253**, Article ID: 109441. <https://doi.org/10.1016/j.knosys.2022.109441>
- [11] Wang, Q., Wu, S., Bai, Y., *et al.* (2023) Neighbor Importance-Aware Graph Collaborative Filtering for Item Recommendation. *Neurocomputing*, **549**, Article ID: 126429. <https://doi.org/10.1016/j.neucom.2023.126429>
- [12] Xiao, Z.Y., *et al.* (2023) Generative Artificial Intelligence GPT-4 Accelerates Knowledge Mining and Machine Learning for Synthetic Biology. *ACS Synthetic Biology*, **12**, 2973-2982. <https://doi.org/10.1021/acssynbio.3c00310>
- [13] Batmaz, Z., Yurekli, A., Bilge, A. and Kaleli, C. (2019) A Review on Deep Learning for Recommender Systems: Challenges and Remedies. *Artificial Intelligence Review*, **52**, 1-37. <https://doi.org/10.1007/s10462-018-9654-y>
- [14] Lika, B., Kolomvatsos, K. and Hadjiefthymiades, S. (2014) Facing the Cold Start Problem in Recommender Systems. *Expert Systems with Applications*, **41**, 2065-2073
- [15] Zeng, C., Xing, C.X., Zhou, L.Z. and Zheng, X.H. (2004) Similarity Measure and Instance Selection for Collaborative Filtering. *International Journal of Electronic Commerce*, **8**, 115-129. <https://doi.org/10.1080/10864415.2004.11044314>
- [16] Shardanand, U. and Maes, P. (1995) Social Information Filtering: Algorithms for Automating "Word of Mouth". *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Denver, 7-11 May 1995, 210-217. <https://doi.org/10.1145/223904.223931>
- [17] Mana, S.C. and Sasipraba, T. (2021) Research on Cosine Similarity and Pearson Correlation Based Recommendation

- Models. *Journal of Physics*, **1770**, Article ID: 012014. <https://doi.org/10.1088/1742-6596/1770/1/012014>
- [18] Bag, S., Kumar, S.K. and Tiwari, M.K. (2019) An Efficient Recommendation Generation Using Relevant Jaccard Similarity. *Information Sciences*, **483**, 53-64. <https://doi.org/10.1016/j.ins.2019.01.023>
- [19] Dabek, F., Cox, R., Kaashoek, F., et al. (2004) Vivaldi: A Decentralized Network Coordinate System. *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, Portland, 30 August-3 September 2004, 15-26. <https://doi.org/10.1145/1015467.1015471>
- [20] Panagiotakis, H., Panagiotakis, C. and Fragopoulou, P. (2017) SCoR: A Synthetic Coordinate Based Recommender System. *Expert Systems with Application*, **79**, 8-19. <https://doi.org/10.1016/j.eswa.2017.02.025>
- [21] Song, S., Chaudhuri, K. and Sarwate, A.D. (2015) Learning from Data with Heterogeneous Noise Using SGD. arXiv: 1412.5617.
- [22] Panagiotakis, C., Papadakis, H., Papagrigoriou, A., et al. (2021) Improving Recommender Systems via a Dual Training Error Based Correction Approach. *Expert Systems with Applications*, **183**, Article ID: 115386. <https://doi.org/10.1016/j.eswa.2021.115386>
- [23] Doods, S., Bellogín, A., Pessemier, T.D., et al. (2016) A Framework for Dataset Benchmarking and Its Application to a New Movie Rating Dataset. *ACM Transactions on Intelligent Systems and Technology*, **7**, 1-28. <https://doi.org/10.1145/2751565>
- [24] He, X.S., Zhou, M.Y., Zhuo, Z., et al. (2015) Predicting Online Ratings Based on the Opinion Spreading Process. *Physica A: Statistical Mechanics and Its Applications*, **436**, 658-664. <https://doi.org/10.1016/j.physa.2015.05.066>
- [25] Lee, S. (2020) Using Entropy for Similarity Measures in Collaborative Filtering. *Journal of Ambient Intelligence and Humanized Computing*, **11**, 363-374. <https://doi.org/10.1007/s12652-019-01226-0>
- [26] Ai, J., Cai, Y., Su, Z., et al. (2022) Predicting User-Item Links in Recommender Systems Based on Similarity-Network Resource Allocation. *Chaos, Solitons and Fractals*, **158**, Article ID: 112032. <https://doi.org/10.1016/j.chaos.2022.112032>
- [27] Su, Z., Zheng, X., Ai, J., et al. (2020) Link Prediction in Recommender Systems Based on Vector Similarity. *Physica A: Statistical Mechanics and Its Applications*, **560**, Article ID: 125154. <https://doi.org/10.1016/j.physa.2020.125154>
- [28] Willmott, C.J. and Matsuura, K. (2005) Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE) in Assessing Average Model Performance. *Climate Research*, **30**, 79-82. <https://doi.org/10.3354/cr030079>
- [29] Javed, U., Shaukat, K., Hameed, I.A., et al. (2021) A Review of Content-Based and Context-Based Recommendation Systems. *International Journal of Emerging Technologies in Learning*, **16**, 274-306. <https://doi.org/10.3991/ijet.v16i03.18851>
- [30] Madadipouya, K. and Chelliah, S. (2017) A Literature Review on Recommender Systems Algorithms, Techniques and Evaluations. *BRAIN. Broad Research in Artificial Intelligence and Neuroscience*, **8**, 109-124.