

《智能信息处理》课程作业

## 基于用户兴趣概念格的推荐评分预测

刘冬帅

作业	分数[20]
得分	

2020 年 11 月 13 日

# 基于用户兴趣概念格的推荐评分预测

刘冬帅

(大连海事大学 信息科学技术学院, 辽宁省大连市 中国 116026)

**摘 要:** 为了解决传统的基于协同过滤推荐, 算法中存在的数据稀疏的问题, 本文提出了一种基于用户兴趣概念格的推荐评分预测(RRP-UICL)方法. 该方法首先通过用户兴趣概念格将“最近邻”分为直接“最近邻”和间接“最近邻”两类, 然后采用不同方法分别计算直接“最近邻”和间接“最近邻”与目标用户的相似度. 最后, 通过相似度值计算目标用户的不可见项目评分值, 在两个稀疏噪声数据集上进行了实验. 实验结果表明, 在数据稀疏的情况下, 该方法具有更好的预测精度和良好的性能.

**关键词:** 用户兴趣; 概念格; 稀疏数据; 协同过滤; 推荐系统;

## Recommendation Rating Prediction Based on User Interest Concept Lattice

Liu Dong Shuai

(School of Information Science and Technology, Dalian maritime university, Liaoning Dalian 116026, China)

**Abstract:** In order to solve the problem of sparse data in the traditional recommendation algorithm based on collaborative filtering, this paper proposes a recommendation rating prediction method based on user interest concept lattice. This method first divides the “nearest neighbor” into direct “nearest neighbor” and indirect “nearest neighbor” through the user interest concept lattice. Then two methods are used to calculate the similarity between the nearest neighbor and the target user respectively. Finally, the calculated similarity value is used to predict the invisible item score value of the target user. Experiments are carried out on two sparse noise data sets. The experimental results show that this method has better prediction accuracy and good performance under the condition of sparse data.

**Key words:** User interest; concept lattice; sparse data; collaborative filtering; recommend systems

## 1 概述

近年来, 随着互联网和信息技术的飞速发展, 推荐系统<sup>[1-3]</sup>被广泛应用于电子商务网站中. 推荐系统支持消费者从过载的信息中选择所需的产品或服务. 协同过滤(collaborative filtering, CF)<sup>[4]</sup>是最常用的推荐算法, 它利用用户的评分来构建用户-用户或项目项目的相似度指标, 并识别用户的邻域来生成推荐. 然而, CF推荐算法存在这样的问题: 当用户-项目评分矩阵非常稀疏, 用户只对少数项目进行评分时, CF推荐算法的推荐质量极大地受到影响<sup>[5]</sup>. Luo 等

人<sup>[6]</sup>提出将用户相似度分为局部用户相似度和全局用户相似度两部分来解决数据稀疏问题. 此外, Patra 等人<sup>[7]</sup>使用全局相似度和一个新的局部相似度来解决稀疏性的问题. 他们使用利用数据库中所有可用的评分来计算用户-用户或项目-项目之间的相似度. 张俊等人<sup>[8]</sup>根据兴趣向量来计算用户的兴趣相似性度, 并引入专家信任度来进行评分预测填充, 以此降低数据稀疏性. 林建辉等人<sup>[9]</sup>针对传统相似性度量只考虑用户对项目评分问题, 提出通过用户之间的相似性与信任关系来查找“最近邻”, 有效地缓解了稀疏性问题. Guo 等人<sup>[10]</sup>提出通过多类型辅助反馈信息

来解决稀疏性问题. Xu 等人<sup>[11]</sup>通过嵌入在用户评论中的形容词特征来解决数据稀疏性问题.

本文提出了一种基于用户兴趣概念格的推荐评分预测 (Recommendation Rating Prediction Based on User Interest Concept Lattice, 简称 RRP-UICL) 方法. 该方法通过用户兴趣概念格将“最近邻”分为直接“最近邻”和间接“最近邻”两类. 然后计算目标用户与直接“最近邻”和间接“最近邻”的相似度. 最后, 将计算所得的相似度值用于预测目标用户的不可见项目评分值, 解决了传统 CF 推荐算法的稀疏性问题.

## 2 相关介绍

### 2.1 基于邻域的协同过滤

基于邻域的协同过滤推荐算法<sup>[12]</sup>的主要目标是基于目标用户的邻居用户向其推荐项目, 根据计算相似度对象的不同而划分为基于用户和基于项目两类算法. 本文的研究基础为基于用户的算法, 主要包括查找“最近邻”和预测推荐两个步骤.

#### 2.1.1 查找“最近邻”

相似度度量可以找到目标用户的“最近邻”用户, 相似度度量考虑已评分项目的评分, 来计算目标用户和其他用户之间的相似度. 假设  $I_u$  为用户  $u$  评分过的项目集合,  $I_u \cap I_v$  为用户  $u$  和  $v$  都评分过的项目集合,  $r_{u,i}$  为用户  $u$  对项目  $i$  的评分,  $\bar{r}_u$  为用户  $u$  评分的平均值. 下面将介绍几种常用的相似度计算方法<sup>[7]</sup>.

1) 修正的余弦相似性 (Adjusted Cosine, ACos) 采用式 (1) 计算用户  $u$  与用户  $v$  之间的相似度.

$$Sim(u, v) = \frac{\sum_{i \in I_u \cap I_v} (r_{u,i} - \bar{r}_u) \cdot (r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{u,i} - \bar{r}_u)^2} \cdot \sqrt{\sum_{i \in I_u \cap I_v} (r_{v,i} - \bar{r}_v)^2}} \quad (1)$$

2) 皮尔逊相关 (Pearson Correlation, PC) 采用式 (2) 计算用户  $u$  与用户  $v$  之间的相似度.

$$Sim(u, v) = \frac{\sum_{i \in I_u \cap I_v} (r_{u,i} - \bar{r}_u) \cdot (r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{u,i} - \bar{r}_u)^2} \cdot \sqrt{\sum_{i \in I_u \cap I_v} (r_{v,i} - \bar{r}_v)^2}} \quad (2)$$

3) 受约束的皮尔逊相关受 (Constrained PC, CPC) 采用式 (3) 计算用户  $u$  与用户  $v$  之间的相似度.

$$Sim(u, v) = \frac{\sum_{i \in I_u \cap I_v} (r_{u,i} - r_{med}) \cdot (r_{v,i} - r_{med})}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{u,i} - r_{med})^2} \cdot \sqrt{\sum_{i \in I_u \cap I_v} (r_{v,i} - r_{med})^2}} \quad (3)$$

式中,  $r_{med}$  为评分等级的中位数.

4) Jaccard 采用式 (4) 计算用户  $u$  与用户  $v$  之间的 Jaccard 相似度.

$$Sim(u, v) = \frac{|I_u \cap I_v|}{|I_u \cup I_v|} \quad (4)$$

式中,  $|I_u \cap I_v|$  为用户  $u$  和用户  $v$  共同评价过的项目个数.

#### 2.1.2 预测推荐

向目标用户推荐项目, 首先需要预测未评分项目的评分, 然后推荐前  $N$  个高分项目. 使用式 (5) 预测目标用户  $u$  对项目  $i$  的评分.

$$R_{u,i} = \bar{r}_u + \frac{\sum_{v \in N_u} Sim(u, v) \cdot (r_{v,i} - \bar{r}_v)}{\sum_{v \in N_u} Sim(u, v)} \quad (5)$$

式中,  $Sim(u, v)$  为用户  $u$  与用户  $v$  的相似度,  $N_u$  为用户  $u$  的“最近邻”集合.

### 2.2 概念格

形式概念分析 (FCA) 理论是由 RWille 提出的, 它是一种数据分析和知识表示的方法. 形式概念分析的两大基本内容是形式背景和形式概念, 其基本的数据结构是概念格.

对于一个形式背景  $K=(G, M, I)$ , 其中,  $G$  是对象集合,  $M$  是属性集合,  $I$  是  $G$  和  $M$  之间的一个关系,  $I \subseteq G \times M$ . 在形式背景的对象集  $A \subseteq G$  和属性集  $B \subseteq M$  之间定义如下两个映射:

$$f(A) = \{b \in M \mid a \in A, aIb\}$$

$$g(B) = \{a \in G \mid b \in B, aIb\}$$

其中,  $f(A)$  是  $A$  中所有对象所共有的属性,  $g(B)$  是  $B$  中所有属性所共有的对象.

由形式背景  $K$  生成的概念格中的每个节点  $Z=(A, B)$  是一个二元组, 满足:  $A \subseteq G, B \subseteq M, f(A)=B, g(B)=A$  则二元组  $Z$  称为 [1] 形式概念, 其中,  $A$  是形式概念  $Z$  的外延, 记作  $Ext(Z)$ ,  $B$  是它的内涵, 记作  $Int(Z)$ .

若  $Z_1=(A_1, B_1)$  和  $Z_2=(A_2, B_2)$  是概念格中两个形式概念, 满足,  $A_1 \subseteq A_2 (B_1 \subseteq B_2)$ , 则称  $Z_1$  为  $Z_2$  的亚概念,  $Z_2$  为  $Z_1$  的超概念, 记为  $Z_1 \leq Z_2$ , 其中“ $\leq$ ”为一个偏序关系.

## 3 RP-UICL 方法

RRP-UICL 方法的主要步骤如图 1 所示. 从图 1 可以看出, RRP-UICL 方法主要分为两个模块: 一是“最近邻”模块, 二是推荐模块. 下面将介绍这两个模块的细节部分.

### 3.1 “最近邻”模块

#### 3.1.1 构造用户兴趣概念格

在这一阶段, 首先将用户对系统中已有项目的所有评分组成的评分矩阵  $R$  转化为二进制矩阵. 在二进制矩阵中, 评分较高的项目被认为是用户感兴趣的项目. 本研究是在 1~5 的评分等级上面进行的, 则将评分 4 和 5 视为最高评分, 其值在二进制矩阵中设置为

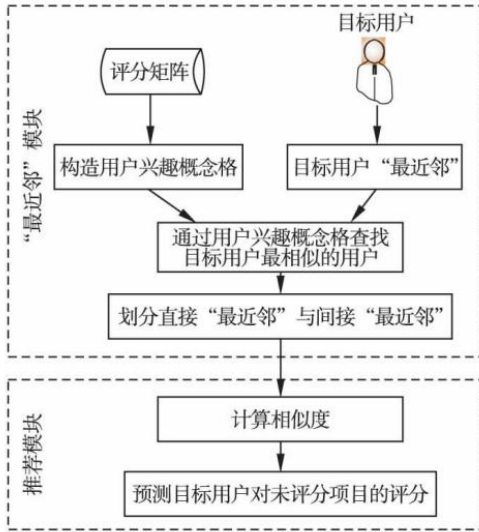


图1 RRP-UICL 方法模型

Fig.1 RRP-UICL method model

$\langle 1 \rangle$ , 其他值均设置为 $\langle 0 \rangle$ . 该二进制矩阵可以看作是一个用户兴趣形式背景  $K = (U, I, R)$ , 其中  $U$  是所有用户集合,  $I$  是所有项目集合,  $R$  是  $U$  和  $I$  之间的一个关系. 然后通过用户兴趣形式背景  $K$  可以构造用户兴趣概念格, 用  $L_c$  表示  $K$  的概念格, 图2给出了实例分析.

为了加快推荐过程, 需要删除用户兴趣概念格  $L_c$  中一些多余的形式概念, 本文定义了两个条件对形式概念进行删除: 对于一个形式概念  $Z$ :

1) 若  $Z \in L_c$ , 使得  $|\text{Ext}(Z)| = 1$ , 即可删除  $Z$ .

2) 若  $Z, Z' \in L_c$ , 使得  $\text{Int}(Z) \in \text{Int}(Z')$ , 即可删除  $Z$ . 图3显示了图2(c)删除多余形式概念后的结果, 为最终的用户兴趣概念格  $L_c$ .

图3显示了图2(c)删除多余形式概念后的结果, 为最终的用户兴趣概念格  $L_c$ .

### 3.1.2 划分直接“最近邻”和间接“最近邻”

在这一阶段, 首先需要查找目标用户的“最近邻”. 然后, 通过用户兴趣概念格来确定与目标用户最相似的用户. 最后, 将“最近邻”与最相似用户进行比较, 以此来划分直接“最近邻”和间接“最近邻”. 下面将详细介绍该过程:

在基于邻域的协同过滤方法中, 对相同项目具有相似偏好的用户被视为“最近邻”用户. 本文也采用了协同过滤方法来查找目标用户的“最近邻”, 选择与目标用户  $u$  兴趣相近的若干用户作为“最近邻”  $N_u$ . 以图2(a)的评分矩阵为例, 若向目标用户1推荐项目5, 从评分矩阵中可以看出, 用户  $U_2, U_3$  和  $U_6$  均参与了对项目5的评分, 这些用户都有可能向目标用户1推荐项目5, 则目标用户1的  $N_u$  为  $\{U_2, U_3, U_6\}$ .

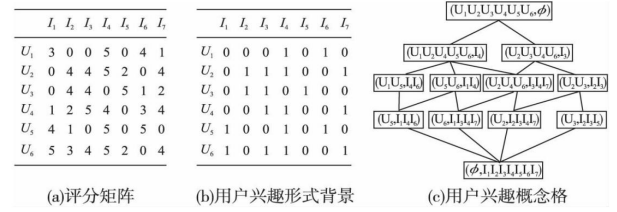


图2 构造用户兴趣概念格实例分析

Fig.2 Example analysis of constructing user interest concept lattice

然后, 查找与目标用户  $u$  最相似的用户  $MN_u$ , 在用户兴趣概念格  $L_c$  中, 采用由底部到顶部的搜索方法来查找  $MN_u$ ,  $MN_u$  的定义如下:

$$MN_u = \{\text{Ext}(Z) \mid \max(|\text{Int}(Z) \cap I_u'|), Z \in L_c\}$$

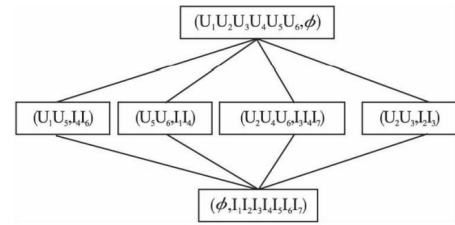


图3 最终的用户兴趣概念格

Fig.3 Final user interest concept lattice

其中,  $Z$  为  $L_c$  中的一个形式概念,  $I_u'$  为目标用户  $u$  感兴趣的项目集合. 对于  $L_c$  中的每个形式概念, 对内涵的所有项目感兴趣的用户都包含在外延中. 为此, 在  $L_c$  中查找内涵与  $I_u'$  重叠最多的形式概念, 该形式概念的外延即为目标用户  $u$  最相似的用户  $MN_u$ . 例如, 如果目标用户  $u$  对项目  $I_2, I_4$  和  $I_7$  感兴趣, 则  $I_u'$  为  $\{I_2, I_4, I_7\}$ , 通过查找  $L_c$  可知, 内涵  $\{I_3, I_4, I_7\}$  与  $I_u'$  重叠最多, 因此,  $MN_u$  为  $\{U_2, U_4, U_6\}$ .

最后, 通过“最近邻”  $N_u$  和最相似用户  $MN_u$ , 划分直接“最近邻”和间接“最近邻”. 目标用户  $u$  的直接“最近邻”  $N_u^d$  的定义如下:

$$N_u^d = \{x \mid x \in N_u \text{ 且 } x \in MN_u\}$$

目标用户  $u$  的间接“最近邻”  $N_u^{\text{id}}$  的定义如下:

$$N_u^{\text{id}} = \{x \mid x \in N_u \text{ 且 } x \in N_u \cap MN_u\}$$

例如, 如果一个目标用户  $u$  的“最近邻”  $N_u$  为  $\{U_1, U_2, U_4, U_5, U_6\}$ , 最相似用户  $MN_u$  为  $\{U_2, U_4, U_6\}$ , 根据上面的定义可知, 直接“最近邻”  $N_u^d$  为  $\{U_2, U_4, U_6\}$ , 间接“最近邻”  $N_u^{\text{id}}$  为  $\{U_1, U_5\}$ .

### 3.2 推荐模块

确定了目标用户的直接“最近邻”和间接“最近邻”后, 就可以计算用户之间的相似度, 以此来推算出目标用户对某个项目的感兴趣程度, 感兴趣程度可以通过预测评分值来表示.

#### 3.2.1 计算相似度

对于直接“最近邻”用户, 由于直接“最近邻”用户的评分在评分预测过程中影响更加明显, 他们与目标用户直接相关. 所以, 直接“最近邻”用户与目标用户

之间的相似度值被设置为 $<1>$ .

对于间接“最近邻”用户,本文提出了一种新的方法来计算间接“最近邻”用户与目标用户之间的相似度.在基于用户的协同过滤推荐方法中,采用式(6)计算用户 $u$ 与用户 $v$ 的相似度.

$$Sim(u, v) = \frac{\max(1, |I_u \cap I_v|) \cdot \sum_{i \in I_u \cap I_v} r_{u,i}}{|I_u| \cdot |I_v| \cdot |I_u \cup I_v|} \quad (6)$$

式中, $I_u$ 为间接“最近邻”用户 $u$ 评分过的项目集, $I_v$ 为目标用户 $v$ 评分过的项目集, $I_u$ 为用户 $u$ 评分过的项目个数.从式(6)可以看出,本文提出的相似度计算方法只需要使用用户的评分项目集,避免了用户之间的共同评分项目的影响.从而,解决数据稀疏问题.

### 3.2.2 预测评分值

计算了相似度后,接下来是对评分值进行预测.首先,计算推荐项目的平均评分.对于推荐项目 $i$ ,采用式(7)计算平均评分.

$$\bar{r}_i = \frac{\sum_{v \in N_u^d \cup N_u^{id}} r_{v,i}}{|N_u^d \cup N_u^{id}|} \quad (7)$$

式中, $N_u^d$ 为用户 $u$ 的直接“最近邻”, $N_u^{id}$ 为用户 $u$ 的间接“最近邻”.

然后,预测推荐项目的评分,使用式(8)预测目标用户 $u$ 对项目 $i$ 的评分 $R_{u,i}$ .式中, $a$ 是评分集合的最高评分.

$$R_{u,i} = \frac{\sum_{v \in N_u^d} r_{v,i} \cdot (a - |r_{v,i} - \bar{r}_i| - 1)^2 + \sum_{v \in N_u^{id}} r_{v,i} \cdot Sim(u, v) \cdot (a - |r_{v,i} - \bar{r}_i| - 1)^2}{\sum_{v \in N_u^d} (a - |r_{v,i} - \bar{r}_i| - 1)^2 + \sum_{v \in N_u^{id}} Sim(u, v) \cdot (a - |r_{v,i} - \bar{r}_i| - 1)^2} \quad (8)$$

## 4 实验结果

### 4.1 数据集

本文数据集为 MOVIELENS 100K, MovieLens 数据集是评价推荐算法有效性最常用的数据集之一.该数据集包含了 943 名用户对 1682 部电影的所有评分,在 1~5 的评分等级范围内有 100000 个匿名评分,每个用户至少对 20 部电影进行了评级.如果评分为 1 意味着用户不喜欢这部电影,评分为 2 意味着用户对这部电影的喜爱程度很低,评分为 3 意味着用户对这部电影持中立态度,评分为 4 意味着用户喜欢这部电影,评分为 5 意味着用户非常喜欢该部电影.原始数据集中可用评级的密度偏大,由于该模型需要在不同高度稀疏的数据集上进行测试,因此从 MovieLens 数据集中随机的抽取 300 个用户对 500 个随机电影的评分,生成了两个样本数据集.然后,将两个样本数据集中的可用评级随机更改为零,以形成两个稀疏度不同的数据集.数据集-1 的可用评级为 3026,稀疏度约为 98%,数据集-2 的可用评级为 1656,稀疏度约为 99%.

### 4.2 评价指标

本文采用平均绝对误差(mean absolute error, MAE)和均分根误差(rooted mean squared error, RMSE)来评价所提方法的准确性.

MAE 表示预测评级与实际评级之间的偏差,可用于度量预测的精度.通常情况下,MAE 越小,预测的精度越高,MAE 的计算使用公式(9).

$$MAE = \frac{\sum_{i=1}^n |r_i - p_i|}{n} \quad (9)$$

其中, $n$ 表示预测的所有评分的个数, $r_i$ 表示项目 $i$ 的实际评分值, $p_i$ 表示项目 $i$ 的预测评分值.

另一个与 MAE 相似的度量是均方根误差(RMSE),该值表示预测评级与实际评级之间的平均误差,它反映了实际值与预测值之间的偏离程度, RMSE 越小,预测的精度越高,使用公式(10)计算.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (r_i - p_i)^2}{n}} \quad (10)$$

### 4.3 结果

为了验证 RRP-UICL 方法的有效性,将 RRP-UICL 方法与传统的基于修正的余弦相似性度量(UCF-ACos),基于皮尔逊相关度量(UCF-PC),基于受约束的皮尔逊相关度量(UCF-CPC),基于 Jaccard 度量(UCF-Jaccard)四种方法分别在数据集-1 和数据集-2 上进行了实验,对五种方法进行了比较,并给出了精度指标.

当推荐项目为 5、10、15、20 时,对于数据集-1 和数据集-2 两种不同稀疏度的数据集,五种方法的 MAE 随推荐项目变化的实验结果分别见图 4 和图 5.

从图 4 和图 5 中可以看出,在 5 种方法中,RRP-UICL 方法在两个数据集中的 MAE 值都小于其余的 4 种方法.随着稀疏度的增加,UCF-ACos 方法和 UCF-PC 方法的 MAE 值明显地在增大,受数据稀疏的影响较大. UCF-CPC 方法和 UCFJaccard 方法的 MAE 值变化不是很明显. RRP-UICL 方法在整个稀疏度实验范围内均取得了较小的 MAE 值,在数据集稀疏度较高的情况下有明显的优势.总结来说,在稀疏场景下,本文方法比传统的协同过滤方法有更好的预测精度.

同样的,在 5、10、15 和 20 这 4 个不同的推荐项目下,图 6 和图 7 分别展示了 5 种方法在数据集-1 和数据集-2 中的 RMSE 值随推荐项目变化的实验结果.

从图 6 和图 7 中可以看出,与 MAE 值相似,RRP-UICL 方法在两个数据集中的 RMSE 值也都小于其余的 4 种方法.当数据集的稀疏度不断增大时,RRP-UICL 方法仍然保持较低的 RMSE 值,而 UCF-ACos 方法和 UCF-PC 方法的 RMSE 值却在不断的变大,从图 7 中可以清楚地看到,UC



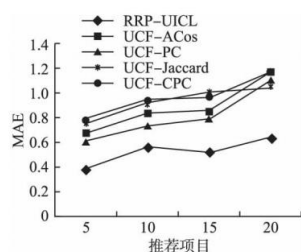


图4 数据集-1 不同方法的 MAE 值比较

Fig. 4 MAE value comparison of the different methods for dataset-1

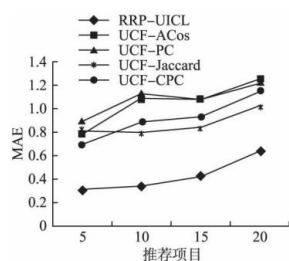


图5 数据集-2 不同方法的 MAE 值比较

Fig. 5 MAE value comparison of the different methods for dataset-2

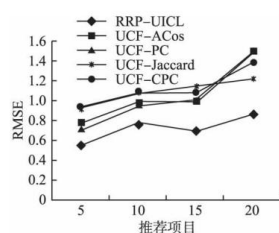


图6 数据集-1 不同方法的 RMSE 值比较

Fig. 6 RMSE value comparison of the different methods for dataset-1

F-ACos 方法和 UCF-PC 方法的 RMSE 值比 RRP-UICL 方法大很多. UCF-CPC 方法和 UCF-Jaccard 方法的 RMSE 值增加幅度不是很大,但也明显大于 RRP-UICL 方法的 RMSE 值. 因此,RRP-UICL 方法比传统的协同过滤方法有更好的预测精度.

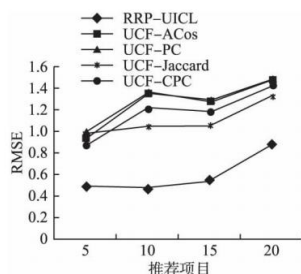


图7 数据集-2 不同方法的 RMSE 值比较

Fig. 7 RMSE value comparison of the different methods for dataset-2

## 5 结束语

本文提出的 RRP-UICL 方法,解决了传统的基于协同过滤推荐算法中存在的用户稀疏的问题. 该方法考虑到目标用户的“最近邻”用户在评分预测过程中的影响

程度不同,所以通过用户兴趣概念格将“最近邻”分为直接“最近邻”和间接“最近邻”两类. 然后采用不同方法计算直接“最近邻”和间接“最近邻”与目标用户的相似度. 最后,通过计算所得的相似度值来预测目标用户的不可见项目评分值. 在真实的电影评分数据集上,验证了 RRP-UICL 方法对稀疏数据提高预测准确度的有效性. 在今后的工作中,我们将研究概念格的构造算法,在本来稀疏用户-项目兴趣数据集上高效地构造概念格,从而提高算法的效率.

## 参考文献

- [1] Lu j, Wu D, Mao M, et al. Recommendersystem application developments: a survey [J].Decision Support Systems,2015,74(6):12-32.
- [2] Duo Lin, Wu Zhao-xia, Xu Sheng-gang.An e-commerce recommender system based on click and purchase data to items and considered of interest shifting of customer [J].China Communications, 2015,12(A2):72-82.
- [3] Song, Y.; Dixon, S.; and Pearce, M.2012. A survey of music recommendation systems and future perspectives. In The International Symposium on Computer Music Modeling and Retrieval. Tencent. 2019. 2018 annual report.
- [4] Daniel V, Javier P, Barreiro Álvaro. Finding and analysing good neighbourhoods to improve collaborative filtering [J]. KnowledgeBased Systems,2018,159(7):193-202.
- [5] Maihami V, Zandi D, Naderi K. Proposing a novel method for improving the performance of collaborative filtering systems regarding the priority of similar users [J]. Physica A: Statistical Mechanics and its Applications,2019,doi:10.1016/j.physa.2019.04.257.
- [6] Luo H, Niu C, Shen R, et al. A collaborative filtering framework based on bothlocal user similarity and global user similarity[J].Machine Learning, 2008, 72(3):231-245.
- [7] Patra B K, Launonen R, Ollikainen V, et al. A new similarity measure using Bhattacharyya coefficient for collaborativefiltering in sparse data [J]. Knowledge-Based Systems, 2015, 82(7):163-177.
- [8] Zhang Jun, Liu Man, Peng Wei-ping. Collaborative filtering recommendation algorithm based on fusion i

- nterest and score [J]. Journal of Chinese Computer Systems, 2017,38(2):357-362.
- [9] Lin Jian-hui, Yan Xuan-hui, Huang Bo. Collaborative filtering recommendation algorithm based on trust users [J]. Computer Systems & Applications, 2017, 26(6):124-130.
- [10] Guo G, Qiu H, Tan Z, et al. Resolving data sparsity by multi-type auxiliary implicit feedback for recommender systems[J]. Knowledge-Based Systems, 2017, 138(12) : 202-207.
- [11] Xu X, Dutta K, Ge C. Do adjective features from user reviews address sparsity and transparency in recommender systems?[J]. Electronic Commerce Research and Applications, 2018, 29(5):113-123. Jiaotong University, 2015, 49(6):77-83.