

DOI:10.13232/j.cnki.jnju.2023.05.006

融入用户风险偏好的三支协同过滤推荐模型

黄树添¹, 胡诗琳¹, 卜祥智¹, 李华雄², 刘久兵^{1*}

(1. 汕头大学商学院, 汕头, 515063; 2. 南京大学工程管理学院, 南京, 210023)

摘要: 基于用户的不同风险偏好特征, 提出一种融入用户风险偏好的三支协同过滤推荐模型来提高推荐规则的准确性. 首先, 考虑用户的不同风险偏好对项目评分的影响, 基于用户-项目评分矩阵定义用户关于项目的偏好概率测度, 建立用户-项目偏好概率模型, 从理论上证明了该模型是现有模型的推广和拓展. 其次, 利用决策粗糙集, 推导出用户在不同风险偏好下的三支推荐模型阈值表达. 然后, 以上述工作为基础, 将推荐准确性和推荐成本作为优化目标, 设计基于粒子群优化算法的用户偏好概率模型参数确定方法. 最后, 在 MovieLens 数据集上的实验验证了提出模型的有效性.

关键词: 协同过滤, 风险偏好, 三支推荐, 偏好概率模型

中图分类号: TP301

文献标志码: A

Three-way collaborative filtering recommendation model integrating user risk preferences

Huang Shutian¹, Hu Shilin¹, Bu Xiangzhi¹, Li Huaxiong², Liu Jiubing^{1*}

(1. School of Business, Shantou University, Shantou, 515063, China;

2. School of Management and Engineering, Nanjing University, Nanjing, 210023, China)

Abstract: Based on the characteristics of different risk preferences of users, a three-way collaborative filtering recommendation model integrating user risk preferences is proposed to improve the accuracy of recommendation rules. First, considering the impact of different risk preferences of users on item ratings, a user's preference probability measure for items is defined based on the user-item rating matrix, and a user-item preference probability model is established, which is theoretically proved to be a generalization and extension of the existing model. Second, by using decision-theoretic rough set, mathematical expressions for the threshold of three-way recommendation under different risk preferences of users are deduced. Moreover, based on the aforementioned work, taken recommendation accuracy and recommendation cost as optimization objectives, a method for determining parameters of the user's preference probability model is designed based on particle swarm optimization algorithm. Finally, experiments on the MovieLens dataset verify the effectiveness of the proposed model.

Key words: collaborative filtering, risk preference, three-way recommendation, preference probability model

互联网时代数据爆炸, 信息过载, 庞大的信息体量在给用户带来巨大内容冲击的同时也消弭了用户的注意力与判断力, 因此, 过滤无用信息并筛

选出用户感兴趣的内容成为一个重要的研究课题. 推荐系统因其有效的过滤机制为该类问题提供了新的解决范式, 它通过挖掘用户的行为与偏

基金项目: 国家自然科学基金(62106135, 62176116), 广东省自然科学基金(2023A1515011390, 2022A1515011571, 2023A1515011029), 广东省基础与应用基础研究青年项目(2020A1515110434), 广东省哲学社会科学“十三五”规划青年项目(GD20YGL13), 广东烟草汕头市有限责任公司项目(2023440500260003)

收稿日期: 2023-06-19

* 通讯联系人, E-mail: jiubingliu@163.com

好来捕捉用户感兴趣但未曾留意的商品信息. 在推荐系统研究领域, 协同过滤是应用最广泛的算法之一^[1-2], 它利用用户对项目的历史评分数据来预测用户对未评分项目的喜好程度, 并根据预测评分对用户进行推荐.

传统的推荐系统通常选择推荐或不推荐, 但若信息不充分或获取信息代价较高, 直接推荐或不推荐可能会带来较大的推荐成本. 因此, Zhang et al^[3-4]首次将三支决策引入推荐系统, 在传统的推荐或不推荐的基础上, 引入第三种策略——延迟推荐, 提出三支推荐的概念.

作为三支推荐的理论基础, 三支决策 (Three-Way Decision) 是一种符合人类认知习惯的决策理论, 自提出以来便获得学者的广泛关注^[5-11], 已在多个领域得到应用, 如医疗诊断^[12]、投资管理^[13]、人脸识别^[14]和机器学习^[15]等. 在推荐系统领域, 叶晓庆等^[16]在三支推荐的基础上, 利用粒计算来弥补传统协同过滤算法评分信息单一的缺陷. Ma et al^[17]提出一种结合全局和局部信息的三支推荐方法, 在预测评分时考虑了全局和局部的信息. Liu and Ye^[18]将三支决策与矩阵分解相结合, 提出多粒度信息下的动态三支推荐算法, 根据不同粒度来动态地调整推荐列表. 秦琴和张恒汝^[19]提出一种基于信任传递机制的三支推荐模型, 将信任约束引入矩阵分解, 可以更好地预测评分. Ye and Liu^[20]提出一种基于协同主题回归的可解释性序贯三支推荐策略, 实现可解释的多层次推荐. Zhang et al^[21]提出三支朴素贝叶斯协同过滤推荐模型, 结合朴素贝叶斯与协同过滤算法来处理用户评分信息的稀疏性. Ye and Liu^[22]提出一种基于成本敏感、时空特征和多粒度决策的三支推荐方法. Ye et al^[23]基于协同深度学习, 提出一种多粒度序贯三支推荐方法.

通常, 在三支推荐过程中, 需要考虑推荐规则的划分阈值与用户对项目的偏好概率, 二者均会对推荐结果产生影响, 而现有文献较少对后者进行深入研究. 在计算偏好概率时, 一般的做法是在本质上将用户的风险态度考虑为风险中立型, 然而, 不同用户的风险态度可能是多样的^[24], 将所有用户的风险态度都划分为同一种会导致偏好概率测度失真, 降低推荐质量.

因此, 本文在已有研究的基础上, 提出一种融入用户风险偏好的三支协同过滤推荐模型. 首先, 基于用户已有的评分信息, 利用协同过滤算法预测用户对未评分项目的评分. 其次, 考虑用户的不同风险态度, 结合效用理论, 定义用户对项目的偏好概率测度, 构建用户-项目偏好概率模型, 并通过理论证明了该模型是现有模型的推广. 随后, 利用偏好概率模型与决策粗糙集, 导出三支推荐模型的阈值表达, 进而设计基于粒子群优化算法的用户偏好概率模型参数确定方法. 最后, 利用 MovieLens 数据集中用户对电影的评分数据对模型进行验证. 本文从用户的风险态度的视角出发, 提供了一个新的具有可拓展性的测度用户偏好概率的方法, 使三支推荐的决策规则划分更加准确, 进一步提高了推荐质量.

1 预备知识

1.1 经典协同过滤算法 协同过滤算法是目前应用最广泛的推荐算法之一, 主要包括基于用户的协同过滤^[25]和基于项目的协同过滤^[26], 后者的性能和推荐质量在很多情形下优于前者^[26]. 因此, 本文主要讨论基于项目的协同过滤. 对于目标用户 u 和候选项目 i , 首先计算用户 u 已评分的项目与项目 i 之间的相似度, 选出与项目 i 最相似的 k 个近邻项目, 利用近邻项目的信息预测用户 u 对项目 i 的可能评分. 主要步骤如下.

步骤 1. 计算项目之间的相似性.

表 1 为用户-项目评分矩阵, 其中 $U = \{u_1, u_2, \dots, u_m\}$ 表示用户集合, $I = \{i_1, i_2, \dots, i_n\}$ 表示项目集合, $R = U \times I$ 表示用户 u 对项目 i 的评分 R_{ui} 构成的集合. 根据该评分矩阵可以计算项目之间的 Pearson 相关系数, 其取值范围为 $[-1, 1]$. 该系数

表 1 用户-项目评分矩阵

Table 1 User-Item rating matrix

用户	项目			
	i_1	i_2	\dots	i_n
u_1	R_{11}	R_{12}	\dots	R_{1n}
u_2	R_{21}	R_{22}	\dots	R_{2n}
\vdots	\vdots	\vdots	\dots	\vdots
u_m	R_{m1}	R_{m2}	\dots	R_{mn}

越接近1说明两个项目越相似,越接近0说明两个项目之间的差异性越大.具体计算如下:

$$\text{sim}(i, j) = \frac{\sum_{u \in K} (R_{ui} - \bar{R}_i)(R_{uj} - \bar{R}_j)}{\sqrt{\sum_{u \in K} (R_{ui} - \bar{R}_i)^2} \sqrt{\sum_{u \in K} (R_{uj} - \bar{R}_j)^2}} \quad (1)$$

其中, $\text{sim}(i, j)$ 表示项目 i 与 j 之间的 Pearson 系数,用来测度 i 和 j 之间的相似程度; K 表示同时评价项目 i 和 j 的用户集合; R_{ui} 表示用户 u 对项目 i 的评分; \bar{R}_i 表示项目 i 已有评分的平均数.

步骤2. 预测用户对目标项目的评分.

选择用户 u 已评分项目中与目标项目 i 最相似的 k 个项目,根据这 k 个近邻项目与项目 i 之间的 Pearson 系数以及相关评分数据,预测用户 u 对未知评分项目 i 的评分 R_{ui}^* . 具体计算如下:

$$R_{ui}^* = \bar{R}_i + \frac{\sum_{j=1}^k \text{sim}(i, j) \cdot (R_{uj} - \bar{R}_j)}{\sum_{j=1}^k \text{sim}(i, j)} \quad (2)$$

其中, $\text{sim}(i, j)$ 表示目标项目 i 与近邻项目 j 之间的相似度, R_{uj} 表示用户 u 对项目 j 的真实评分, \bar{R}_i 和 \bar{R}_j 分别表示项目 i 和 j 已知评分的平均数.

1.2 三支决策 传统协同过滤推荐是一种二支推荐,只有推荐与不推荐两种状态.然而,当信息不充分或获取信息代价较高时,很难即时作出推荐或不推荐的判定,若此时强行将项目推荐给用户,可能会产生较大的误推荐成本.基于此,本文参考三支决策的延迟策略,将传统的二支决策推荐拓展为三支决策推荐,减少在信息不充分的时候因错误决策造成的严重损失.

三支决策来源于决策粗糙集理论^[27-28],它为决策粗糙集正域、负域、边界域给出了一种合理的语义解释.不同的域对应不同的决策规则,正域对应接受规则,负域对应拒绝规则,边界域对应不承诺或延迟决策规则.若在信息不充分的情况下仍然只考虑接受或拒绝两种选择,类似接受或拒绝一个假设,就可能会作出弃真或择假的错误决策.换言之,此时贸然作出接受或拒绝的判定,将产生错误接受或错误拒绝的成本,这两种决策成本可能高于不作任何判定的成本.在这种情形下,边界域提供的延迟决策选项能更好地体现定量决策的容错性.

根据三支决策理论^[10,29], $\Omega = \{T, \neg T\}$ 表示对象属于 T 和不属于 T 的状态集合, $\Delta = \{a_P, a_B, a_N\}$ 表示对于以上两种状态,分别采取接受决策、延迟决策或拒绝决策三种行动.在不同状态下采取三种行动的损失各不相同,表2给出了其决策代价损失.当对象处于状态 T 时,采取行动 a_P, a_B 和 a_N 分别产生代价损失 $\lambda_{PP}, \lambda_{BP}$ 和 λ_{NP} ; 当对象处于状态 $\neg T$ 时,采取行动 a_P, a_B 和 a_N 分别产生代价损失 $\lambda_{PN}, \lambda_{BN}$ 和 λ_{NN} ^[10].

表2 代价损失矩阵

Table 2 Cost loss matrix

行动	状态	
	$T(P)$	$\neg T(N)$
a_P	λ_{PP}	λ_{PN}
a_B	λ_{BP}	λ_{BN}
a_N	λ_{NP}	λ_{NN}

对于对象 $[x]$,采取 a_P, a_B, a_N 三种行动产生的期望损失分别为:

$$R(a_P|[x]) = \lambda_{PP} \cdot Pr(T|[x]) + \lambda_{PN} \cdot Pr(\neg T|[x])$$

$$R(a_B|[x]) = \lambda_{BP} \cdot Pr(T|[x]) + \lambda_{BN} \cdot Pr(\neg T|[x])$$

$$R(a_N|[x]) = \lambda_{NP} \cdot Pr(T|[x]) + \lambda_{NN} \cdot Pr(\neg T|[x])$$

已知 $Pr(T|[x]) + Pr(\neg T|[x]) = 1$,由贝叶斯最小风险决策理论可知决策规则如下:

(P) 若 $R(a_P|[x]) \leq R(a_B|[x])$ 且 $R(a_P|[x]) \leq R(a_N|[x])$,则 $x \in POS(T)$;

(B) 若 $R(a_B|[x]) \leq R(a_P|[x])$ 且 $R(a_B|[x]) \leq R(a_N|[x])$,则 $x \in BND(T)$;

(N) 若 $R(a_N|[x]) \leq R(a_P|[x])$ 且 $R(a_N|[x]) \leq R(a_B|[x])$,则 $x \in NEG(T)$.

根据文献^[10,29-31],规则(P)-(N)为三支决策,三个规则对应的域分别为正域、边界域和负域.考虑一种合理情形,即满足 $\lambda_{PP} \leq \lambda_{BP} < \lambda_{NP}$ 且 $\lambda_{NN} \leq \lambda_{BN} < \lambda_{PN}$,假设边界域存在,则需满足 $(\lambda_{BP} - \lambda_{PP})(\lambda_{BN} - \lambda_{NN}) < (\lambda_{PN} - \lambda_{BN})(\lambda_{NP} - \lambda_{BP})$.此时,上述(P)-(N)规则可简化为(P1)-(N1).

(P1) 若 $Pr(T|[x]) \geq \alpha$,则 $x \in POS(T)$;

(B1) 若 $\beta < Pr(T|[x]) < \alpha$,则 $x \in BND(T)$;

(N1)若 $Pr(T[x]) \leq \beta$, 则 $x \in NEG(T)$,

其中,

$$\alpha = \frac{\lambda_{PN} - \lambda_{BN}}{(\lambda_{PN} - \lambda_{BN}) + (\lambda_{BP} - \lambda_{PP})}$$

$$\beta = \frac{\lambda_{BN} - \lambda_{NN}}{(\lambda_{BN} - \lambda_{NN}) + (\lambda_{NP} - \lambda_{BP})}$$

从(P1)-(N1)可知,比较条件概率与阈值 α, β 的大小可以确定每个对象 $[x]$ 的决策行动.

2 融入用户风险偏好的三支协同过滤推荐

三支协同过滤推荐在传统协同过滤仅有推荐和不推荐两种策略的基础上,增加了第三种策略——延迟推荐,这有利于降低推荐成本.在三支协同过滤推荐中需要确定用户关于项目的偏好概率,该概率可以利用用户对项目的预测评分来进行测度^[16-23],但不同用户的评分标准各不相同.例如,有的用户倾向给项目打更高的分数,整体评分偏高;有的则倾向较低的分数,整体评分偏低.如果仅仅考虑预测评分,则用户评分标准的不同带来的评分偏差可能会降低偏好概率的准确度.文献[16, 18, 21]采用最小最大值归一化方法来测度偏好概率,消除评分偏差,但值得注意的是,该方法假定所有用户的风险偏好均为风险中立型.事实上,用户对风险的态度可分为风险中立型、厌恶型和偏好型^[32],不同风险偏好的用户评分的偏差不尽相同,仅把用户当作中立型不足以消除评分偏差.因此,在构建三支协同过滤推荐模型时,需要考虑不同用户的风险偏好.为此,本文提出一种新模型,给定用户 u 与候选推荐项目 i ,构建用户-项目对 (u, i) ,利用协同过滤算法预测用户对项目的评分 R_{ui}^* .根据预测评分,基于用户-项目偏好概率模型得到用户 u 对项目 i 的偏好概率 $Prob_{ui}$;根据决策成本矩阵得出推荐规则的阈值对 α 和 β ;最后,利用用户偏好概率与阈值生成三支推荐规则,如表 3 所示.首先,构建用户-项目偏好概率模型,继而根据决策粗糙集与偏好概率导出三支推荐阈值对 α 和 β 的数学表达,最后提出确定用户偏好概率模型具体参数的方法.

2.1 用户-项目偏好概率模型构建 在微观经济

表 3 三支推荐规则

Table 3 Rules of three-way recommendation

划分条件	决策区域	决策规则
$Prob_{ui} \geq \alpha$	$POS(P)$	推荐(R)
$\beta < Prob_{ui} < \alpha$	$BND(P)$	延迟推荐(B)
$Prob_{ui} \leq \beta$	$NEG(P)$	不推荐(N)

学中,效用是指消费者从消费某种物品中得到的满足程度;在决策领域,效用是决策者对于决策后果或决策损益值的一种感受、反应或倾向,是决策者的价值观和偏好在决策活动中的综合反映^[28].用户在决定是否购买某种商品或服务时,既是消费者,也是决策者.作为决策者,用户的风险偏好会影响其对效用的估计,进而影响消费行为.风险中立型、厌恶型或偏好型^[24, 33]的用户对应不同的效用函数,同一类型的用户由于具体偏好的差异,效用函数也不尽相同.

根据上述分析,用户对项目评分的高低反映项目对用户的效用大小及用户对项目的喜好程度.一般地,评分越高说明项目对于用户的效用越高,用户喜欢该项目的概率越大,但是,用户对项目的喜好程度不是影响用户对项目评分的唯一因素,用户自身的风险偏好同样会影响其对项目的评分.例如,同一效用水平下,和风险偏好型用户相比,风险厌恶型用户的评分会偏低,而风险中立型用户的评分介于二者之间,如图 1 所示.因此,考虑用户的不同风险偏好,基于表 1,本文采用 L-A 拟合法^[24, 34]来刻画不同风险偏好下用户关于项目的效用函数.具体地,假设:

$$Utility_u(R_{ui}) = q(R_{ui} + d)^\rho \quad (3)$$

其中, R_{ui} 表示用户 u 对项目 i 的评分, q 为尺度参数, d 为位置参数, ρ 为用户 u 的风险偏好系数, $Utility_u(R_{ui}) \in [0, 1]$.

如图 1 所示,当 $0 < \rho < 1$ 时,用户 u 为风险厌恶型,其效用曲线是上凸的;当 $\rho = 1$ 时,用户 u 为风险中立型,其效用曲线是直线;当 $\rho > 1$ 时,用户 u 为风险偏好型,其效用曲线是上凹的. R_u^1 和 R_u^0 为用户 u 的效用阈值,即当用户 u 对项目的评分不小于 R_u^1 时,用户认为该项目对其的效用为 1;当用户 u 对项目的评分不大于 R_u^0 时,用户认为该项目对其的效用为 0;当用户 u 对项目的评分介于

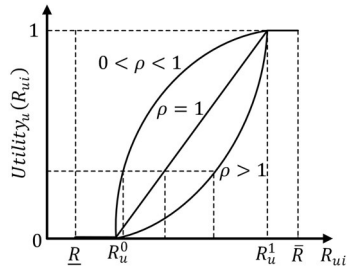


图1 L-A拟合法下的用户-项目效用函数

Fig. 1 L - A simulation method for User - Item utility function

R_u^0 和 R_u^1 之间时,该项目对用户的效用可由式(3)表示.并且,设置推荐系统的评分范围为 $[\underline{R}, \bar{R}]$,例如,淘宝平台的评分范围通常为1~5,则 $\underline{R} \leq R_u^0 < R_u^1 \leq \bar{R}$.

预测评分是判断用户对候选推荐项目偏好程度的依据,通常预测评分越高,说明该项目对用户的效用越大,用户偏好该项目的概率越高.受文献[35]的启发,下面给出用户-项目效用偏好概率模型的定义.

定义 设在表1所示的评分系统中,用户对项目的评分范围为 $[\underline{R}, \bar{R}]$, R_{ui}^* 为用户 u 关于项目 i 的预测评分,则用户 u 关于项目 i 的偏好概率 $Prob_{ui}$ 定义为:

$$Prob_{ui} = \begin{cases} 1, & R_u^1 \leq R_{ui}^* \leq \bar{R} \\ q(R_{ui}^* + d)^\rho, & R_u^0 < R_{ui}^* < R_u^1 \\ 0, & \underline{R} \leq R_{ui}^* \leq R_u^0 \end{cases} \quad (4)$$

其中, R_u^1 和 R_u^0 为用户 u 的效用阈值, q 为尺度参数, d 为位置参数, ρ 为用户 u 的风险偏好系数.

图2为式(4)对应的图像.当 $0 < \rho < 1$ 时,用户 u 为风险厌恶型,对于被推荐到不喜欢的项目的容忍度较低,偏好概率相同时其评分较低;当 $\rho > 1$ 时,用户 u 为风险偏好型,对于被推荐到不

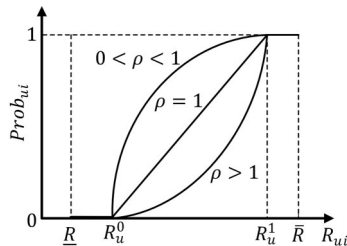


图2 用户-项目效用偏好概率模型

Fig. 1 User-Item utility preference probability model

喜欢的项目的容忍度较高,相同的偏好概率下其评分较高;当 $\rho = 1$ 时,用户 u 为风险中立型,在偏好概率相同时,其评分介于上述两者之间.

式(4)中的参数 q, d 和 ρ 可用三个效用点确定,分别为 $(R_u^1, 1)$, $(R_u^0, 0)$ 和 (R_u^m, m) , 其中, $R_u^0 < R_u^m < R_u^1$ 且 $0 < m < 1$. 则三个参数的解析式为:

$$\begin{cases} \rho = \frac{-\ln(m)}{\ln(R_u^1 - R_u^0) - \ln(R_u^m - R_u^0)} \\ d = -R_u^0 \\ q = \frac{1}{(R_u^1 - R_u^0)^\rho} \end{cases} \quad (5)$$

下面用具体的例子来说明前面的定义.

例 在某电影网站中,用户对电影的评分系统为五分制,最低分为1,最高分为5,评分步长为0.5. 假设用户 u_1 的三个效用点分别为 $(1.5, 0)$, $(2, 0.5)$ 和 $(5, 1)$, 用户 u_2 的三个效用点分别为 $(1, 0)$, $(3, 0.5)$ 和 $(5, 1)$, 用户 u_3 的三个效用点分别为 $(1, 0)$, $(3.5, 0.5)$ 和 $(4.5, 1)$. 基于式(4)和式(5),可以分别得到用户 u_1, u_2 和 u_3 的偏好概率 $Prob_{u_1i}, Prob_{u_2i}$ 和 $Prob_{u_3i}$. 具体如下:

$$Prob_{u_1i} = \begin{cases} 1, & R_{u_1i}^* = 5 \\ 0.64(R_{u_1i}^* - 1.5)^{0.356}, & 1.5 < R_{u_1i}^* < 5 \\ 0, & 1 \leq R_{u_1i}^* \leq 1.5 \end{cases}$$

$$Prob_{u_2i} = \begin{cases} 1, & R_{u_2i}^* = 5 \\ 0.25(R_{u_2i}^* - 1), & 1 < R_{u_2i}^* < 5 \\ 0, & R_{u_2i}^* = 1 \end{cases}$$

$$Prob_{u_3i} = \begin{cases} 1, & R_{u_3i}^* = 5 \\ 0.076(R_{u_3i}^* - 1)^{2.06}, & 1.5 < R_{u_3i}^* < 5 \\ 0, & 1 \leq R_{u_3i}^* \leq 1.5 \end{cases}$$

用户 u_1 的风险系数 $\rho_1 = 0.356 < 1$, 风险厌恶型;用户 u_2 的风险系数 $\rho_2 = 1$, 风险中立型;用户 u_3 的风险系数 $\rho_3 = 2.06 > 1$, 风险偏好型.

根据定义,可得如下命题.

命题 设 R_u^{\max} 和 R_u^{\min} 分别为用户 u 已知评分的最大和最小值,且令 $R_u^1 = R_u^{\max}$, $R_u^0 = R_u^{\min}$. 当用户 u 为风险中立型(即 $\rho = 1$)时,下式恒成立:

$$Prob_{ui} = \begin{cases} 1, & R_u^{\max} \leq R_{ui}^* \leq \bar{R} \\ \frac{R_{ui}^* - R_u^{\min}}{R_u^{\max} - R_u^{\min}}, & R_u^{\min} < R_{ui}^* < R_u^{\max} \\ 0, & \underline{R} \leq R_{ui}^* \leq R_u^{\min} \end{cases} \quad (6)$$

证明 当 $\rho = 1$, 则 $R_u^1 = R_u^{\max}$ 和 $R_u^0 = R_u^{\min}$ 时, 根据式(5), 可以解得:

$$\begin{cases} \rho = 1 \\ d = -R_u^{\min} \\ q = \frac{1}{R_u^{\max} - R_u^{\min}} \end{cases}$$

将 ρ, d 和 q 代入式(4), 即可得式(6). 因此, 命题成立.

从命题可知, 式(6)是最小最大值归一化模型, 换句话说, 该模型是本文提出的用户-项目偏好概率模型的一种特例. 因此, 本文提出的模型是文献[16, 18, 21]中最小最大值归一化模型的拓展, 更具有-般性.

2.2 融入用户风险偏好的三支推荐阈值推导

如表3所示, 三支推荐中给定用户 u 和项目 i , 最优推荐结果取决于用户对项目的偏好概率以及阈值对 α 和 β . 利用构建的用户-项目偏好概率模型, 可计算出用户关于项目的偏好概率. 下面根据决策粗糙集推导出阈值对, 获得三支推荐规则.

表4给出喜欢和不喜欢两种状态下采取三种推荐策略的成本, 其中 $\lambda_{RP}, \lambda_{BP}$ 和 λ_{NP} 分别表示当用户喜欢(P)某项目时, 将该项目推荐(R)、延迟推荐(B)及不推荐(N)给用户时产生的成本; $\lambda_{RN}, \lambda_{BN}$ 和 λ_{NN} 分别表示用户不喜欢(N)时, 将该项目推荐(R)、延迟推荐(B)及不推荐(N)给用户时产生的成本. 其中, λ_{RP} 和 λ_{NN} 表示正确推荐的成本; λ_{BP} 和 λ_{BN} 表示延迟推荐产生的学习成本, 即推荐系统暂时无法确定用户偏好, 需要进一步学习相关信息时产生的成本; λ_{NP} 和 λ_{RN} 表示错误推荐的成本, 其中, λ_{NP} 表示系统拒绝给用户推荐其喜欢的项目时产生的成本, λ_{RN} 表示系统将用户不喜欢的项目推荐给该用户时产生的成本. 通常, 延迟推荐的成本远小于错误推荐的成本, 正确推荐的成本小于延迟推荐的成本, 即表4的成本满足:

表4 三支推荐成本的矩阵

Table 4 The matrix of three-way recommendation cost

决策规则	用户对项目的偏好	
	喜欢(P)	不喜欢(N)
推荐(R)	λ_{RP}	λ_{RN}
延迟推荐(B)	λ_{BP}	λ_{BN}
不推荐(N)	λ_{NP}	λ_{NN}

$$\lambda_{RP} \leq \lambda_{BP} < \lambda_{NP}, \lambda_{NN} \leq \lambda_{BN} < \lambda_{RN} \quad (7)$$

基于用户对项目的偏好概率 $Prob_{ui}$, 决策规则(详见表3)和成本矩阵(详见表4), 三种推荐策略的成本为:

$$C_R = \lambda_{RP} Prob_{ui} + \lambda_{RN} (1 - Prob_{ui})$$

$$C_B = \lambda_{BP} Prob_{ui} + \lambda_{BN} (1 - Prob_{ui})$$

$$C_N = \lambda_{NP} Prob_{ui} + \lambda_{NN} (1 - Prob_{ui})$$

其中, C_R, C_B 和 C_N 分别表示采取推荐、延迟推荐和不推荐策略时产生的决策成本.

根据贝叶斯决策理论^[10], 可得如下规则:

(R) 当 $C_R \leq C_N$ 且 $C_R \leq C_B$ 时, 则 $(u, i) \in POS(P)$;

(B) 当 $C_B \leq C_R$ 且 $C_B \leq C_N$ 时, 则 $(u, i) \in BND(P)$;

(N) 当 $C_N \leq C_R$ 且 $C_N \leq C_B$ 时, 则 $(u, i) \in NEG(P)$.

基于式(7), 三支推荐规则(R)-(N)可进一步简化为规则(R1)-(N1):

(R1) 当 $Prob_{ui} \geq \alpha$ 且 $Prob_{ui} \geq \gamma$ 时, $(u, i) \in POS(P)$;

(B1) 当 $Prob_{ui} \leq \alpha$ 且 $Prob_{ui} \geq \beta$ 时, $(u, i) \in BND(P)$;

(N1) 当 $Prob_{ui} \leq \beta$ 且 $Prob_{ui} < \gamma$ 时, $(u, i) \in NEG(P)$.

其中,

$$\alpha = \frac{\lambda_{RN} - \lambda_{BN}}{(\lambda_{RN} - \lambda_{BN}) + (\lambda_{BP} - \lambda_{RP})}$$

$$\beta = \frac{\lambda_{BN} - \lambda_{NN}}{(\lambda_{BN} - \lambda_{NN}) + (\lambda_{NP} - \lambda_{BP})}$$

$$\gamma = \frac{\lambda_{RN} - \lambda_{NN}}{(\lambda_{RN} - \lambda_{NN}) + (\lambda_{NP} - \lambda_{RP})}$$

假设:

$$(\lambda_{RN} - \lambda_{BN})(\lambda_{NP} - \lambda_{BP}) > (\lambda_{BP} - \lambda_{RP})(\lambda_{BN} - \lambda_{NN})$$

则三支推荐规则(R1)-(N1)可简化为规则:

(R2) 当 $Prob_{ui} \geq \alpha$ 时, $(u, i) \in POS(P)$;

(B2) 当 $\beta < Prob_{ui} < \alpha$, $(u, i) \in BND(P)$;

(N2) 当 $Prob_{ui} \leq \beta$ 时, $(u, i) \in NEG(P)$.

特别地, 当

$$(\lambda_{RN} - \lambda_{BN})(\lambda_{NP} - \lambda_{BP}) \leq (\lambda_{BP} - \lambda_{RP})(\lambda_{BN} - \lambda_{NN})$$

时, 三支协同过滤推荐算法退化为二支协同过滤推荐算法, 此时三支推荐规则(R1)-(N1)简化为:

(R3) 当 $Prob_{ui} \geq \gamma$ 时, $(u, i) \in POS(P)$;

(N3) 当 $Prob_{ui} \leq \gamma$ 时, $(u, i) \in NEG(P)$.

下面, 以2.1和2.2的工作为基础, 提出确定用户-项目偏好概率模型参数的方法.

2.3 基于粒子群优化的偏好概率模型参数确定 基于表1, 可利用式(4)构建用户的偏好概率模型, 但特定评分下用户的偏好概率是未知的, 难

以使用拟合方法确定用户的偏好概率模型参数. 对于用户-项目偏好概率模型,其测度越精确,三支推荐的准确性会越高,推荐成本也越低;反之,当推荐准确性越高、推荐成本越低时,偏好概率模型的测度越精确. 因此,可以将这两个指标作为目标,从优化的角度出发,使用粒子群优化算法来确定用户偏好概率模型的参数,以最大化推荐准确性和最小化推荐成本,使用户-项目偏好概率模型的测度尽可能准确.

由定义可知,用户的偏好概率模型由三个参数 q , d 和 ρ 决定. 由式(5),令 $R_u^m = R_u^{0.5}$, $m = 0.5$,则三个参数的表达式为:

$$\begin{cases} \rho = \frac{\ln(2)}{\ln(R_u^1 - R_u^0) - \ln(R_u^{0.5} - R_u^0)} \\ d = -R_u^0 \\ q = \frac{1}{(R_u^1 - R_u^0)^\rho} \end{cases} \quad (8)$$

如式(8)所示,参数 d , ρ 和 q 可由 $(R_u^1, 1)$, $(R_u^0, 0)$ 和 $(R_u^{0.5}, 0.5)$ 这三个效用点对应的用户评分来确定. 因此,确定用户的偏好概率的问题可以转化为确定用户对应的最优评分组合 R_u^1, R_u^0 和 $R_u^{0.5}$,使模型能够更好地预测用户对项目的偏好概率,优化三支推荐的准确性,降低推荐成本.

表5中, n_{RP}^u, n_{BP}^u 和 n_{NP}^u 分别表示将用户 u 喜欢的项目推荐、延迟推荐和不推荐给该用户的项目数量; n_{RN}^u, n_{BN}^u 和 n_{NN}^u 分别表示将用户 u 不喜欢的项目推荐、延迟推荐和不推荐给该用户的项目数量. 则用户 u 的三支推荐准确性 ($Accuracy_u$) 定义为^[36]:

$$Accuracy_u = \frac{n_{RP}^u + n_{NN}^u}{n_{RP}^u + n_{NP}^u + n_{RN}^u + n_{NN}^u} \quad (9)$$

基于表4和表5,用户 u 的推荐成本 (TC_u) 可表达为:

$$TC_u = \lambda_{RP} n_{RP}^u + \lambda_{BP} n_{BP}^u + \lambda_{NP} n_{NP}^u + \lambda_{RN} n_{RN}^u + \lambda_{BN} n_{BN}^u + \lambda_{NN} n_{NN}^u \quad (10)$$

在三支推荐中,推荐准确性越高,推荐成本通常越低. 然而,在计算推荐准确性时(见式(9))没有考虑划分到延迟推荐域的项目,可能导致大量项目被划分到延迟推荐域,使延迟推荐的成本显著增加,产生准确性很高但推荐成本很大的问题,这是不符合优化目标的. 因此,在进行粒子群优

表5 三支推荐数量矩阵

Table 5 The matrix of three-way recommendation number

决策规则	用户对项目的偏好	
	喜欢(P)	不喜欢(N)
推荐(R)	n_{RP}^u	n_{RN}^u
延迟推荐(B)	n_{BP}^u	n_{BN}^u
不推荐(N)	n_{NP}^u	n_{NN}^u

化时,需要综合考虑准确性和推荐成本两个目标,即准确性越大越好,推荐成本越小越好. 本文使用线性加权法将二者合成一个单一的目标函数 $Goal_u$,优化目标为 $Goal_u$ 最小,具体如下:

$$Goal_u = \omega_1 TC_u / TC_u^{\max} - \omega_2 Accuracy_u \quad (11)$$

其中, TC_u^{\max} 表示对用户 u 进行推荐时可能产生的最大推荐成本, TC_u / TC_u^{\max} 表示对 TC_u 进行标准化, TC_u / TC_u^{\max} 和 $Accuracy_u$ 均在 $0 \sim 1$; ω_1 和 ω_2 分别表示推荐成本与推荐准确性的权重系数,满足 $\omega_1 + \omega_2 = 1$.

当用户 u 对应的 R_u^1, R_u^0 和 $R_u^{0.5}$ 给定时,可以计算用户 u 的偏好概率模型参数 d, ρ 和 q ,得到用户对应的偏好概率模型. 利用该模型对训练集的项目进行预测,可得到目标函数值 $Goal_u$. 当 R_u^1, R_u^0 和 $R_u^{0.5}$ 变化时, $Goal_u$ 也随着变化. 下面采用粒子群算法来确定使目标函数 $Goal_u$ 最小的评分组合 R_u^1, R_u^0 和 $R_u^{0.5}$,进而得到用户 u 的偏好概率模型.

粒子群优化算法^[37] (Particle Swarm Optimization, PSO) 是一种基于群体智能的优化算法,它模拟了鸟群、鱼群等生物集体行为的过程,通过迭代搜索找到最优解,具有较强的全局搜索能力和较好的收敛性. 在粒子群算法中,假设有 N 个粒子和 D 维空间,每个粒子 i ($i = 1, 2, \dots, N$) 拥有一个位置向量 $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ 和速度向量 $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. 在每次迭代中,粒子 i 的个体最优位置为 $Pb_i = (Pb_{i1}, Pb_{i2}, \dots, Pb_{iD})$,整个粒子群的全局最优位置为 $Gb = (Gb_1, Gb_2, \dots, Gb_D)$. 利用个体最优位置和全局最优位置,可以分别采用式(12)和式(13)更新粒子 i 的速度和位置:

$$v_{id}(t+1) = \eta(t) v_{id}(t) + c_1 r_1 (Pb_{id}(t) - x_{id}(t)) + c_2 r_2 (Gb_d(t) - x_{id}(t)) \quad (12)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (13)$$

其中, t 表示第 t 次迭代; c_1 和 c_2 表示学习因子, 通常设置为 2; r_1 和 r_2 表示两个在 $[0, 1]$ 中随机产生的数; $\eta(t)$ 为线性递减惯性权重, 满足^[38]:

$$\eta(t) = 0.9 - (0.9 - 0.4) \frac{t}{T_{\max}} \quad (14)$$

其中, T_{\max} 表示最大迭代次数.

本文中 $D = 3$, 将 R_u^0 , $R_u^{0.5}$ 和 R_u^1 分别作为每个粒子 i 的位置向量, 即令 $x_{i1} = R_u^0$, $x_{i2} = R_u^{0.5}$, $x_{i3} = R_u^1$, 以式(11)作为适应函数, 通过粒子群算法搜寻最优评分组合 R_u^0 , $R_u^{0.5}$ 和 R_u^1 , 具体步骤如下.

算法 基于 PSO 的用户偏好概率模型最优评分组合确定算法

输入:

1. 用户-项目评分信息, 推荐系统的评分最小值 \underline{R} 和最大值 \overline{R} ;
2. 给定三支推荐成本矩阵;
3. 初始化粒子群算法的参数, 包括适应函数式(11)、学习因子 c_1 和 c_2 、最大迭代次数 T_{\max} 、粒子维度 D 、粒子群总数 N .

输出: 用户的偏好概率模型最优评分组合 R_u^0 , $R_u^{0.5}$ 和 R_u^1 .

1. begin
2. for $i = 1$ to N do % 初始化粒子位置与速度
3. $x_{i1} = \underline{R}$;
4. $x_{i2} = (\underline{R} + \overline{R})/2$
5. $x_{i3} = \overline{R}$;
6. for $d = 1$ to D do
7. $v_{id} = 2 \times \text{rand} - 1$
8. end
9. end
10. 基于粒子初始位置、式(8)与成本矩阵求得目标函数值 $Goal_u$
11. for $i = 1$ to N do
12. $P_{\text{best}_i} = Goal_u$ % 初始化每个粒子的最优目标函数值
13. $Pb_{id} = X_i$ % 初始化每个粒子的最优位置
14. end
15. $G_{\text{best}} = Goal_u$ % 初始化所有粒子的最优目标函数值
16. for $d = 1$ to D do
17. $Gb_d = x_{i1}$ % 初始化所有粒子的最优位置
18. end
19. $t = 1$ % 记录迭代次数

20. while $t \leq T_{\max}$ do
21. $\eta(t) = 0.9 - (0.9 - 0.4) \times t/T_{\max}$
22. for $i = 1$ to N do
23. for $d = 1$ to D do
24. 用式(12)更新速度 $v_{id}(t+1)$
25. 当 $v_{id}(t+1) > 1$ 或 $v_{id}(t+1) < -1$ 时调整速度
26. 用式(13)更新位置 $x_{id}(t+1)$
27. 当 $x_{id}(t+1) > \overline{R}$ 或 $x_{id}(t+1) < \underline{R}$ 时调整位置
28. end
29. 基于标准化后的粒子位置和式(8)确定用户此时的偏好概率模型参数, 进而求出目标函数值 $Goal_u$;
30. if $P_{\text{best}_i} > Goal_u$ then
31. $P_{\text{best}_i} = Goal_u$;
32. $Pb_i = X_i(t+1)$ % 更新粒子最优位置
33. end
34. if $G_{\text{best}} > P_{\text{best}_i}$ then
35. $G_{\text{best}} = P_{\text{best}_i}$;
36. $Gb = Pb_i$ % 更新所有粒子最优位置
37. end
38. end
39. end
40. 输出 Gb 中的用户偏好概率模型最优评分组合 R_u^0 , $R_u^{0.5}$ 和 R_u^1
41. end

基于上述算法得到用户最优评分组合, 利用式(8)可得到参数 q , d 和 ρ , 进而得到用户的偏好概率模型.

3 实验分析

3.1 数据集 使用 MovieLens-100k 电影评分数据集 (<https://grouplens.org/datasets/movielens/latest>)^[39] 进行实验, 该数据集的评分在 0.5~5, 评分步长为 0.5. 将 60% 的数据随机划分为预测评分训练集, 将 20% 的数据划分为用户偏好概率模型训练集, 剩下 20% 的数据为测试集. 同时, 参考文献[40], 清洗评分量过少且不具有意义的数据, 所得数据集的基本信息如表 6 所示.

3.2 实验设置 对于提出的融入用户风险偏好的三支协同过滤推荐(P_CF), 首先, 基于协同过滤算法, 利用预测评分训练集预测剩余 40% 的用

表6 MovieLens-100k数据集的描述

Table 6 The description of MovieLens-100k dataset

数据集	用户数	项目数	评分数量	稀疏度
MovieLens-100k	599	1650	73713	92.54%

户-项目集的评分. 然后, 基于前文算法与式(8), 利用用户偏好概率模型训练集得出每个用户的偏好概率模型, 其中, 经调参后目标函数式(11)中的权重取 $\omega_1 = 0.1$, $\omega_2 = 0.9$, 三支推荐成本矩阵取 $\lambda_{RP} = \lambda_{NN} = 0$, $\lambda_{BP} = \lambda_{BN} = 0.25$ 和 $\lambda_{RN} = \lambda_{NP} = 0.8$. 最后, 对测试集进行三支推荐, 将P_CF的推荐结果与基于最大最小值归一化模型^[16,18,21]的三支协同过滤推荐算法, 即将用户视为风险中立型的算法(M_CF)的结果进行对比.

为了衡量推荐结果, 需要判断用户对项目的真实偏好. 对于五分制评分系统, 遵循推荐系统研究的通常做法^[40-41], 将用户的喜好阈值设置为3.5分.

选取推荐成本(TC)、准确性(Accuracy)、覆盖率(Generality)、精准度(Precision)、召回率(Recall)和得分(F)作为衡量指标^[18,41-42]. 精准度、召回率和得分是传统二支推荐的常用指标, 也适用于三支推荐; 准确性和覆盖率更适合用来衡量三支推荐的质量, 准确性指三支推荐将项目划分到推荐和不推荐域时正确划分的比例, 覆盖率指划分到推荐和不推荐域的项目占所有待推荐项目的比例, 覆盖率越高表示越多用户得到系统的推荐. 通常, 准确性和覆盖率之间存在权衡, 即为了给更多的用户推荐项目, 需要牺牲一部分准确性; 反之, 为了更准确地向用户推荐项目, 需要牺牲一部分覆盖率^[36]. 上述指标的定义如下:

$$TC = \lambda_{RP}n_{RP} + \lambda_{BP}n_{BP} + \lambda_{NP}n_{NP} + \lambda_{RN}n_{RN} + \lambda_{BN}n_{BN} + \lambda_{NN}n_{NN} \quad (15)$$

$$Generality = \frac{n_{all} - n_{BP} - n_{BN}}{n_{all}} \quad (16)$$

$$Accuracy = \frac{n_{RP} + n_{NN}}{n_{RP} + n_{NP} + n_{RN} + n_{NN}} \quad (17)$$

$$Precision = \frac{n_{RP}}{n_{RP} + n_{RN}} \quad (18)$$

$$Recall = \frac{n_{RP}}{n_{RP} + n_{NP}} \quad (19)$$

$$F = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (20)$$

其中,

$$n_{all} = n_{RP} + n_{BP} + n_{NP} + n_{RN} + n_{BN} + n_{NN}$$

结合表5, n_{RP} , n_{BP} , n_{NP} , n_{RN} , n_{BN} 和 n_{NN} 的表达式分别为:

$$\begin{aligned} n_{RP} &= \sum_{u \in U} n_{RP}^u & n_{BP} &= \sum_{u \in U} n_{BP}^u & n_{NP} &= \sum_{u \in U} n_{NP}^u \\ n_{RN} &= \sum_{u \in U} n_{RN}^u & n_{BN} &= \sum_{u \in U} n_{BN}^u & n_{NN} &= \sum_{u \in U} n_{NN}^u \end{aligned}$$

其中, U 表示所有用户的集合.

本文主要考虑错误推荐或错误不推荐产生的误分类成本和延迟推荐产生的学习成本. 假设系统正确推荐时不产生推荐成本, 即 $\lambda_{RP} = \lambda_{NN} = 0$, 则式(15)可简化为:

$$TC = \lambda_{BP}n_{BP} + \lambda_{NP}n_{NP} + \lambda_{RN}n_{RN} + \lambda_{BN}n_{BN} \quad (21)$$

3.3 实验结果分析 表7列出了 $\lambda_{NP} = 0.8$ 时, P_CF 与 M_CF 在不同的 λ_{RN} , λ_{BP} 和 λ_{BN} 成本条件下的推荐成本; 表8列出了对应成本条件下两个算法的推荐覆盖率; 表9和表10分别列出了 $\lambda_{RN} = 0.8$ 时, 两个算法在不同的 λ_{NP} , λ_{BP} 和 λ_{BN} 成本条件下的推荐成本和覆盖率. 表中黑体字表示结果更优. 综合对比四个表格, 分析如下:

(1) 由表7和表9可知, 学习成本或误分类成本越低, 两个算法的推荐成本均越低; 由表8和表10可知, 学习成本越低或误分类成本越高, 两个算法的覆盖率水平越低, 即能获得推荐的用户越少.

(2) 当学习成本较高或误分类成本较低时, P_CF 的推荐成本低于 M_CF, 即使 P_CF 的覆盖率更高时也是如此.

(3) 当学习成本较低且误分类成本较高时, M_CF 的推荐成本总体占优, 但优势不明显. 结合表7~10可以看出, 当 M_CF 的推荐成本占优时, 其对应的覆盖率均明显低于 P_CF. 以成本为 $\lambda_{BP} = \lambda_{BN} = 0.15$, $\lambda_{RN} = 1$ 且 $\lambda_{NP} = 0.8$ 时为例, M_CF 的推荐成本比 P_CF 低 2.7%, 但其覆盖率比 P_CF 低 12.5%. 所以, M_CF 的优势可能是因为它将大量的项目划分到延迟推荐域, 延迟推荐的成本较低时其推荐成本也较低, 延迟推荐成本较高时, 这个优势就不明显, 甚至消失. 例如, 当成本为 $\lambda_{BP} = \lambda_{BN} = 0.25$, $\lambda_{RN} = 1.2$ 且 $\lambda_{NP} = 0.8$ 时, P_CF 的覆盖率比 M_CF 高 12.9%, 而其对应成本也低于 M_CF.

因此, 本文提出的融入用户风险偏好的算法

表 7 固定 λ_{NP} 时推荐成本的比较Table 7 Recommendation cost with fixed λ_{NP}

$(\lambda_{RN}, \lambda_{NP})$	$\lambda_{BR} = \lambda_{BN} = 0.35$		$\lambda_{BR} = \lambda_{BN} = 0.25$		$\lambda_{BR} = \lambda_{BN} = 0.15$	
	M_CF	P_CF	M_CF	P_CF	M_CF	P_CF
(1.2, 0.8)	4582.4	4223.1	3526.9	3454.2	2234.3	2345.1
(1.0, 0.8)	4504.5	4060.4	3471.7	3366.6	2208.8	2269.1
(0.8, 0.8)	4353.9	3816.0	3417.1	3217.9	2164.2	2200.1
(0.6, 0.8)	—	—	3306.0	3019.8	2111.6	2120.2
(0.4, 0.8)	—	—	2753.3	2524.3	2048.8	1957.5

注:“—”表示三支决策退化成二支决策的情况

表 8 固定 λ_{NP} 时推荐覆盖率的比较Table 8 Recommendation generality with fixed λ_{NP}

$(\lambda_{RN}, \lambda_{NP})$	$\lambda_{BR} = \lambda_{BN} = 0.35$		$\lambda_{BR} = \lambda_{BN} = 0.25$		$\lambda_{BR} = \lambda_{BN} = 0.15$	
	M_CF	P_CF	M_CF	P_CF	M_CF	P_CF
(1.2, 0.8)	0.468	0.570	0.204	0.333	0.064	0.193
(1.0, 0.8)	0.614	0.658	0.290	0.381	0.089	0.214
(0.8, 0.8)	0.813	0.807	0.443	0.475	0.142	0.247
(0.6, 0.8)	—	—	0.691	0.646	0.265	0.318
(0.4, 0.8)	—	—	0.969	0.949	0.571	0.512

注:“—”表示三支决策退化成二支决策的情况

表 9 固定 λ_{RN} 时推荐成本的比较Table 9 Recommendation cost with fixed λ_{RN}

$(\lambda_{RN}, \lambda_{NP})$	$\lambda_{BR} = \lambda_{BN} = 0.35$		$\lambda_{BR} = \lambda_{BN} = 0.25$		$\lambda_{BR} = \lambda_{BN} = 0.15$	
	M_CF	P_CF	M_CF	P_CF	M_CF	P_CF
(0.8, 1.2)	4603.7	4218.8	3456.0	3359.4	2177.2	2228.7
(0.8, 1.0)	4541.7	4077.1	3446.2	3316.3	2171.7	2214.4
(0.8, 0.8)	4353.9	3816.0	3417.1	3217.9	2164.2	2200.1
(0.8, 0.6)	—	—	3286.9	3025.6	2153.1	2158.9
(0.8, 0.4)	—	—	2644.2	2491.0	2095.9	2040.1

注:“—”表示三支决策退化成二支决策的情况

表 10 固定 λ_{RN} 时推荐覆盖率的比较Table 10 Recommendation generality with fixed λ_{RN}

$(\lambda_{RN}, \lambda_{NP})$	$\lambda_{BR} = \lambda_{BN} = 0.35$		$\lambda_{BR} = \lambda_{BN} = 0.25$		$\lambda_{BR} = \lambda_{BN} = 0.15$	
	M_CF	P_CF	M_CF	P_CF	M_CF	P_CF
(0.8, 1.2)	0.729	0.667	0.421	0.419	0.136	0.233
(0.8, 1.0)	0.752	0.712	0.426	0.437	0.138	0.238
(0.8, 0.8)	0.813	0.807	0.443	0.475	0.142	0.247
(0.8, 0.6)	—	—	0.502	0.572	0.150	0.272
(0.8, 0.4)	—	—	0.847	0.900	0.198	0.362

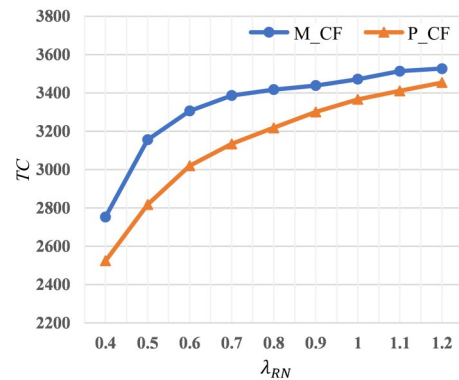
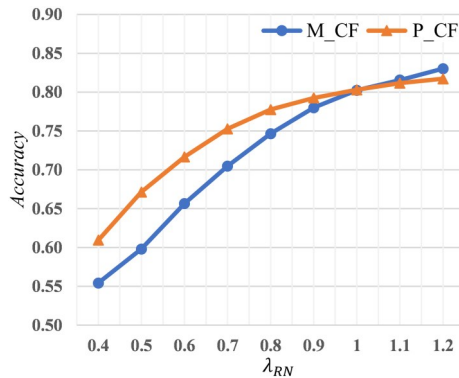
注:“—”表示三支决策退化成二支决策的情况

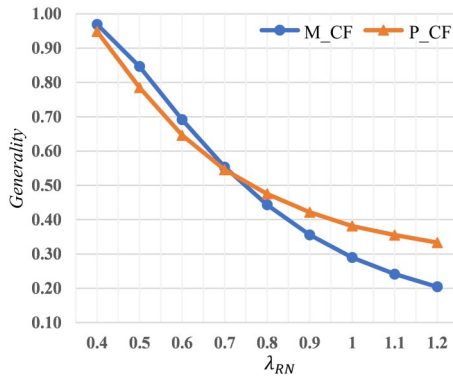
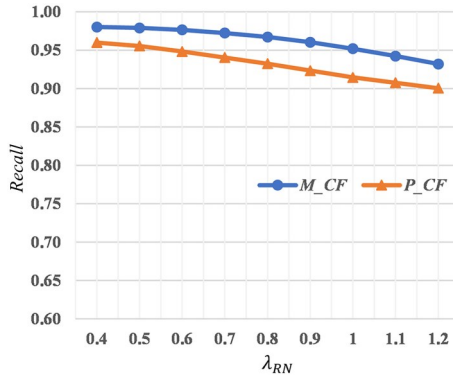
在学习成本较高或误分类成本较低时,在保证覆盖率的前提下能使系统的推荐成本更低,在学习成本较低且误分类成本较高时,能大幅提高覆盖率,但会小幅牺牲成本。

进一步,选取一组有代表性的成本条件来对比两个算法的推荐质量。设置 $\lambda_{NP} = 0.8$ 和 $\lambda_{BP} = \lambda_{BN} = 0.25$, 图 3~8 给出了两个算法在不同的错误推荐成本 (λ_{RN}) 下各指标的对比。分析如下:

(1) 随着错误推荐成本的增加,两个算法的推荐成本、准确性、精准度和得分均变大,推荐覆盖率和召回率变小,其中,召回率的变化幅度较小,始终保持在较高的水平。

(2) 当 λ_{RN} 不大于 0.8 时, M_CF 的推荐覆盖率大于 P_CF, 但除召回率之外, M_CF 的其他指标均逊色于 P_CF。当 M_CF 的覆盖率占优时,一般意味着它的推荐成本、准确性和精准度处于更

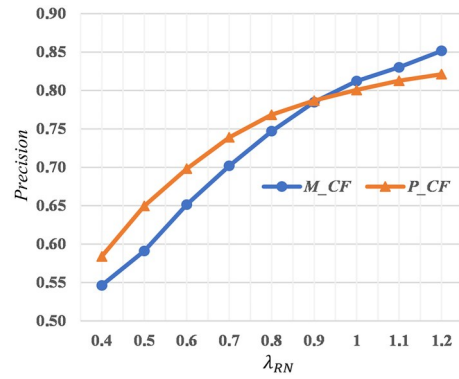
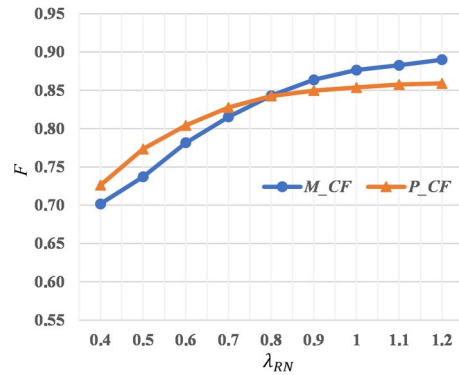
图 3 $\lambda_{NP} = 0.8$ 和 $\lambda_{BP} = \lambda_{BN} = 0.5$ 时推荐成本的比较Fig. 3 Recommendation cost with $\lambda_{NP} = 0.8$ and $\lambda_{BP} = \lambda_{BN} = 0.5$ 图 4 $\lambda_{NP} = 0.8$ 和 $\lambda_{BP} = \lambda_{BN} = 0.5$ 时推荐准确性的比较Fig. 4 Recommendation accuracy with $\lambda_{NP} = 0.8$ and $\lambda_{BP} = \lambda_{BN} = 0.5$

图5 $\lambda_{NP}=0.8$ 和 $\lambda_{BP}=\lambda_{BN}=0.5$ 时推荐覆盖率的比较Fig. 5 Recommendation generality with $\lambda_{NP}=0.8$ and $\lambda_{BP}=\lambda_{BN}=0.5$ 图7 $\lambda_{NP}=0.8$ 和 $\lambda_{BP}=\lambda_{BN}=0.5$ 时推荐召回率的比较Fig. 7 Recommendation recall with $\lambda_{NP}=0.8$ and $\lambda_{BP}=\lambda_{BN}=0.5$

大的劣势.例如, $\lambda_{RN}=0.6$ 时,M_CF的覆盖率比P_CF高4.5%,但其成本、准确性和精准度分别比P_CF低8.7%,6.0%和4.7%.

(3)当 λ_{RN} 不小于0.8时,M_CF的推荐覆盖率小于P_CF,且二者的差距随着 λ_{RN} 的增大而越来越大.与M_CF的覆盖率占优时不同, $\lambda_{RN}<1$ 时,P_CF的准确性仍然高于M_CF; $\lambda_{RN}>1$ 时,P_CF的覆盖率比M_CF高10%左右,其准确性只比M_CF低不到2%,精准度低不到4%.

因此,与仅将用户考虑为风险中立型的算法相比,当错误推荐的成本较低时,融入用户风险偏好的算法能在牺牲小部分覆盖率的同时,较大幅度地提高推荐准确性与精准度;当错误推荐的成本较高时,本文提出的算法能在牺牲小部分甚至不牺牲准确性与精准度的同时,较大幅度地提高推荐覆盖率.

图6 $\lambda_{NP}=0.8$ 和 $\lambda_{BP}=\lambda_{BN}=0.5$ 时推荐精确度的比较Fig. 6 Recommendation precision with $\lambda_{NP}=0.8$ and $\lambda_{BP}=\lambda_{BN}=0.5$ 图8 $\lambda_{NP}=0.8$ 和 $\lambda_{BP}=\lambda_{BN}=0.5$ 时推荐得分的比较Fig. 8 Recommendation F-measure with $\lambda_{NP}=0.8$ and $\lambda_{BP}=\lambda_{BN}=0.5$

综上,本文提出的融入用户风险偏好的三支协同过滤推荐模型更适用于对推荐成本、推荐覆盖率、推荐准确性和推荐精准度综合考虑的场景.

4 结论

为了优化三支推荐规则的准确性,本文提出融入用户风险偏好的三支协同过滤推荐模型,构建了一个能够预测用户对项目的偏好概率的模型,并基于粒子群算法给出了确定用户偏好概率模型的算法.实验结果表明,本文提出的方法在不同的初始条件下能够保持推荐成本、推荐覆盖率、准确性与精准度的相对优势.未来会将本文提出的用户-项目偏好概率模型与更多的三支推荐算法相结合,进一步验证该模型的可拓展性.

参考文献

- [1] Aggarwal C C. Recommender systems: The textbook. Cham: Springer International Publishing, 2016, 498.
- [2] Patel K, Patel H B. A state-of-the-art survey on recommendation system and prospective extensions. Computers and Electronics in Agriculture, 2020(178):105779.
- [3] Zhang H R, Min F. Three-way recommender systems based on random forests. Knowledge-Based Systems, 2016(91):275–286.
- [4] Zhang H R, Min F, Shi B. Regression-based three-way recommendation. Information Sciences, 2017(378):444–461.
- [5] Liu D, Liang D C, Wang C C. A novel three-way decision model based on incomplete information system. Knowledge-Based Systems, 2016(91):32–45.
- [6] Yao Y Y. Three-way decisions and cognitive computing. Cognitive Computation, 2016, 8(4):543–554.
- [7] Liu J B, Li H X, Zhou X Z, et al. An optimization-based formulation for three-way decisions. Information Sciences, 2019(495):185–214.
- [8] Xu J F, Zhang Y J, Miao D Q. Three-way confusion matrix for classification: A measure driven view. Information Sciences, 2020(507):772–794.
- [9] Liu J B, Huang B, Li H X, et al. Optimization-based three-way decisions with interval-valued intuitionistic fuzzy information. IEEE Transactions on Cybernetics, 2023, 53(6):3829–3843.
- [10] Yao Y Y. Three-way decisions with probabilistic rough sets. Information Sciences, 2010, 180(3):341–353.
- [11] Liu J B, Hu S L, Li H X, et al. Achieving threshold consistency in three-way group decision using optimization methodology and expert-weight-updating-strategy. International Journal of Approximate Reasoning, 2023(158):108922.
- [12] Yao J T, Azam N. Web-based medical decision support systems for three-way medical decision making with game-theoretic rough sets. IEEE Transactions on Fuzzy Systems, 2015, 23(1):3–5.
- [13] Liu D, Li T R, Liang D C. Three-way government decision analysis with decision-theoretic rough sets. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2012, 20(S1):119–132.
- [14] Li H X, Zhang L B, Huang B, et al. Sequential three-way decision and granulation for cost-sensitive face recognition. Knowledge-Based Systems, 2016(91):241–251.
- [15] Campagner A, Ciucci D, Svensson C M, et al. Ground truthing from multi-rater labeling with three-way decision and possibility theory. Information Sciences, 2021(545):771–790.
- [16] 叶晓庆, 刘盾, 梁德翠. 基于协同过滤的三支粒推荐算法研究. 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. Computer Science, 2018, 45(1):90–96.)
- [17] Ma Y Y, Zhang H R, Xu Y Y, et al. Three-way recommendation integrating global and local information. The Journal of Engineering, 2018(16):1397–1401.
- [18] Liu D, Ye X Q. A matrix factorization based dynamic granularity recommendation with three-way decisions. Knowledge-Based Systems, 2020(191):105243.
- [19] 秦琴, 张恒汝. 基于信任传递机制的三支推荐. 模式识别与人工智能, 2020, 33(7):600–609. (Qin Q, Zhang H R. Three-way recommendation based on trust transfer mechanism. Pattern Recognition and Artificial Intelligence, 2020, 33(7):600–609.)
- [20] Ye X Q, Liu D. An interpretable sequential three-way recommendation based on collaborative topic regression. Expert Systems with Applications, 2021(168):114454.
- [21] Zhang C Y, Duan X M, Liu F C, et al. Three-way Naive Bayesian collaborative filtering recommendation model for smart city. Sustainable Cities and Society, 2022(76):103373.
- [22] Ye X Q, Liu D. A cost-sensitive temporal-spatial three-way recommendation with multi-granularity decision. Information Sciences, 2022(589):670–689.
- [23] Ye X Q, Liu D, Li T R. Multi-granularity sequential three-way recommendation based on collaborative deep learning. International Journal of Approximate Reasoning, 2023(152):434–455.

- [24] Zhang Q H, Xie Q, Wang G Y. A novel three-way decision model with decision - theoretic rough sets using utility theory. *Knowledge-Based Systems*, 2018 (159):321—335.
- [25] Sarwar B, Karypis G, Konstan J, et al. Analysis of recommendation algorithms for E - commerce// *Proceedings of the 2nd ACM Conference on Electronic Commerce*. Minneapolis, MN, USA: Association for Computing Machinery, 2000: 158—167.
- [26] Sarwar B, Karypis G, Konstan J, et al. Item - based collaborative filtering recommendation algorithms// *Proceedings of the 10th International Conference on World Wide Web*. Hong Kong, China: Association for Computing Machinery, 2001:285—295.
- [27] 于洪,王国胤,姚一豫. 决策粗糙集理论研究现状与展望. *计算机学报*, 2015, 38(8):1628—1639. (Yu H, Wang G Y, Yao Y Y. Current research and future perspectives on decision - theoretic rough sets. *Chinese Journal of Computers*, 2015, 38(8): 1628—1639.)
- [28] 王国胤,姚一豫,于洪. 粗糙集理论与应用研究综述. *计算机学报*, 2009, 32(7):1229—1246. (Wang G Y, Yao Y Y, Yu H. A survey on rough set theory and applications. *Chinese Journal of Computers*, 2009, 32 (7):1229—1246.)
- [29] Yao Y Y. Decision - theoretic rough set models// *Proceedings of the 2nd International Conference on Rough Sets and Knowledge Technology*. Springer Berlin Heidelberg, 2007:1—12.
- [30] Li H X, Zhou X Z. Risk decision making based on decision - theoretic rough set: A three - way view decision model. *International Journal of Computational Intelligence Systems*, 2011, 4(1):1—11.
- [31] 刘久兵,顾萍萍,周献中,等. 基于优化模型的直觉模糊三支群决策方法. *南京大学学报(自然科学)*, 2018, 54(5):944—957. (Liu J B, Gu P P, Zhou X Z, et al. A method for intuitionistic fuzzy three - way group decisions based on optimization models. *Journal of Nanjing University (Natural Science)*, 2018, 54 (5):944—957.)
- [32] 李华,胡奇英. 预测与决策教程. 北京:机械工业出版社, 2012, 349.
- [33] 冯文权,傅征. 经济预测与决策技术. 第6版. 武汉: 武汉大学出版社, 2018, 405.
- [34] 袁捷敏. 决策效用函数两类拟合方法比较. *统计与决策*, 2006(2): 49—50. (Yuan J M. Comparison of two kinds of fitting methods for decision utility function. *Statistics & Decision*, 2006(2):49—50.)
- [35] 陈秀琼,黄福才. 旅行效用概率模型与旅游业滞区发展策略研究. *浙江大学学报(理学版)*, 2007, 34(6): 702—708, 715. (Chen X Q, Huang F C. Research on the probabilistic travel utility model and the strategy of developing tourism in stagnation area. *Journal of Zhejiang University (Science Edition)*, 2007, 34(6): 702—708, 715.)
- [36] Azam N, Yao J T. Game - theoretic rough sets for recommender systems. *Knowledge-Based Systems*, 2014(72):96—107.
- [37] Shami T M, El - Saleh A A, Alswaiti M, et al. Particle swarm optimization: A comprehensive survey. *IEEE Access*, 2022(10):10031—10061.
- [38] Shi Y, Eberhart R C. Empirical study of particle swarm optimization//*Proceedings of 1999 Congress on Evolutionary Computation - CEC99*. Washington D C, USA:IEEE, 1999:1945—1950.
- [39] Harper F M, Konstan J A. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 2016, 5(4):19.
- [40] Adomavicius G, Kwon Y O. Improving aggregate recommendation diversity using ranking - based techniques. *IEEE Transactions on Knowledge and Data Engineering*, 2012, 24(5):896—911.
- [41] Herlocker J L, Konstan J A, Terveen L G, et al. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 2004, 22(1):5—53.
- [42] Wu C Y, Zhang Q H, Zhao F, et al. Three - way recommendation model based on shadowed set with uncertainty invariance. *International Journal of Approximate Reasoning*, 2021(135):53—70.

(责任编辑 杨可盛)