

# 基于模糊偏好向量和物品质量的推荐算法

艾均, 孙浩, 苏湛, 周越

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1月28日; 录用日期: 2024年2月18日; 发布日期: 2024年4月11日

## 摘要

推荐系统能够快速有效地帮助人们在海量信息中进行筛选和过滤, 为用户智能化地推荐其感兴趣的物品, 因此具有重要的理论意义和广泛的应用价值。针对现有研究中推荐误差和推荐列表排序可以进一步优化的需求。本文应用模糊隶属度函数以计算了多维物品品类向量上的用户偏好, 并基于用户偏好特征和物品品类向量计算用户间相似性, 然后, 进一步基于相似性和用户相关性筛选构建用户相似性网络, 利用用户相似性网络中的社区信息, 对预测过程进行加权。同时, 利用统计方法过滤物品的异常评分以度量物品质量排序, 利用物品质量排序的优化结果辅助预测的误差修正。基于MovieLens的小型数据集进行了对比试验, 与时下流行的几个算法进行对比, 本文设计的算法有效地提高了预测准确性及推荐排序准确性, 实验结果表明了利用用户模糊偏好、物品质量排序、相似性网络及误差修正手段, 可以进一步挖掘推荐系统地潜在信息, 在仅增加少量计算量的情况下, 进一步提升了系统性能, 并且保持了系统的可解释性。

## 关键词

推荐系统, 用户模糊偏好, 物品质量排序, 相似性网络, 误差修正

# Recommendation Algorithm Based on Fuzzy Preference Vector and Item Quality

Jun Ai, Hao Sun, Zhan Su, Yue Zhou

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

Received: Jan. 28<sup>th</sup>, 2024; accepted: Feb. 18<sup>th</sup>, 2024; published: Apr. 11<sup>th</sup>, 2024

## Abstract

Recommendation systems can quickly and effectively help people filter and sift through massive

amounts of information, intelligently recommending items of interest to users. Therefore, they have significant theoretical importance and wide application value. This paper addresses the need for further optimization of recommendation errors and recommendation list ranking in existing research. The fuzzy membership function is applied to calculate user preferences on multi-dimensional item category vectors. Based on user preference features and item category vectors, user similarity is calculated. Then, further based on similarity and user relevance, a user similarity network is constructed by filtering and selection. The community information in the user similarity network is used to weight the prediction process. At the same time, statistical methods are used to filter out abnormal ratings of items to measure item quality ranking, and the optimized results of item quality ranking are used to assist in the correction of prediction errors. A comparative experiment was conducted on a small dataset from MovieLens, and the algorithm designed in this paper was compared with several popular algorithms. The results show that the designed algorithm effectively improves prediction accuracy and recommendation ranking accuracy. The experimental results demonstrate that by using user fuzzy preferences, item quality ranking, similarity network, and error correction methods, the potential information of the recommendation system can be further mined. With only a small amount of additional computation, system performance is further improved, and the system's interpretability is maintained.

## Keywords

Recommendation Systems, User Fuzzy Preference, Item Quality Ranking, Similarity Network, 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

推荐系统能对海量信息进行有效过滤, 为用户推荐能够满足其兴趣和需求的信息。它根据用户偏好对信息进行排序, 保留用户感兴趣的内容。缓解了信息过载带来的挑战, 使用户在面对海量信息时不必花费过多的时间和精力。

推荐系统是一种信息过滤系统。它利用用户的历史记录、搜索记录和人口统计信息等因素, 预测用户的兴趣和需求, 从而帮助用户发现可能感兴趣的物品, 如音乐、商品和电影等。因此, 推荐系统在当今信息化社会中具有重要的理论研究意义和广泛的应用价值, 许多科学家和工程师们都在对推荐系统进行不懈的研究, 并取得了令人瞩目的成果。

目前, 推荐系统根据其工作方法主要分为三种。包括基于内容的推荐算法[1]、协同过滤推荐算法[2]和混合推荐算法[3]。基于内容的推荐算法是利用物品的属性和特征, 向用户推荐与他们过去喜欢的物品相似的物品。协同过滤推荐算法是利用用户之间的相似性, 向用户推荐与其相似性高的其他用户喜欢的物品。混合推荐算法结合了基于内容的推荐算法和协同过滤推荐算法, 利用了不同算法的优势, 提高推荐系统的效果。

其中, 协同过滤推荐算法因其高可解释性、易于实现、便于训练的特点, 仍然是推荐系统领域备受关注 and 广泛研究的主题[4], 各种相似性度量方法被不断提出[5], 例如多级模糊相似性[6]和共振相似性[7]等。此外, 经典相似性算法及其改进算法, 也提升了协同过滤算法的性能, 如余弦相似性[8]和距离加权余弦相似性度量法[9]; 皮尔逊相似性[10]及基于物品频率改进的皮尔逊相似性[11]。

在协同过滤算法中,应用了模糊逻辑(Fuzzy Logic)的推荐系统称为模糊逻辑推荐系统[12],大部分模糊逻辑的应用都集中在扩展经典的相似性度量方法。例如 Wang 等人[13], Surya Kant 等人[14]的研究工作,主要思想是将用户对物品的评分转化为模糊偏好,然后用模糊偏好来替代评分进行后续的计算。

此外,许多学者都对协同过滤方法进行了不同的改进。文献[15]构建了复杂网络模型进行链接预测,将复杂网络的重要元素被抽象为节点,元素之间的相互作用被表示为边,提升了链路预测的准确度。Jiao 等人[16]考虑物品质量(Item Quality)和用户评分倾向性对用户打分的影响。利用物品质量评估算法实现了用户评分修正,改进了查找与用户兴趣相似的最近邻过程,并在此基础上进行推荐。

随着互联网应用与服务的不断增多,推荐系统仍然面临着非常多的挑战和问题。包括预测或排序准确性不高、难以同时提高准确性与多样性、算法的可扩展性有限、冷启动问题、稀疏数据处理以及用户隐私保护等[17]。

本文针对启发式的协同过滤算法仍存在评分预测准确性和推荐列表排序有待提升的需求,研究了如何结合用户模糊偏好向量、物品质量、相似性网络及误差修正手段,以提升评分预测准确性和排序性能。

## 2. 相关工作

模糊逻辑善于处理现实世界中缺乏精确表示的概念或信息[18]。大多数协同过滤推荐算法都依赖用户对物品的评分,然而用户通常会表现出模糊或不确定的情感,例如一般或者没有感觉。这些模糊的情感,受用户主观影响较小,有助于改进用户相似性计算、邻居选择并减少预测误差。Zhang 等人[19]将模糊方法用在推荐系统上,减轻了用户和物品数据中的各种不确定性。本文利用模糊隶属度函数将用户评分转换成用户模糊偏好,旨在更准确地表示用户偏好,减轻了用户的主观性对相似性计算造成的影响。

相似性网络(Similarity Network)利用物品或用户之间的相似性关系进行推荐。通过计算物品或用户之间的相似性,建立一个网络结构,实现个性化推荐。通过将推荐系统建模成复杂网络,可以借助图论特性和复杂网络理论,更深入地分析用户、物品以及评分之间的关系。Yan 等人[20]提出了一种链接预测算法,该算法基于社区结构。他们认为,相较于不同社区的节点,同一社区内的节点更可能有联系。本文构建了一个以用户为节点,用户间相似性为边的无向网络模型,旨在探索相似用户之间的潜在特征信息并提高推荐质量,并利用网络中的社区信息,对预测过程进行加权。

异常检测(Anomaly Detection)是一种可以有效帮助算法识别和处理异常数据的技术。评分数据的质量会限制协同过滤推荐系统的性能。具体来说,Ekstrand 等人[21]指出评分收集过程可能会有意外发生,因此评分可能包含噪声。他们提到,这种噪声,先前在[22]中被称为自然噪声,可能是由人为错误、不确定性和其他因素引起的,检测和修正这种噪声应该能提供更准确的推荐。本文利用统计方法过滤物品的异常评分以度量物品质量排序,再利用物品质量排序的优化结果辅助预测的误差修正。

通过上文可以看出,运用模糊逻辑能够更准确地表示用户偏好、对物品质量进行分析和优化能够给予用户高质量的推荐、使用相似性网络能够利用网络拓扑结构获得节点间隐藏的信息、异常检测方法能够减小数据集中噪声对评分预测的影响,并更加准确地度量物品质量。这些方法都已经学者将其运用在推荐系统领域,并且都在一定程度上提升了推荐系统的性能。因此,本文的研究旨在通过探索一种将上述方法相结合的新方法,即基于模糊偏好向量和物品质量的推荐算法(Recommendation Algorithm Based on Fuzzy Preference Vector and Item Quality, FPVIQ)来提升评分预测准确性并得到更好的排序性能。

## 3. 方法

### 3.1. 方法概述

如图 1 所示,本文提出的算法是一种基于模糊偏好向量和物品质量的推荐算法。该算法首先将用户

对商品的评分转化为“喜欢”和“不喜欢”的模糊偏好。其次，将物品转换为以品类为元素的物品品类向量，将其与模糊偏好相结合，提出了结合二者的相似性计算公式，并基于该相似性建立用户相似性网络，根据社区划分结果提取社区信息。同时利用统计方法过滤物品的异常评分以度量物品质量排序，并基于正常评分集利用遗传算法优化最佳物品质量排序，基于所生成的物品质量与真实评分得到用户最佳误差，最终进行预测和推荐。

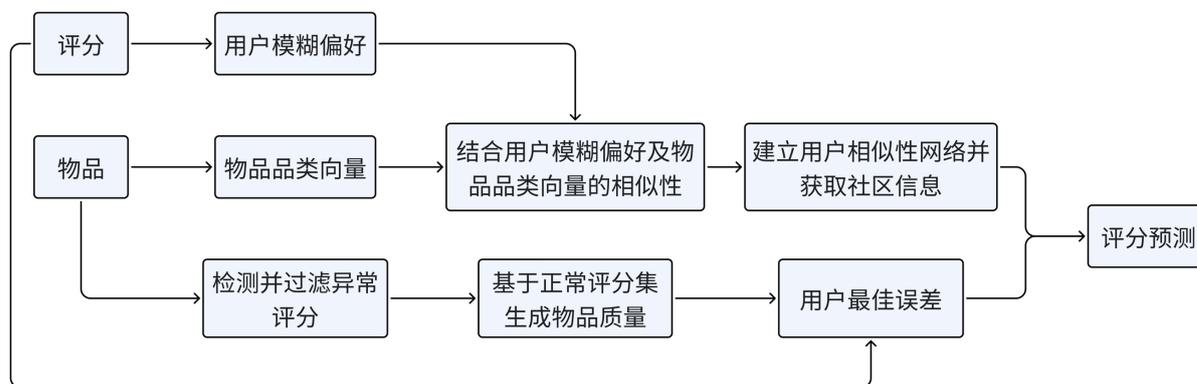


Figure 1. Algorithm structure diagram

图 1. 算法结构图

### 3.2. 评分模糊化

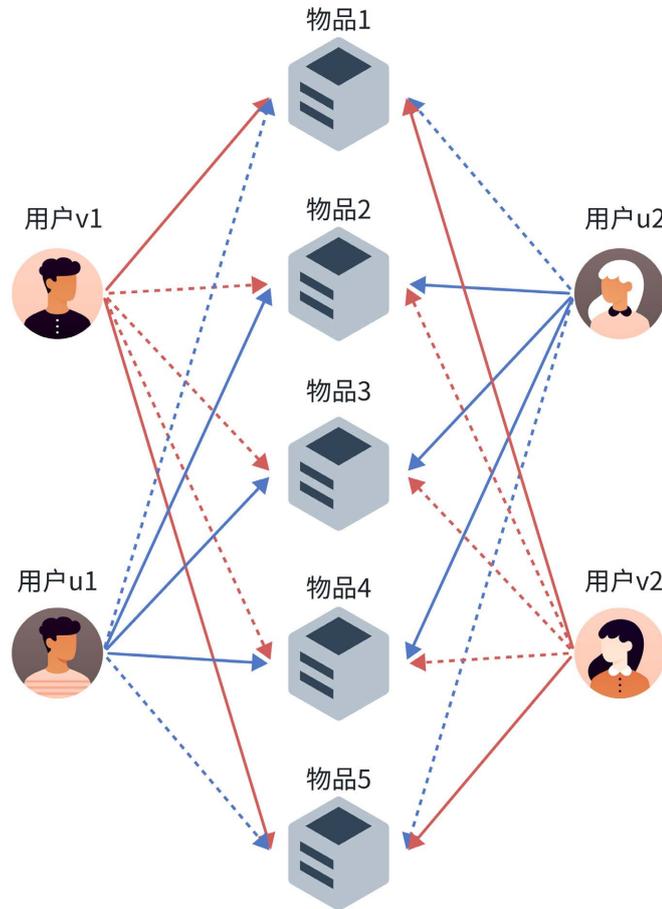
本文对评分矩阵中的每个用户进行分析，利用模糊隶属度函数以获得每个目标用户对每个物品的模糊偏好。在实际生活中，用户对物品的感受往往是模糊和不确定的，这些模糊的感受通常受到更少的用户主观因素的影响，因此会更加准确。所以，它帮助算法更准确地计算用户的相似性，选出真正相似的邻居，从而降低预测误差。具体来说，本文利用公式(1)和公式(2)计算出相应的喜欢指数  $fp^{like}$  和不喜欢指数  $fp^{dislike}$ ，提高了用户相似性计算的客观性、逻辑性和准确性，从而减少了最终分数预测的误差。本文将用户对物品的模糊偏好分为“喜欢”和“不喜欢”，通过比较每个用户对物品的评分与它们各自的评分平均值，确定了他们的模糊偏好。如果用户对某个物品的评分高于该物品的平均评分，将其视为“喜欢”；否则，视为“不喜欢”。

$$fp^{like} = \begin{cases} \mu_1(r_{ui}), & r_{ui} > \bar{r}_u \\ 0, & r_{ui} \leq \bar{r}_u \end{cases} \quad (1)$$

$$fp^{dislike} = \begin{cases} 0, & r_{ui} > \bar{r}_u \\ \mu_2(r_{ui}), & r_{ui} \leq \bar{r}_u \end{cases} \quad (2)$$

其中， $r_{ui}$  表示用户  $u$  对物品  $i$  的评分， $\bar{r}_u$  表示用户  $u$  给出的平均评分， $\mu_1(r_{ui})$  表示用户  $u$  对物品  $i$  的模糊喜欢指数，而  $\mu_2(r_{ui})$  表示模糊不喜欢指数。

图 2 展示了不同用户对不同物品的偏好。实线和虚线分别代表“喜欢”偏好和“不喜欢”偏好。例如，用户  $u_1$  和  $u_2$  都喜欢物品 2、3 和 4，但不喜欢物品 1 和 5。类似地，用户  $v_1$  和  $v_2$  都喜欢物品 1 和 5，但不喜欢物品 2、3 和 4。基于这些偏好，可以确定用户  $u_1$  和  $u_2$  之间的相似性是高于用户  $u_1$  和其他三名用户的相似性的。类似地，用户  $v_1$  和  $v_2$  之间的相似性是高于用户  $v_1$  和其他三名用户的相似性的。在获得用户对物品的偏好后，利用公式(3)和公式(4)来获得用户对物品的模糊偏好指数。下一步是将该指数与物品的品类向量结合，以生成模糊偏好向量。



**Figure 2.** Preferences of different users for different items  
**图 2.** 不同用户对不同物品的偏好

$$\mu_1(r_{ui}) = \begin{cases} 1, & r_{ui} = r_u^{\max} \\ \frac{r_{ui} - \bar{r}_u}{r_u^{\max} - \bar{r}_u}, & \bar{r}_u \leq r_{ui} < r_u^{\max} \\ 0, & r_{ui} < \bar{r}_u \end{cases} \quad (3)$$

$$\mu_2(r_{ui}) = \begin{cases} 1, & r_{ui} = r_u^{\min} \\ \frac{\bar{r}_u - r_{ui}}{\bar{r}_u - r_u^{\min}}, & r_u^{\min} \leq r_{ui} < \bar{r}_u \\ 0, & r_{ui} > \bar{r}_u \end{cases} \quad (4)$$

其中， $r_u^{\min}$  和  $r_u^{\max}$  分别表示用户  $u$  评分的最低和最高分。由于用户对一个物品的偏好只能是“喜欢”或“不喜欢”， $\mu_1(r_{ui})$  和  $\mu_2(r_{ui})$  是互斥的，并且会根据用户对物品的偏好有一个值为 0。

### 3.3. 物品向量化

在推荐系统研究中，相似性计算方法对目标用户评分预测和推荐结果的准确性有很大影响。以往关于模糊偏好的研究是通过将两个用户共同评分的物品的喜爱指数和不喜爱指数的皮尔逊相似性取平均值，以获得用户相似性。然而，该方法将用户相似性视为一个标量，忽略了物品详细属性的差异对用户相似性

计算的影响。

以用户  $u$  和用户  $v$  为例，这两个用户都对体育视频表现出模糊的喜爱偏好。根据以前的研究，推荐系统会认为这两个用户非常相似。然而，当预测一个关于特定运动的视频时，用户  $u$  和  $v$  的偏好可能会因视频的具体属性而不同，例如用户  $u$  喜欢篮球而用户  $v$  喜欢足球。如果使用之前的算法进行评分预测和推荐，结果可能会有明显的偏差。

本文提出的算法旨在更有效地利用评分和物品的详细属性信息来应对评分预测和推荐挑战。假设数据集中的物品有  $k$  种类别，则生成包含所有类型的  $k$  维品类向量  $L$ ，即  $L = \{l_1, l_2, \dots, l_t, \dots, l_k\}$ 。图 3 展示了物品品类向量的示意图。每个物品的品类向量长度为  $k$ ，将物品中存在的品类对应的元素设置为 1，其余元素设置为 0，从而生成所有物品的原始品类向量。



Figure 3. Item genre vector  
图 3. 物品的品类向量

### 3.4. 结合用户模糊偏好及物品品类向量的相似性

在获得物品品类向量后，本文应用了结合用户模糊偏好及物品品类向量的相似性度量方法，它有效地利用了之前计算出的模糊偏好指数，将用户相似性进一步划分为喜欢和不喜欢。最后，对两个相似性进行平均，得到最终的用户相似性。对于任意两个用户  $u$  和  $v$ ，相似性是一个  $k$  维向量，定义如公式 5 所示：

$$s_{uv} = \{s_1, s_2, \dots, s_t, \dots, s_k\} \quad (5)$$

其中， $\{s_1, s_2, \dots, s_t, \dots, s_k\}$  分别表示用户  $u$ 、 $v$  在品类  $\{l_1, l_2, \dots, l_t, \dots, l_k\}$  上的相似性。为了便于区分，本文将用户相似性向量  $s_{uv}$  称为用户之间的全局相似性，将每个品类上的用户相似性  $s_i$  称为用户之间的品类相似性。

将用户对物品的评分转化为一对模糊偏好指数，改变了传统的主观评分分值，并基于这两个模糊偏好指数计算用户之间的全局相似性。为此，首先使用公式(6)、公式(7)和公式(8)分别计算用户之间的喜欢指数和不喜欢指数的相似性。然后使用公式 9 计算两个相似性的平均值，得到用户之间的全局相似性。

$$T' = \left\{ \frac{1}{\sum_{i=1}^n g_{ui} g_{vi} \cdot l_{i1}}, \frac{1}{\sum_{i=1}^n g_{ui} g_{vi} \cdot l_{i2}}, \dots, \frac{1}{\sum_{i=1}^n g_{ui} g_{vi} \cdot l_{ik}} \right\} \quad (6)$$

$$s_{uv}^{like} = T' \circ \sum_{i=1}^n \left[ g_{ui} g_{vi} \left[ 1 - |\mu_1(r_{ui}) - \mu_1(r_{vi})| \right] \cdot L_i \right] \quad (7)$$

$$s_{uv}^{dislike} = T' \circ \sum_{i=1}^n [g_{ui} g_{vi} [1 - |\mu_1(r_{ui}) - \mu_2(r_{vi})|]] \cdot L_i \quad (8)$$

$$s_{uv} = \frac{s_{uv}^{like} + s_{uv}^{dislike}}{2} \quad (9)$$

其中,  $n$  表示用户  $u$  和  $v$  共同评过分的物品数量,  $g_{ui}$  和  $g_{vi}$  分别表示用户  $u$  和用户  $v$  是否对物品  $i$  进行了评分。如果用户  $u$  和  $v$  都对物品  $i$  给出了评分, 那么  $g_{ui}$  和  $g_{vi}$  等于 1; 否则,  $g_{ui}$  和  $g_{vi}$  等于 0。  $L_i$  是物品  $i$  的品类向量,  $l_{i1}$  是品类向量中的第一个元素。“ $\circ$ ”表示向量的哈达玛积。如果存在一个品类没有被任何用户  $u$  和  $v$  共同评分, 那么包含相应品类的  $T'$  向量中的元素的分子为零, 这也使得公式(7)和(8)中哈达玛积右侧的向量的相应元素变为零。 $\mu_1(r_{ui})$  和  $\mu_2(r_{vi})$  分别表示用户  $u$  对物品  $i$  的喜欢和不喜欢指数。最后, 使用公式(9)得到用户  $u$  和  $v$  之间的全局相似性。

### 3.5. 建立用户相似性社区网络

本文算法依赖于邻居用户对目标物品的评分。邻居用户被定义为与目标用户共享相同评分物品的用户。使用邻居用户对目标物品的评分和他们与目标用户的全局相似性, 可以预测目标用户对目标物品的评分。本文选取的策略为只保留与目标用户皮尔逊相似性大于 0 的邻居。对于同一用户的不同目标物品可能包含不同的品类, 从而导致其邻居对这些项目的模糊偏好和品类相似性不同。因此, 当预测评分时, 用户和其邻居之间的目标物品的相似性会根据项目的品类变化。为了简化计算, 本文使用公式 10 将向量全局相似性转换为标量。

$$s_{uvi} = \sqrt{\sum_{t=1}^k (s_{uvt})^2} \quad (10)$$

其中,  $s_{uvt}$  表示用户  $u$  和  $v$  在品类  $l_t$  上的相似性。如果物品  $i$  不包含品类  $l_t$ , 则  $s_{uvt}$  等于 0。 $s_{uvi}$  表示目标用户的每个邻居对最终预测的影响, 作为预测值中的权重。

本文运用复杂网络建模方法建立了用户相似性网络, 从而在相似性的基础上提供更多信息以用来辅助推荐。在推荐系统领域的研究中, 由于复杂网络能够有效分析大量多维数据, 因而常常被研究人员应用于各类推荐类任务中。社区信息在最后的评分预测公式中会作为权重, 参与到评分预测中。

### 3.6. 获取用户最佳误差

异常检测方法可以通过分析用户的行为模式和评分分布, 识别出与正常模式偏离较大的评分。可以在一定程度上消除噪声的干扰。为了提高推荐的准确性, 本文利用了统计方法检测并过滤物品的异常评分以度量物品质量排序。然后利用遗传算法进一步优化物品质量排序, 在本文实验中, 遗传算法会进行 30 轮迭代或直到满足适应度函数的要求才停止, 适应度函数选取排序准确性[23] (Sorting Accuracy, SA), 其判断方式为公式 11。SA 关注推荐列表中各物品的绝对位置, 而不是相对位置, 是一个更严格的检验排序准确性的指标。假设在一个推荐列表中, 若某个物品的排名与其真实评分在此列表中的排名相同, 则可判断这个物品处在此列表中正确的位置上。具体地, 对排序正确的物品取 1、错误的取 0。

$$OS(S_i) = \prod_{j=i+1}^{|L|} I(S_i \geq S_{j+1}) \quad (11)$$

其中,  $s_i$  为推荐列表中, 用户对第  $i$  个位置的物品的实际评分,  $I$  为指示函数, 即当括号内的条件满足时, 函数值取 1, 反之取 0。 $|L|$  为推荐列表包含的物品数量,  $OS(S_i)$  为推荐列表中第  $i$  个位置的物品的判断值。

对于某个推荐列表, 如果其中物品的排序与用户对这些物品的评分排序越接近则认为推荐系统的 SA

值越高。因此 SA 值越高代表推荐系统越能把更准确的物品优先推荐给用户。SA 值能反映推荐系统推荐的总体正确率，在计算完推荐列表中每个物品的判断值后，再利用公式 12 即可得到此列表的 SA 值。

$$SA = \frac{\sum_{i=1}^{|L|} OS(S_i)}{|L|} \quad (12)$$

其中，分子可解释为推荐列表中物品排序正确的总个数，分母为当前推荐列表的长度。

利用遗传算法对所有物品质量排序优化后，通过公式 14 计算每名用户的最佳误差，该误差是通过优化用户真实评分与物品质量之间的平均绝对误差[24] (Mean Absolute Error, MAE)所得到并起到辅助预测的作用。MAE 的计算公式如公式(13)所示。

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{r}_i - r_i| \quad (13)$$

其中， $n$  表示测试集中的样本数， $\hat{r}_i$  表示用户  $u$  对测试集中物品  $i$  的预测评分， $r_i$  表示用户  $u$  对测试集中物品  $i$  的实际评分。

$$bias_u = \min_e MAE(q_i + e, r_{ui} - \bar{r}_u) \quad (14)$$

其中， $r_{ui}$  表示用户  $u$  对物品  $i$  的评分， $\bar{r}_u$  表示用户  $u$  给出的平均评分。 $q_i$  表示物品  $i$  的质量， $e$  表示要求的误差值，它在-3 到 3 之间递增，步长为 0.02，所求的  $bias_u$  即使上述函数最小的  $e$  值。

### 3.7. 评分预测

利用得到的社区信息对预测公式进行加权并加上用户的最佳误差得到最终的预测公式(15)进行评分预测。

$$\hat{r}_{ui} = \bar{r}_u + \frac{\sum_{v=1}^N [(r_{vi} - \bar{r}_v) \cdot s_{uvi} \cdot \lambda]}{\sum_{v=1}^N s_{uvi}} + bias_u \quad (15)$$

其中， $\bar{r}_u$  和  $\bar{r}_v$  分别表示用户  $u$  和  $v$  给出的平均评分， $N$  表示与目标用户  $u$  的邻居数量， $\lambda$  表示社区相似性权重。如果两个用户属于同一个社区， $\lambda$  设为 1，否则设为 0.7。 $bias_u$  表示目标用户的最佳偏差。

综上所述，本文提出的算法具体过程为算法 1 所示：

算法 1 基于模糊偏好向量和物品质量推荐算法

输入：用户对集合  $U$ ，评分集合  $R$ ，物品集合  $I$ ，用户物品对集合  $Q$ ，预测目标用户物品对集合  $P$

输出：用户  $u$  对目标物品  $i$  的预测评分  $\hat{r}_{ui}$

- 1: FOR 每个用户物品对  $(u, i) \in Q$  DO
- 2: 计算  $\bar{r}_u$ 、 $r_u^{\max}$ 、 $r_u^{\min}$
- 3: 计算用户模糊偏好
- 4: END FOR
- 5: FOR 每个物品  $i \in I$  DO
- 6: 生成原始品类向量  $T_i$
- 7: 计算物品评分标准差并获得异常评分集
- 8: 过滤异常评分并得到正常评分集
- 9: 基于正常评分集生成物品质量  $q_i$
- 10: 计算用户最佳误差  $bias_u$
- 11: END FOR

```

12: FOR 每个用户对  $(u, v) \in U$  DO
13:   获取共同评分物品集  $I_{uv}$ 
14:   FOR 每个物品  $i \in I_{uv}$  DO
15:     计算  $s_{uv}^{like}$ 、 $s_{uv}^{dislike}$ 、 $s_{uv}$ 、 $s_{uvi}$ 
16:   END FOR
17: END FOR
18: 利用  $s_{uvi}$  建立用户相似性网络  $G$ 
19: FOR 每对预测目标  $p_{ui} \in P$  DO
20:   取皮尔逊相似性大于 0 的用户作为邻居集  $V$ 
21:   根据公式 15 计算  $\hat{r}_{ui}$ 
22: END FOR
23: RETURN 用户  $u$  对目标物品  $i$  的预测评分  $\hat{r}_{ui}$ 

```

## 4. 实验

### 4.1. 数据集

本文采用了 MovieLens [25] 的小型数据集 ml-latest-small 进行实验。该数据集是由 GroupLens Research 收集和维持的数据集。它包含了 610 名用户对 9742 部电影的 100,836 条评分记录。最低为 0.5 分，最高为 5 分，增量为 0.5 分。

### 4.2. 实验设计

实验使用 FSharp 编程语言实现了本文涉及到的所有算法，并使用了一致的缓存技术和计算流程。实验采用的验证方法是折五验证，其基本原理是将数据集随机划分为五个相等大小的子集，每次使用其中四个子集作为训练集，剩下一个子集作为验证集，然后重复这个过程五次，使得每个子集都充当过一次验证集。最后将每次验证的结果进行平均，作为模型的性能评估指标。

本文实验是在一台配备了 Intel(R) Core(TM) i7-10700K CPU、所用的显卡为 NVIDIA GeForce RTX 4070 GPU、内存容量 16 GB、操作系统为 64 位 Windows 10 的计算机上进行的。所有的实验均是在上述的实验环境中进行的，使本文能够在相同的条件下对实验结果进行公正的比较。对于所有算法，采用的实验参数也均相同，并分别在邻居数量为 1 至 200 的情况下进行了实验。

### 4.3. 对比算法

#### 4.3.1. 相似性资源分配(Similarity-Network Resource Allocation, SRA)

SRA 算法[26]通过建立基于用户物品关系的相似性网络，计算度中心度和社区结构来有效地识别准确的邻居。使用资源分配来平衡程度中心性，并使用归一化度值和社区来修正预测结果。

#### 4.3.2. 向量相似性(Vector Similarity, VS)

向量相似性[27]引入了一种新的方法来使用基于向量的度量来计算用户相似性，该方法考虑了基于物品属性的多个维度。定义了全局相似性、局部相似性和元相似性，建立了用户相似性的向量指标，其特点是相似性会随着目标物品的不同而不同。

#### 4.3.3. 信息熵协同过滤 Entropy

用户评分的信息熵反映了用户对物品的整体评分行为。Soojung Lee 在他的研究[28]中将用户评分的

信息熵与推荐算法相结合，并基于皮尔逊相似性对其进行了改进。研究认为，在计算两个用户之间的相似性时，如果他们共同评分的物品熵值越高，他们的相似性就越大，否则，相似性结果就会越小。类似地，当一个物品的熵较小时，评分差异越大，用户之间的相似性就越低。

#### 4.3.4. 巴氏 Bhattacharyya Coefficient

巴氏相似性[10]的特点是考虑用户对之间所有评分的分值，而并非共同评分物品。同时考虑局部相似性和全局相似性，并且允许两个用户共同评分物品集为空。

#### 4.4. 评价指标

本文选取了共 5 个指标来做度量，分别是平均绝对误差(MAE)、均方根误差[24] (Root Mean Square Error, RMSE)、准确率[29] (Accuracy)、F1 值[30] (F1-score)及半衰期效用[23] (Half Life Utility, HLU)。这些指标的计算公式和含义如下：

平均绝对误差(MAE)和均方根误差(RMSE)是协同过滤推荐系统中广泛使用的两个评价指标。其中 MAE 衡量的是预测评分与用户给出的实际评分之间的平均绝对误差。可以通过所有用户的平均 MAE 来反映系统的整体性能。而 RMSE 则强调较大的误差值对预测精度的影响。两个误差度量参数均为越低越好。MAE 的计算公式在上文已给出，RMSE 的计算方法如公式(16)所示：

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (r_i - \hat{r}_i)^2} \quad (16)$$

其中， $n$  表示测试集中的样本数， $\hat{r}_i$  表示用户  $u$  对测试集中物品  $i$  的预测评分， $r_i$  表示用户  $u$  对测试集中物品  $i$  的实际评分。

准确率(Accuracy)是指推荐系统正确推荐的物品数占全部物品数量的比例，它反映了推荐系统的整体效果，其值越高越好。准确率的计算公式如公式(17)所示：

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (17)$$

其中， $TP$  表示算法推荐用户喜欢的物品总数， $FP$  表示算法推荐用户不喜欢的物品总数， $FN$  表示算法未推荐用户喜欢的物品总数， $TN$  表示算法未推荐用户不喜欢的物品总数。

F1 值(F1-score)是精确率(Precision)和召回率(Recall)的调和平均数，它综合考虑了两者的影响，是一种平衡指标。Precision 和 Recall 的计算公式如公式 18 和公式 19 所示，F1 值越高，说明推荐系统的性能越好。F1 值的计算公式如公式 20 所示：

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (18)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (19)$$

$$\text{F1-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (20)$$

半衰期效用(HLU)是指特定物质的浓度在一定反应后降至其初始值的一半所需的时间。在推荐列表中，排名较高的物品更有可能被用户注意到，导致排名较低的物品被查看的概率迅速下降。该指标考虑了用户浏览产品的概率与其在推荐列表中的位置之间的指数关系，使其能够衡量用户的实际评分与系统预测评分之间的差异程度，其值越高越好。目标用户的 HLU 计算公式如公式(21)所示：

$$HLU_u = \sum_{i=1}^N \frac{\max(r_{ui} - \bar{r}_u, 0)}{2^{i-1}} \quad (21)$$

其中,  $h$  表示半衰期的阈值。

要获得推荐系统的半衰期结果, 需要利用公式(22)计算所有用户的半衰期值。值越高, 表示系统的排序性能越好。在本文研究中, HLU 是一个基于排名的评价指标, 在实验中, 半衰期的阈值  $h$  设置为 2。

$$HLU_{total} = \frac{\sum_{u=1}^m HLU_u}{m} \quad (22)$$

其中,  $m$  表示用户个数。

#### 4.5. 实验结果

图 4 展示了本文所提出的 FPVIQ 算法与各算法的 MAE 实验结果对比图。可以看出 FPVIQ 算法的 MAE 值始终保持最低, 效果好于其他四个算法, MAE 最小值约为 0.6736。比 Entropy 算法 0.6861 降低了约 1.82%, 比巴氏算法 0.6935 降低了约 2.87%, 比 VS 算法 0.6919 降低了约 2.64%, 比 SRA 算法 0.6803 降低了约 0.98%。

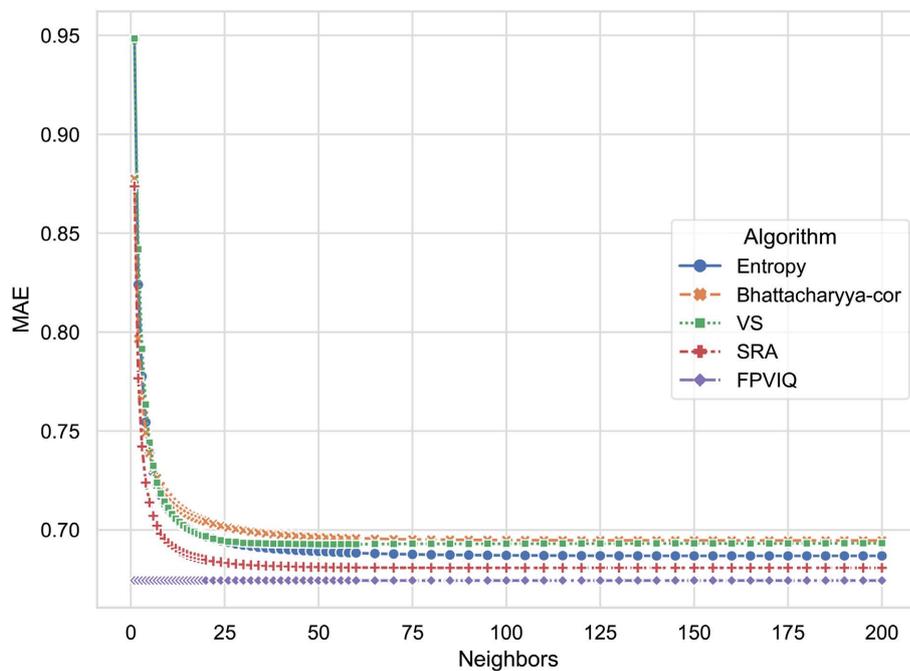


Figure 4. MAE comparison results

图 4. MAE 对比结果

图 5 展示了本文所提出的 FPVIQ 算法与各算法的 RMSE 实验结果对比图。可以看出 FPVIQ 算法的 RMSE 值始终保持最低, 效果好于其他四个算法, RMSE 最小值约为 0.8859。比 Entropy 算法 0.8971 降低了约 1.27%, 比巴氏算法 0.9128 降低了约 2.95%, 比 VS 算法 0.9016 降低了约 1.73%, 比 SRA 算法 0.8863 降低了约 0.05%。

图 6 展示了本文所提出的 FPVIQ 算法与各算法的 Accuracy 实验结果对比图。可以看出 FPVIQ 算法的 Accuracy 值始终保持最高, 效果好于其他四个算法, Accuracy 值约为 0.639。比 Entropy 算法 0.6306

提高了约 1.33%，比巴氏算法 0.6378 提高了约 0.19%，比 VS 算法 0.6272 提高了约 2.02%，比 SRA 算法 0.6305 提高了约 1.36%。

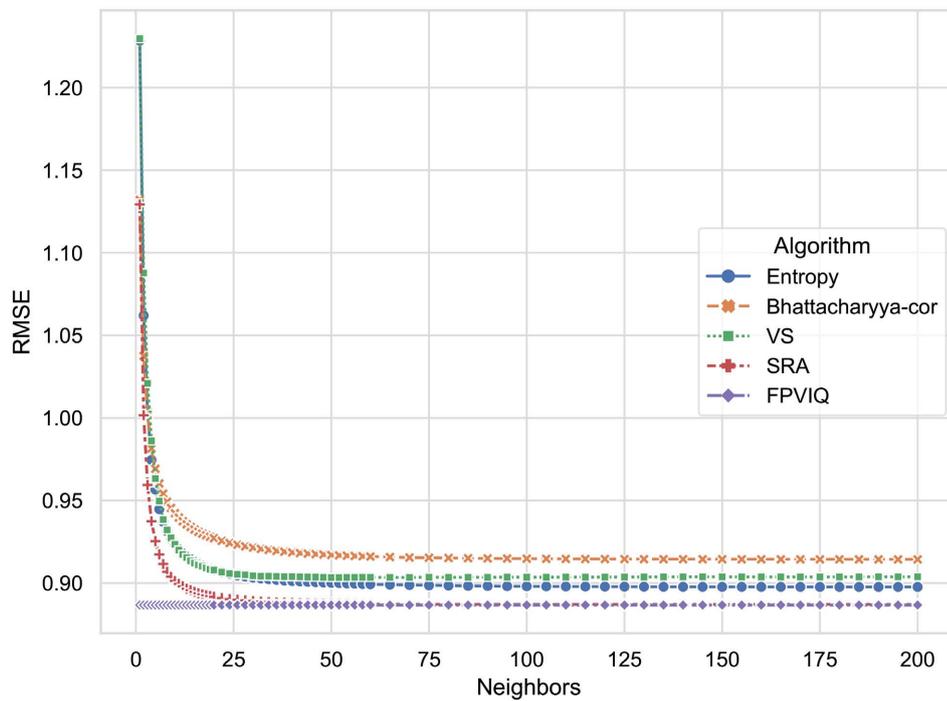


Figure 5. RMSE comparison results

图 5. RMSE 对比结果

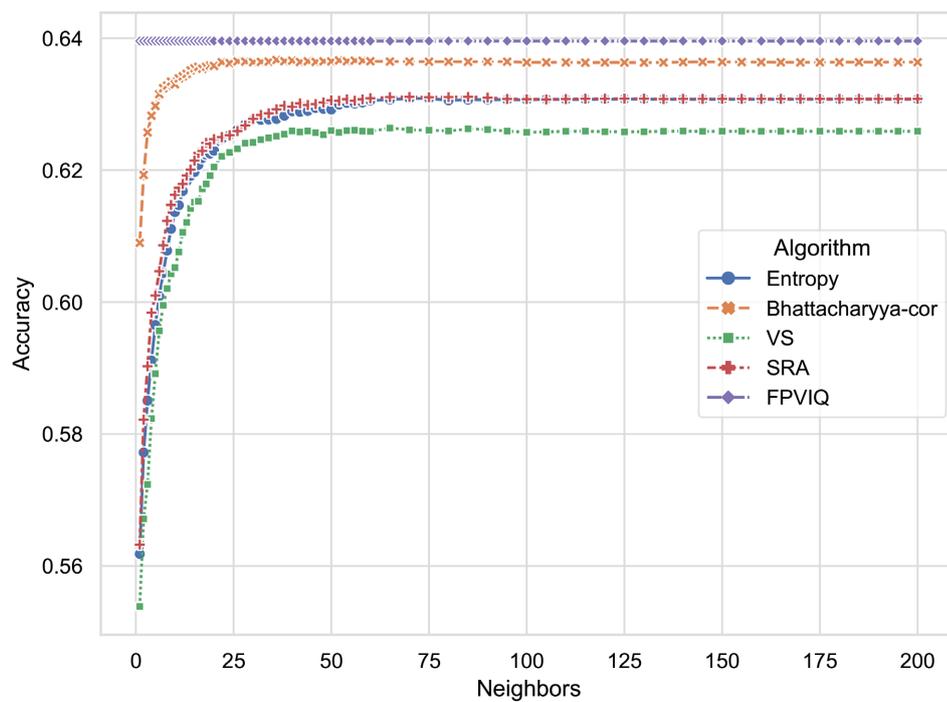


Figure 6. Accuracy comparison results

图 6. Accuracy 对比结果

图 7 展示了本文所提出的 FPVIQ 算法与各算法的 F1 值实验结果对比图。可以看出 FPVIQ 算法的 F1 值始终保持最高,效果好于其他四个算法,F1 最大值约为 0.6984。比 Entropy 算法 0.6773 提升了约 3.12%,比巴氏算法 0.6954 提升了约 0.43%,比 VS 算法 0.6917 提升了约 0.96%,比 SRA 算法 0.6774 提升了约 3.1%。

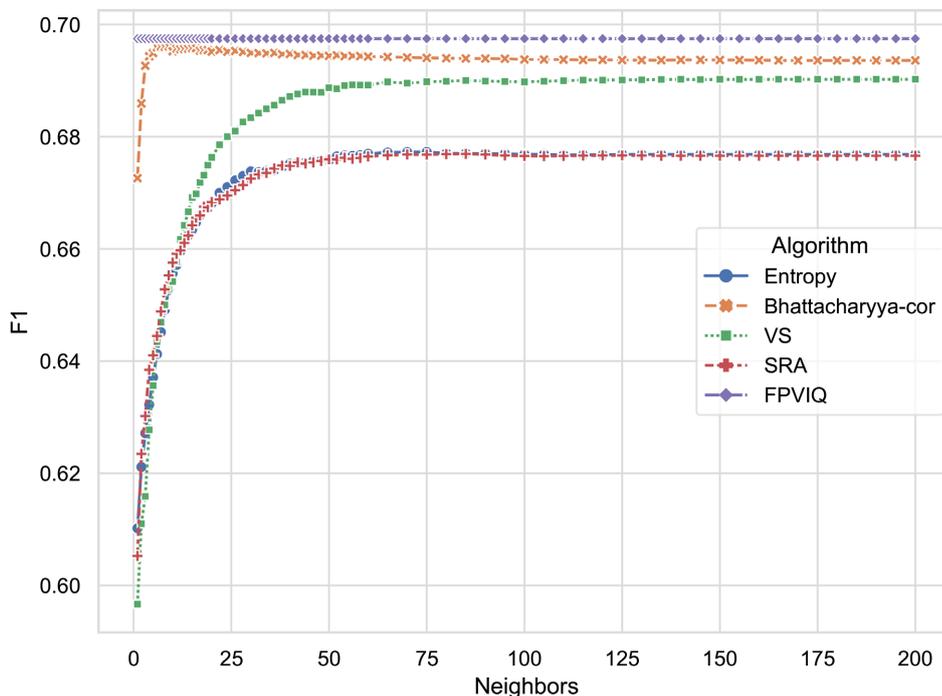


Figure 7. F1-score comparison results  
图 7. F1 值对比结果

图 8 展示了本文所提出的 FPVIQ 算法与各算法的 HLU 实验结果对比图。可以看出 FPVIQ 算法的 HLU 值稳定在约 1.0644,排名第二,略低于排名第一的巴氏算法的 1.0985 降约 3.1%,相较 Entropy 算法的最大值 1.0595 提升了约 0.46%,相较 VS 算法的最大值 0.9819 提升了约 8.41%,相较 SRA 算法的最大值 1.0528 提升了约 1.1%。

#### 4.6. 实验分析

通过实验可以发现,在评分预测误差方面,FPVIQ 算法在 MAE、RMSE 两个度量指标上均排名第一,表现最好。这表明 FPVIQ 算法能够提高预测的准确度,预测结果与真实结果之间的误差较小。

在分类准确性方面,FPVIQ 算法在 Accuracy、F1 值两个指标上排名第一,这表明 FPVIQ 算法比其他算法能够更好地分辨用户喜爱的物品,准确地找到用户更喜欢的类别。

在排序准确性方面,FPVIQ 算法在 HLU 指标上排名第二,这表明 FPVIQ 算法在推荐系统中,会更优先地将用户喜欢的物品放在推荐列表的前列。相较于其他推荐算法来说,FPVIQ 算法能够提升用户对推荐系统的满意程度。

### 5. 结论

针对现有研究中推荐误差和推荐列表排序可以进一步优化的需求。本文应用了模糊隶属度函数以计算多维物品品类向量上的用户偏好,并基于用户偏好特征和物品品类向量计算用户间相似性,然后进一

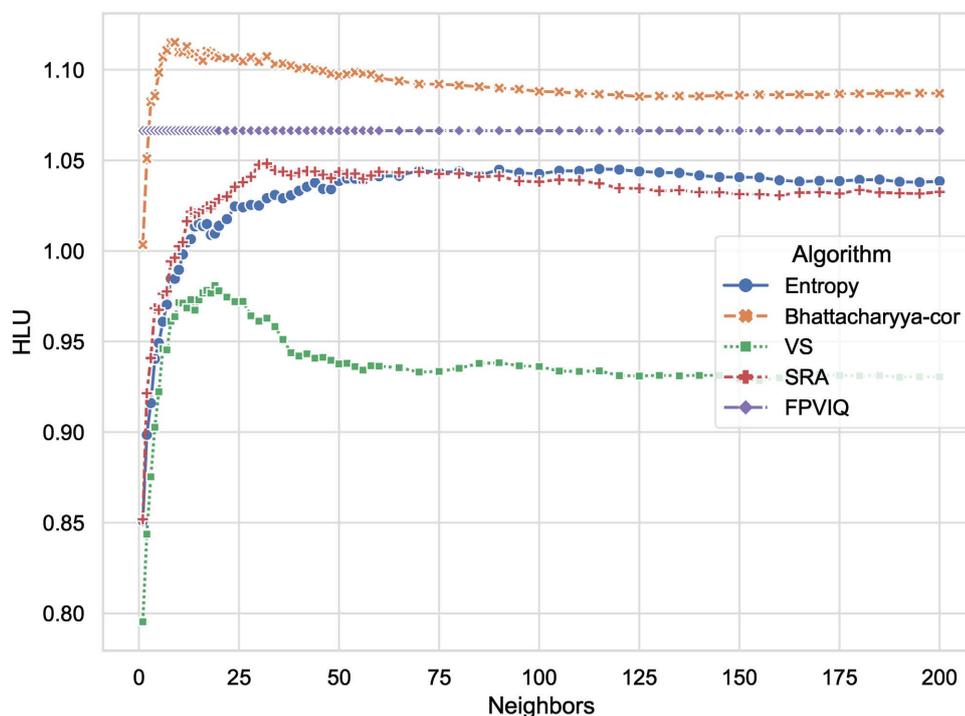


Figure 8. HLU comparison results  
图 8. HLU 对比结果

步基于相似性和用户相关性筛选构建用户相似性网络，利用用户相似性网络中的社区信息，对预测过程进行加权。同时，利用统计方法过滤物品的异常评分以度量物品质量排序，利用物品质量排序的优化结果辅助预测的误差修正。通过实验，发现本文提出的 FPVIQ 算法对比算法在预测误差和推荐排序准确性方面均有不同地提升。实验结果表明了利用用户模糊偏好、物品质量排序、相似性网络及误差修正手段，可以进一步挖掘推荐系统地潜在信息，在仅增加少量计算量的情况下，进一步提升了系统性能，并且保持了系统的可解释性。

在后续的工作中将尝试更多不同的模糊隶属度函数，进一步减少用户的主观影响，提高相似性计算准确度，并将尝试不同的误差修正方法，提升系统的排序性能。

## 基金项目

国家自然科学基金项目(61803264)。

## 参考文献

- [1] Aggarwal, C.C. (2016) An Introduction to Recommender Systems. In: Aggarwal, C.C., Ed., *Recommender Systems*, Springer, Cham, 1-28. [https://doi.org/10.1007/978-3-319-29659-3\\_1](https://doi.org/10.1007/978-3-319-29659-3_1)
- [2] Roy, D. and Dutta, M. (2022) A Systematic Review and Research Perspective on Recommender Systems. *Journal of Big Data*, 9, Article No. 59. <https://doi.org/10.1186/s40537-022-00592-5>
- [3] Lekakos, G. and Caravelas, P. (2008) A Hybrid Approach for Movie Recommendation. *Multimedia Tools and Applications*, 36, 55-70. <https://doi.org/10.1007/s11042-006-0082-7>
- [4] Zarzour, H., Al-Sharif, Z., Al-Ayyoub, M. and Jararweh, Y. (2018) A New Collaborative Filtering Recommendation Algorithm Based on Dimensionality Reduction and Clustering Techniques. 2018 9th International Conference on Information and Communication Systems (ICICS), Irbid, 3-5 April 2018, 102-106. <https://doi.org/10.1109/IACS.2018.8355449>

- [5] Gazdar, A. and Hidri, L. (2020) A New Similarity Measure for Collaborative Filtering Based Recommender Systems. *Knowledge-Based Systems*, **188**, Article ID: 105058. <https://doi.org/10.1016/j.knosys.2019.105058>
- [6] Shojaei, M. and Saneifar, H. (2021) MFSR: A Novel Multi-Level Fuzzy Similarity Measure for Recommender Systems. *Expert Systems with Applications*, **177**, Article ID: 114969. <https://doi.org/10.1016/j.eswa.2021.114969>
- [7] Tan, Z. and He, L. (2017) An Efficient Similarity Measure for User-Based Collaborative Filtering Recommender Systems Inspired by the Physical Resonance Principle. *IEEE Access*, **5**, 27211-27228. <https://doi.org/10.1109/ACCESS.2017.2778424>
- [8] Schwarz, M., Lobur, M. and Stekh, Y. (2017) Analysis of the Effectiveness of Similarity Measures for Recommender Systems. 2017 14th International Conference the Experience of Designing and Application of CAD Systems in Microelectronics (CADSM), Lviv, 21-25 February 2017, 275-277. <https://doi.org/10.1109/CADSM.2017.7916133>
- [9] Li, B. and Han, L. (2013) Distance Weighted Cosine Similarity Measure for Text Classification. In: Yin, H., et al., Eds., *Intelligent Data Engineering and Automated Learning—IDEAL 2013*, Springer, Berlin, 611-618. [https://doi.org/10.1007/978-3-642-41278-3\\_74](https://doi.org/10.1007/978-3-642-41278-3_74)
- [10] Davoudi, A. and Chatterjee, M. (2018) Social Trust Model for Rating Prediction in Recommender Systems: Effects of Similarity, Centrality, and Social Ties. *Online Social Networks and Media*, **7**, 1-11. <https://doi.org/10.1016/j.osnem.2018.05.001>
- [11] Zhang, F., Zhou, W., Sun, L., et al. (2017) Improvement of Pearson Similarity Coefficient Based on Item Frequency. 2017 International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR), Ningbo, 9-12 July 2017, 248-253. <https://doi.org/10.1109/ICWAPR.2017.8076697>
- [12] Yadav, D.K. and Katarya, R. (2018) Study on Recommender System Using Fuzzy Logic. 2018 Second International Conference on Computing Methodologies and Communication (ICCMC), Erode, 15-16 February 2018, 50-54. <https://doi.org/10.1109/ICCMC.2018.8487669>
- [13] Wang, W., Lu, J. and Zhang, G. (2014) A New Similarity Measure-Based Collaborative Filtering Approach for Recommender Systems. In: Wen, Z. and Li, T., Eds., *Foundations of Intelligent Systems*, Springer, Berlin, 443-452. [https://doi.org/10.1007/978-3-642-54924-3\\_42](https://doi.org/10.1007/978-3-642-54924-3_42)
- [14] Kant, S., Mahara, T., Jain, V.K. and Jain, D.K. (2019) Fuzzy Logic Based Similarity Measure for Multimedia Contents Recommendation. *Multimedia Tools and Applications*, **78**, 4107-4130. <https://doi.org/10.1007/s11042-017-5260-2>
- [15] 白梅, 裴仕涵, 王习特. 基于位置的路网 Skyline 查询处理研究[J]. 计算机工程, 2022, 48(1): 127-134.
- [16] 焦富森, 李树青. 基于物品质量和用户评分修正的协同过滤推荐算法[J]. 数据分析与知识发现, 2019, 3(8): 62-67.
- [17] 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](https://doi.org/10.1007/978-3-540-92814-0_12)
- [18] Yera, R. and Martinez, L. (2017) Fuzzy Tools in Recommender Systems: A Survey. *International Journal of Computational Intelligence Systems*, **10**, 776-803. <https://doi.org/10.2991/ijcis.2017.10.1.52>
- [19] Zhang, Z., Lin, H., Liu, K., et al. (2013) A Hybrid Fuzzy-Based Personalized Recommender System for Telecom Products/Services. *Information Sciences*, **235**, 117-129. <https://doi.org/10.1016/j.ins.2013.01.025>
- [20] Yan, B. and Gregory, S. (2012) Finding Missing Edges in Networks Based on Their Community Structure. *Physical Review E*, **85**, Article ID: 056112. <https://doi.org/10.1103/PhysRevE.85.056112>
- [21] Ekstrand, M.D., Riedl, J.T. and Konstan, J.A. (2011) Collaborative Filtering Recommender Systems. *Foundations and Trends® in Human-Computer Interaction*, **4**, 81-173. <https://doi.org/10.1561/1100000009>
- [22] O'Mahony, M.P., Hurley, N.J. and Silvestre, G.C.M. (2006) Detecting Noise in Recommender System Databases. *Proceedings of the 11th International Conference on Intelligent User Interfaces*, Sydney, 29 January-1 February 2006, 109-115. <https://doi.org/10.1145/1111449.1111477>
- [23] 苏湛, 陈学谦, 艾均, 等. 基于用户相似性选择及标签距离的推荐算法[J]. 应用科学学报, 2023, 41(6): 940-957.
- [24] Peng, M., Zeng, G., Sun, Z., et al. (2018) Personalized App Recommendation Based on App Permissions. *World Wide Web*, **21**, 89-104. <https://doi.org/10.1007/s11280-017-0456-y>
- [25] Salloum, S. and Rajamanthri, D. (2021) Implementation and Evaluation of Movie Recommender Systems Using Collaborative Filtering. *Journal of Advances in Information Technology*, **12**, 189-196. <https://doi.org/10.12720/jait.12.3.189-196>
- [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 & 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] 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>
- [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] Al-Badarnah, I., Habib, M., Aljarah, I. and Faris, H. (2022) Neuro-Evolutionary Models for Imbalanced Classification Problems. *Journal of King Saud University-Computer and Information Sciences*, **34**, 2787-2797. <https://doi.org/10.1016/j.jksuci.2020.11.005>