

# 基于三支聚类的协同过滤推荐方法

康 凯<sup>1,2</sup>, 胡 军<sup>1,2</sup>

(1. 重庆邮电大学 计算智能重庆市重点实验室 重庆 400065;  
2. 重庆邮电大学 计算机科学与技术学院 重庆 400065)

**摘要:** 通过聚类可以缩小用户近邻空间,从而一定程度缓解传统协同过滤推荐算法存在的可扩展性问题,但因部分用户丢失了有效邻居而使得推荐精度不高。为解决该问题,结合三支聚类提出了一种新的协同过滤方法。该方法分为线下聚类和线上推荐两个步骤。对用户先进行聚类,进而将用户划为核心用户和边界用户,并对这两类用户分别应用不同的聚类规则进行聚类;然后在目标用户所属的簇中产生一个预测评分,对属于多个簇的用户,则聚合每个簇的评分得到其预测结果。实验结果表明,该方法与现有基于聚类的协同过滤算法相比,能有效地提高推荐精度。

**关键词:** 协同过滤; 三支决策; 聚类; 推荐系统

中图分类号: TP391

文献标志码: A

文章编号: 1671-6841(2022)03-0022-06

DOI: 10.13705/j.issn.1671-6841.2021237

## Collaborative Filtering Recommendation Method Based on Three-way Clustering

KANG Kai<sup>1,2</sup>, HU Jun<sup>1,2</sup>

(1. *Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*; 2. *College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*)

**Abstract:** Clustering can be used to reduce the users' neighbor space, thereby alleviating the scalability of traditional collaborative filtering algorithms to a certain extent, but the recommendation accuracy is not high due to the loss of effective neighbors of some users. To solve this problem, a new collaborative filtering method combined with three-way clustering was proposed. The proposed method included two steps: offline clustering and online recommendation. The users were divided into core users and boundary users by clustering, and different clustering rules were applied to these two types of users respectively. The prediction score was calculated based on the cluster the target user belonged to. For a user belonging to multiple clusters, the scores of each cluster were aggregated to obtain its prediction score. Experimental results showed that the proposed method could achieve higher accuracy than the current clustering-based collaborative filtering algorithms.

**Key words:** collaborative filtering; three-way decision; clustering; recommendation system

## 0 引言

互联网的发展产生了海量的信息,也导致了信息过载<sup>[1-2]</sup>问题。因此研究人员提出了能自动根据用户的偏好信息将匹配的物品推荐给用户的推荐系统(recommender system, RS)<sup>[3]</sup>。其中协同过滤推荐(collaborative filtering, CF)算法使用用户和物品的交互信息进行推荐,并在许多领域都取得了很好的推荐效果。协同过滤算法可分为基于用户的协同过滤(user-based CF, UCF)算法和基于物品的协同过滤算法<sup>[4]</sup>。传统协同过滤算法搜索近邻时需要遍历整个用户空间,算法效率随着数据规模增长而下降,其可扩展性面临挑战<sup>[5]</sup>。文献[6]首先提出了基于聚类的协同过滤算法,算法使用目标用户所在的聚类簇替代全体用户空间,从而有效缓解可扩展性问题,但是预测评分精度不高。

为此,陆续有学者对聚类算法在推荐中的应用进行了研究,其主要目标是在保持聚类推荐算法的可扩展性优势的同时提高推荐精度。文献[7]提出了一种填充空白评分项的推荐算法,算法在填充后的评分矩阵上使用基于物品聚类的协同过滤算法进行预测,实验证明该算法可以提高预测评分精度。文献[8]通过对不同聚类簇的用户间的相似度添加不同的惩罚措施,并使用新的相似度产生预测评分,实验表明预测精度相比于传统协同过滤算法有所提升。文献[9]认为多次聚类能获得多个聚类结果,选择和目标用户最相似的聚类簇作为近邻搜索空间能提高近邻质量,实验结果表明该算法可以提高推荐精度和在线推荐效率,但聚类成本会增加。文献[10]提出了针对 Top-N 推荐任务的用户模糊聚类方法,实验表明该方法在精确度等指标上优于使用 K-Means 的协同过滤算法。文献[11]将模糊聚类与协同过滤结合,对目标用户在所属的所有簇中分别进行预测,将聚合后的评分作为最终的预测评分,实验表明精度优于传统基于聚类的协同过滤算法。文献[12]融合了用户偏好信息对聚类结果进行优化,实验表明推荐精度有所提高。

经过对基于聚类的推荐算法的研究和分析,现有研究按照聚类算法的不同可以分为硬聚类和软聚类算法<sup>[7-11, 13]</sup>。现有聚类推荐算法都是使用单一的聚类规则进行聚类,其在用户的评分数据较少、信息不充足的情况下存在推荐准确率不高的问题。鉴于聚类的结果对发现的近邻质量有较大影响<sup>[6, 9, 11]</sup>,对聚类环节进行改进就成为提升推荐精度的一种可

行方案。基于此,本文提出了一种基于三支聚类的协同过滤推荐方法,并通过实验验证了该方法的有效性。

## 1 相关知识

### 1.1 基于用户的协同过滤算法

基于用户的协同过滤算法使用近邻用户对物品的意见在一定程度上预测目标用户的意见。常用的预测评分计算方法为

$$p_{u,i} = \bar{r}_u + \frac{\sum_{v \in N(u)} \text{sim}(u,v) \cdot (r_{v,i} - \bar{r}_v)}{\sum_{v \in N(u)} \text{sim}(u,v)},$$

其中: $p_{u,i}$ 表示用户 $u$ 对物品 $i$ 的预测评分; $\bar{r}_u$ 是用户 $u$ 的平均评分; $N(u)$ 是用户 $u$ 的近邻用户(一般选取前 $K$ 个最相似邻居); $r_{v,i}$ 是用户 $v$ 对物品 $i$ 的评分; $\bar{r}_v$ 是用户 $v$ 的平均评分。 $\text{sim}(u,v)$ 为用户 $u$ 和 $v$ 的相似度,其计算方法为

$$\text{sim}(u,v) = \frac{\min\{|\mathcal{J}_{uv}|, th\}}{th} \cdot PC(u,v),$$

其中: $\mathcal{J}_{uv}$ 是同时被用户 $u$ 和 $v$ 评分过的物品集合。考虑到用户评分数量的影响,对评分少于阈值 $th$ 的用户进行惩罚,其值一般由经验值确定,本文实验中取值为50。 $PC(u,v)$ 使用皮尔逊相关系数进行度量,其计算方法为

$$PC(u,v) = \frac{\sum_{i \in \mathcal{J}_{uv}} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in \mathcal{J}_{uv}} (r_{u,i} - \bar{r}_u)^2 \sum_{i \in \mathcal{J}_{uv}} (r_{v,i} - \bar{r}_v)^2}}.$$

### 1.2 三支聚类

在信息不充足的情况下,人们往往会采取不同的策略进行决策,三支决策正是这样的策略方法。而推荐和聚类都可以视为决策的过程,已经有学者将三支决策用于优化推荐成本<sup>[14-15]</sup>。

近年研究表明三支聚类<sup>[16]</sup>在不完整数据上聚类准确度超过了传统聚类。通过将三支决策思想应用到聚类中,可将聚类簇分为三个区域,即核心域( $Co(C^m)$ )、边界域( $Fr(C^m)$ )和外部域( $Tr(C^m)$ )。核心域的用户完全属于该簇,外部域的用户完全不属于该簇,边界域的用户部分属于该簇,且可能同时属于多个簇。对不确定归属的用户采用的决策规则为

$$\begin{cases} Co(C^m) = \{x \in U \mid v(x) \geq \alpha\}, \\ Fr(C^m) = \{x \in U \mid \beta < v(x) < \alpha\}, \\ Tr(C^m) = \{x \in U \mid v(x) \leq \beta\}. \end{cases}$$

$Tr(C^m)$  可以通过对  $Co(C^m) \cup Fr(C^m)$  求补集得到,因此使用  $\{Co(C^m), Fr(C^m)\}$  就能刻画一个簇。

### 1.3 样本密度及距离

文献[17]提出了一种快速寻找到密度峰值点聚类 (clustering by fast search and find of density peaks, DPC) 算法,通过计算样本  $i$  的局部密度值  $\rho_i$  和距离值  $\delta_i$ ,绘制决策图后可以发现潜在的聚类中心。本文经过分析后认为决策图还可以区分聚类簇边缘的离散样本。本文使用文献[18]提出的改进方法计算为

$$\rho_i = \sum_{j \in Neighbor(i)} \exp(-d_{ij}),$$

$$\delta_i = \begin{cases} \min(d_{ij}), & \text{如果 } \exists j: \rho_j > \rho_i, \\ \max(d_{ij}), & \text{其他,} \end{cases}$$

其中:  $\rho_i$  为样本  $i$  的密度值;  $\delta_i$  为样本  $i$  的距离值;  $d_{ij}$  为样本  $i$  与  $j$  之间的欧氏距离。样本点  $i$  与属于  $Neighbor(i)$  的样本距离越近,  $\rho_i$  越大,一般  $Neighbor(i)$  的大小取值为数据集规模的 1.5%。

## 2 基于三支聚类的协同过滤推荐

现有的聚类推荐算法在面对信息丰富程度不同的样本时依然采用单一聚类算法进行划分,因此聚

类结果易受到干扰。同时有些用户的评分过少,直接在原始数据上进行聚类的结果往往不够准确。

对图1所示的模拟数据为例进行聚类,可以看出位于中间区域的样本①、②难以确定归属,而其他样本则可以确定所属的簇。因此若能先对确定归属的样本进行聚类,并对不确定的样本延迟决策,则可以提高聚类结果准确性。

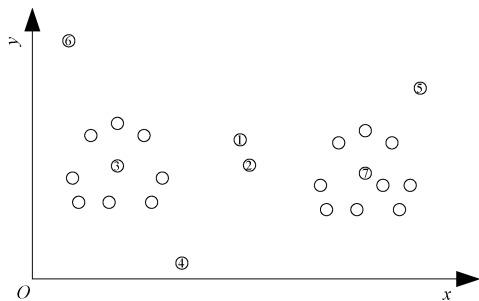


图1 模拟数据

Figure 1 Artificial data

基于上述分析,为提升基于聚类的协同过滤算法的推荐精度,本文提出了基于用户的三支聚类方法,并在此基础上实现了一种协同过滤推荐方法。如图2所示,方法分为线下聚类和线上推荐两部分。线下聚类环节中,使用三支聚类算法对用户进行聚类;线上推荐环节中,在目标用户所在的簇中寻找最近邻,计算目标用户对目标物品的预测评分。

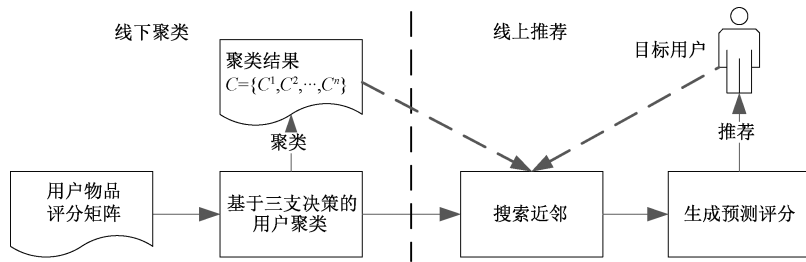


图2 基于三支聚类的协同过滤推荐方法

Figure 2 Collaborative filtering recommendation method based on three-way clustering

### 2.1 三支用户聚类

推荐系统评分数据集的用户评分数据稀疏度高,特别是数据集中不同用户的评分数量相差较多,对评分数目过少的用户进行预测推荐的依据是不充足的。为此,先对用户进行划分,再进行聚类。

**2.1.1 用户划分规则** 1) 将全体用户中评分数量低于某个阈值的用户划分到边界用户  $fringes$ 。

评分数据中每个用户的评分数量各不相同,提供给推荐系统的信息充足程度也不同,因此应首先考虑对信息更充足的用户先聚类,阈值选取为 10。

2) 使用 1.3 节介绍的方法计算每个用户的密度值  $\rho_u$  和距离值  $\delta_u$ ,并寻找到潜在的密度峰值点

$c$ ; 随后将用户划分到  $fringes$  或  $cores$ , 划分规则为

$$\begin{cases} u \in cores, & \rho_u > 0 \text{ 且 } \frac{\delta_u}{\rho_u} < \frac{\delta_c}{\rho_c}, \\ u \in fringes, & \text{其他,} \end{cases}$$

其中:  $u$  为待划分的用户;  $cores$  为核心用户;  $fringes$  为边界用户。

文献[17]指出,密度值  $\rho_u$  越低表示其周围的邻居数量较少,距离值  $\delta_u$  越低则表示其不是密度峰值样本。图3为图1的决策图,从图1中可看出样本自动形成两个簇,位于中间区域的样本则需要确定所属的簇。根据密度值  $\rho_u$  和距离值  $\delta_u$  的物理含义,可判断出图3中区域1的样本更可能会被划分到核

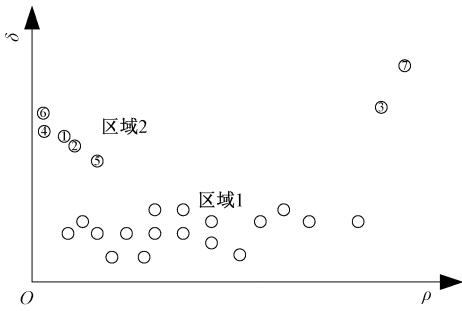


图3 模拟数据上的DPC决策图

Figure 3 Decision graph of DPC on artificial data

心域,而区域2的样本更可能被划分到边界域。

依据2)将样本①、②、④、⑤、⑥划分到边界域中,样本③和⑦作为聚类中心,这与我们的直观印象是符合的,说明了划分规则的有效性。

**2.1.2 三支决策规则** 由于边界用户 *fringes* 中的用户信息不充足,使用传统聚类方法划分可能出现误分类。因此考虑对 *fringes* 用户采用三支决策规则决定一个用户的归属。在三支决策中,用户可能被划分到某一个簇的核心域,也可能被划分到某一个或多个簇的边界域。具体划分规则如下。

1) 计算用户  $u$  的邻居分布在各个簇的比例,该值也用来度量用户对边界域的隶属度,

$$P(u | C^m) = \frac{|(v | v \in Neighbor(u) \cap v \in Co(C^m))|}{|Neighbor(u)|},$$

其中:  $P(u | C^m)$  表示用户  $u$  的邻居中属于  $C^m$  的比例;  $Neighbor(u)$  为目标用户  $u$  的前  $\lambda$  个最近邻居;  $C^m$  为三支用户聚类算法第2步核心域聚类后的第  $m$  个簇;  $Co(C^m)$  为  $C^m$  的核心域。实验中  $\lambda$  取值过大或过小都不利于样本的分配,本实验中取值为10。

2) 结合三支决策,使用  $P(u | C^m)$  值作为划分的参考依据。划分规则为

$$\begin{cases} Co(C^m) = \{u \in U | P(u | C^m) \geq \alpha\}, \\ Fr(C^m) = \{u \in U | \beta < P(u | C^m) < \alpha\}, \end{cases}$$

其中:  $Co(C^m)$  为第  $m$  个聚类簇  $C^m$  的核心域;  $Fr(C^m)$  为  $C^m$  的边界域;  $U$  为全体用户集合;  $\alpha$  和  $\beta$  为基于三支决策的用户聚类算法的一对阈值参数。

针对稀疏评分数据的用户聚类算法,如算法1。

**算法1** 基于用户的三支聚类算法

输入: 训练集  $U$ , 聚类簇数目  $N$ , 三支决策阈值  $\alpha$  和  $\beta$ , 三支决策近邻数目  $\lambda$ 。

输出: 三支聚类结果  $C$ 。

Step1 数据划分 按2.1.1节的划分规则,将训练集  $U$  划分为两个互补子集,即核心用户 *cores* 和边界用户 *fringes* ;

Step2 核心用户聚类 对 *cores* 用户调用

*K*-Means 算法,聚类结果即为聚类簇的核心域;

Step3 边界用户划分 按2.1.2节将 *fringes* 用户分配到已有簇的核心域  $Co(C^m)$  或边界域  $Fr(C^m)$  。

**2.2 预测评分**

在1.1节中介绍了基于用户的协同过滤算法的预测评分计算方法,但三支用户聚类中存在另一个不同之处:  $Fr(C^m)$  中的用户可能同时属于另一个簇。那么在每个簇中都可以产生一个用户  $u$  的预测评分  $p_{u,i}^{C^m}$  和  $p_{u,i}^{C^n}$ 。本文采取的方法是对每个聚类簇计算一个预测评分值,聚合后得到最终预测值,评分聚合示意图如图4所示。

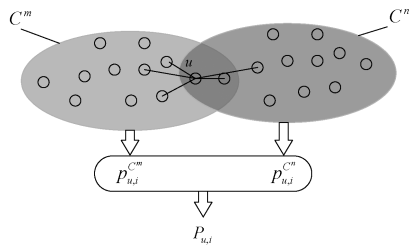


图4 评分聚合示意图

Figure 4 The fusion of prediction scores

具体计算方法为

$$P_{u,i} = (\sum_{c^m \in C(u)} \mu_{u,c^m} \cdot p_{u,i}^{C^m}) / (\sum_{c^m \in C(u)} \mu_{u,c^m}),$$

其中:  $P_{u,i}$  为最终预测评分;  $\mu_{u,c^m}$  为用户  $u$  在聚类簇  $C^m$  中的隶属程度,若用户  $u \in Co(C^m)$  则  $\mu_{u,c^m} = 1$ , 否则  $\mu_{u,c^m} = P(u | C^m)$ ;  $p_{u,i}^{C^m}$  为得到的用户  $u$  在聚类簇  $C^m$  中的预测评分。

**3 实验方法及结果**

**3.1 实验设计**

为验证本文提出的基于用户的三支聚类协同过滤推荐方法的有效性,设计了对比实验,参与对比的实验算法有:基于用户的协同过滤算法(UCF)、基于 *K*-Means 用户聚类的协同过滤推荐算法(UCCF(*K*-Means))、基于模糊用户聚类的协同过滤推荐算法(UCCF(fcm))和本文提出的算法。

实验数据采用 MovieLens-100k 数据集,包含了943个用户对1682部电影的100000条评分数据,评分范围为1~5,分值高低代表喜好程度。采用 *K*-fold 交叉验证法将数据集按4:1比例随机划分为互不相交的训练集和测试集,共产生5组不同的训练集和测试集。

采用的评价指标为平均绝对误差(MAE)和均

方根误差 ( $RMSE$ ), 计算为

$$MAE = \left( \sum_{i=1}^k |p_{u,i} - r_{u,i}| \right) / k,$$

$$RMSE = \sqrt{\left( \sum_{i=1}^k (p_{u,i} - r_{u,i})^2 \right) / k},$$

其中:  $p_{u,i}$  是用户  $u$  对物品  $i$  的预测评分;  $r_{u,i}$  为测试集中的真实评分;  $k$  为测试集评分数量。  $MAE$  与  $RMSE$  值代表的意义相似, 值越低意味着推荐精度越高, 并且  $RMSE$  还能体现方差大小。

### 3.2 实验结果

实验采取网格搜索法确定三支决策阈值。首先固定  $\alpha = 0.6, \beta$  在  $[0.1, 0.4]$  区间按步长 0.1 进行遍历取值, 选取合适的  $\beta$  值; 随后  $\alpha$  在  $[0.5, 0.9]$  区间按 0.1 的步长遍历并选取合适的  $\alpha$  值。其他参数取值为聚类数目  $N = 3$ ; 三支决策近邻数目  $\lambda = 10$ ; 推荐系统近邻数目  $K = 10$ 。综合对比后  $\alpha$  和  $\beta$  分别取值 0.6 和 0.2。

随后验证聚类数目  $N$  对推荐精度的影响。聚类数目  $N$  取值范围为  $[2, 6]$ , 其他参数不变。实验结果见图 5。从整体趋势看, 改变聚类簇数目  $N$  可以使推荐结果更加精准; 但是当  $N$  取值过小或过大时, 聚类结果都不能达到最符合真实分布的情况。从具体对比看,  $UCCF(K\text{-Means})$  和  $UCCF(fcm)$  的  $MAE$  值随着  $N$  增加而降低,  $N = 3$  时  $MAE$  值达到最低, 本文算法在  $N$  小于 4 时取得最小值;  $N$  达到 4 之后三种方法的  $MAE$  值都持续增加。在相同条件下,  $UCCF(K\text{-Means})$  的  $MAE$  值最高, 本文提出算法的  $MAE$  值在三种算法中最低。

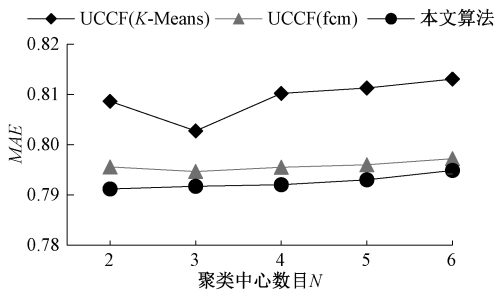


图 5 MAE 随聚类中心数目变化图

Figure 5 Effect of cluster numbers on MAE

接着验证近邻数目  $K$  的影响。实验参数更改为聚类数目  $N = 3$ , 其他参数不变。实验中  $K$  取值区间为  $[5, 20]$ , 步长为 5。实验结果如图 6。从整体趋势看,  $MAE$  值呈先下降后平缓趋势, 说明随着  $K$  的增加, 新增的邻居和目标用户的相似度是逐渐降低的, 对提升推荐精度作用有限。从不同算法的对比看,  $K$  值取 5 时, 各个算法的  $MAE$  值比较接近; 随

着  $K$  值增长到 10,  $UCCF(K\text{-Means})$  的表现最差, 其余三个算法有差距但比较接近, 本文的算法的  $MAE$  值最低; 随着  $K$  值增大,  $UCCF(K\text{-Means})$ 、 $UCF$ 、本文的算法三者之间的差距变大后保持相对稳定, 本文的算法的精度依然最高。

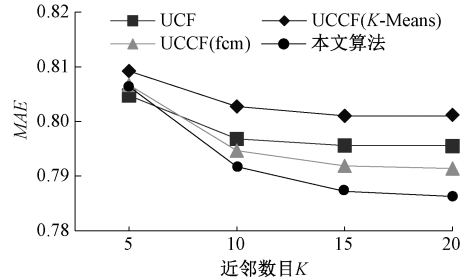


图 6 MAE 随近邻数目变化图

Figure 6 Effect of neighbor numbers on MAE

最后为了验证算法的稳定性, 我们在划分好的 5 组训练集和测试集上进行了实验, 分别对比了  $MAE$  和  $RMSE$  指标, 近邻数目  $K = 15$ 。实验结果如图 7、8。

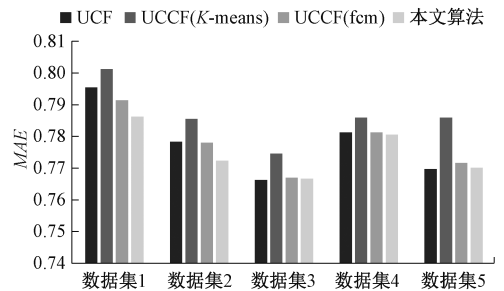


图 7 算法在不同数据集下的 MAE

Figure 7 Comparison of MAE on different datasets

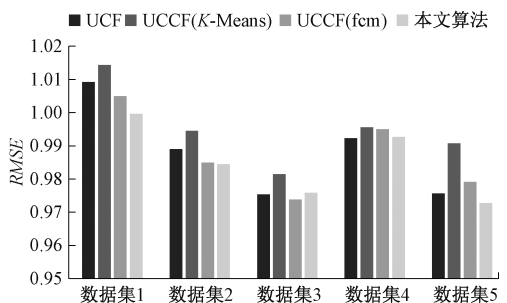


图 8 算法在不同数据集下的 RMSE

Figure 8 Comparison of RMSE on different datasets

从图 7 可以看出在  $MAE$  指标上,  $UCCF(K\text{-Means})$  在所有数据集下的  $MAE$  值都是最高的; 与  $UCCF(fcm)$  相比, 本文算法的  $MAE$  值更低, 在 2 个数据集上有明显下降, 另外 2 个数据集上略有降低; 与  $UCF$  相比, 本文算法的  $MAE$  值都不高于  $UCF$

的。从图8可以看出, *RMSE* 指标仅在数据集5上略高于UCF的,但仍和UCCF(fcm)的相当,其他情况下优于UCF和UCCF(fcm)的值。

## 4 结论

本文将三支决策引入到推荐系统的聚类环节,将用户划分为核心用户和边界用户并分别应用不同的聚类规则,避免信息不充足的用户对聚类造成的干扰。实验结果表明提出的方法可以在不使用额外信息的前提下提高推荐预测精度。本文的研究对象是静态数据,后续可将本文的研究思路进一步应用于动态数据。

## 参考文献:

- [1] 王刚, 蒋军, 王含茹. 社会化推荐研究综述[J]. 计算机科学, 2018, 45(S2): 37-42, 62.  
WANG G, JIANG J, WANG H R. Review of social recommendation[J]. Computer science, 2018, 45(S2): 37-42, 62.
- [2] SOHAIL S S, SIDDIQUI J, ALI R. Classifications of recommender systems: a review[J]. Journal of engineering science and technology review, 2017, 10(4): 132-153.
- [3] RESNICK P, VARIAN H R. Recommender systems[J]. Communications of the ACM, 1997, 40(3): 56-58.
- [4] ADOMAVICIUS G, TUZHILIN A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions[J]. IEEE transactions on knowledge and data engineering, 2005, 17(6): 734-749.
- [5] LINDEN G, SMITH B, YORK J. Amazon.com recommendations: item-to-item collaborative filtering [J]. IEEE internet computing, 2003, 7(1): 76-80.
- [6] SARWAR B M, KARYPIS G, KONSTAN J, et al. Recommender systems for large-scale e-commerce: scalable neighborhood formation using clustering[C]//Proceedings of the 5th International Conference on Computer and Information Technology. New York: ACM Press, 2002: 291-324.
- [7] GONG S J. A collaborative filtering recommendation algorithm based on user clustering and item clustering[J]. Journal of software, 2010, 5(7): 745-752.
- [8] TORRES N, MENDOZA M. Clustering approaches for top-k recommender systems[J]. International journal on artificial intelligence tools, 2019, 28(5): 1950019.
- [9] KUŹELEWSKA U. Multi-clustering used as neighbourhood identification strategy in recommender systems [M]//Advances in Intelligent Systems and Computing. Cham: Springer International Publishing, 2019: 293-302.
- [10] BEZDEK J C, EHRLICH R, FULL W. FCM: The fuzzy C-means clustering algorithm[J]. Computers & geosciences, 1984, 10(2/3): 191-203.
- [11] HONG Y. A framework of three-way cluster analysis[C]//Proceedings of the International Joint Conference on Rough Sets. Berlin: Springer Press, 2017: 300-312.
- [12] KOOHI H, KIANI K. User based collaborative filtering using fuzzy C-means[J]. Measurement, 2016, 91: 134-139.
- [13] 朱威威, 赵岩松, 李艳灵. 一种基于集合划分的鲁棒性自适应模糊聚类分割算法[J]. 信阳师范学院学报(自然科学版), 2019, 32(1): 146-152.  
ZHU W W, ZHAO Y S, LI Y L. A Robust adaptive fuzzy clustering segmentation algorithm based on set partition [J]. Journal of Xinyang normal university (natural science edition), 2019, 32(1): 146-152.
- [14] ZHANG H R, MIN F. Three-way recommender systems based on random forests[J]. Knowledge-based systems, 2016, 91: 275-286.
- [15] 叶晓庆, 刘盾, 梁德翠. 基于协同过滤的三支粒推荐算法研究[J]. 计算机科学, 2018, 45(1): 90-96.  
YE X Q, LIU D, LIANG D C. Three-way granular recommendation algorithm based on collaborative filtering [J]. Computer science, 2018, 45(1): 90-96.
- [16] 李悦, 谢珺, 侯文丽, 等. 融合用户偏好优化聚类的协同过滤推荐算法[J]. 郑州大学学报(理学版), 2020, 52(2): 29-35.  
LI Y, XIE J, HOU W L, et al. Collaborative filtering recommendation algorithm based on optimized clustering with user preference[J]. Journal of Zhengzhou university (natural science edition), 2020, 52(2): 29-35.
- [17] RODRIGUEZ A, LAIO A. Clustering by fast search and find of density peaks[J]. Science, 2014, 344(6191): 1492-1496.
- [18] 谢娟英, 高红超, 谢维信. K近邻优化的密度峰值快速搜索聚类算法[J]. 中国科学: 信息科学, 2016, 46(2): 258-280.  
XIE J Y, GAO H C, XIE W X. K-nearest neighbors optimized clustering algorithm by fast search and finding the density peaks of a dataset[J]. Scientia sinica (informationis), 2016, 46(2): 258-280.