

模糊概念集的启发式构造方法及其推荐应用

刘忠慧¹, 姜帅¹, 闵帆^{1,2*}

(1.西南石油大学计算机科学学院, 四川 成都 610500; 2.西南石油大学人工智能研究院, 四川 成都 610500)

摘要:针对模糊形式概念分析在推荐应用中难以用于大规模数据集的问题,提出了一种基于模糊概念集启发式构造的推荐方法。根据用户之间的相似度,为每个用户构建子背景,在子背景上采用新的启发式信息,分别以用户和项目为线索生成模糊概念。利用模糊概念内部信息,设计了融入用户权重的推荐置信度,实现了对用户的个性化推荐。在6个真实数据集上进行试验,本方法的推荐效率较高,与经典的协同过滤算法相比,在稀疏的数据集上能够取得更好的推荐效果。

关键词:形式概念分析;模糊概念;概念构造;推荐系统;用户相似度

中图分类号: TP181 **文献标志码:** A

引用格式: 刘忠慧,姜帅,闵帆. 模糊概念集的启发式构造方法及其推荐应用[J]. 山东大学学报(理学版), 2024, 59(3): 14-26.

Heuristic construction method of fuzzy concept set and its recommended application

LIU Zhonghui¹, JIANG Shuai¹, MIN Fan^{1,2*}

(1. School of Computer Science, Southwest Petroleum University, Chengdu 610500, Sichuan, China; 2. Institute for Artificial Intelligence, Southwest Petroleum University, Chengdu 610500, Sichuan, China)

Abstract: Aiming at the problem that fuzzy formal concept analysis is difficult to apply to large-scale datasets in recommendation applications, a recommendation method based on a heuristic construction of fuzzy concept set is proposed. Sub-contexts are constructed for each user based on the similarity between users. Then, new heuristic information is used on the sub-contexts to generate fuzzy concepts with users and items as clues, respectively. Finally, using the internal information of fuzzy concepts, a recommendation confidence integrated with user weights is designed to achieve personalized recommendations for users. The experimental results on six real datasets show that the proposed method has higher recommendation efficiency, and can achieve better recommendation results on sparse data sets compared with classical collaborative filtering algorithms.

Key words: formal concept analysis; fuzzy concept; concept construction; recommender system; user similarity

0 引言

形式概念分析(formal concept analysis, FCA)^[1]是由德国数学家 Wille 于 1982 年提出的一种数据分析与知识处理方法,通过描述对象与属性之间的二元关系、概念与概念之间的层次关系,反映了事物之中隐含的知识结构与内在联系。目前,形式概念分析已在机器学习^[2-3]、数据挖掘^[4]、软件工程^[5]、推荐系统^[6-7]等领域得到广泛应用。

经典的形式概念只能描述确定的关系,但在客观现实中,信息多是不确定、模糊的,因此一些学者将经典概念扩展到了模糊集^[8]上,形成了模糊形式概念分析(fuzzy formal concept analysis, FFCA)^[9]。Burusco 等^[10]将形式概念分析推广到模糊背景中,引入了 L-模糊概念;刘宗田等^[11]提出了一种带有模糊参数的模糊

收稿日期: 2023-04-28; 网络出版时间: 2023-10-18 10:53:53

网络出版地址: <https://link.cnki.net/urlid/37.1389.N.20231017.0930.002>

基金项目: 国家自然科学基金资助项目(61976245); 中央引导地方科技发展专项资助项目(2021ZYD0003)

第一作者: 刘忠慧(1980—),女,教授,硕士生导师,硕士,研究方向为机器学习、形式概念分析与粗糙集等。E-mail: lz_hui@126.com

* 通信作者: 闵帆(1973—),男,教授,博士生导师,博士,研究方向为粒计算、主动学习与多标签学习等。E-mail: minfan@swpu.edu.cn

概念格模型,并给出了其渐近式构造算法;Krajci^[12]和Yahia等^[13]定义并研究了单边模糊概念;Zhang等^[14]采用模糊截集的思想,定义了3种变精度概念格。模糊形式概念分析理论发展迅速,研究方向主要包括模糊概念格构造^[15]、知识约简^[16]、概念认知学习^[17-18]、区间值模糊概念格^[19]等。

在推荐系统领域,形式概念分析由于能够在高稀疏度条件下描述用户与项目之间的关系,因此受到了学者的青睐。例如,Zou等^[6]结合关联规则挖掘算法与协同过滤推荐算法,设计了一种基于概念格的个性化推荐系统;Fang等^[20]将模糊逻辑用于模糊概念,提出了一种基于模糊形式概念分析的协同过滤推荐方法;张喜征等^[21]通过构建带有相似度的模糊概念格,将模糊形式概念分析用于个性化知识推荐。基于形式概念分析的推荐大多通过概念格来实现,但是格构造存在高时空复杂度问题。为解决此问题,Liu等^[22]提出了一种通过启发式方式构建概念集合进行协同过滤推荐的方法,利用概念集合替代概念格,提高了推荐效率,但是,该方法只研究了经典概念的启发式生成,并未考虑模糊概念的生成情况,且生成时忽略了用户之间的关系,生成的概念外延与内涵极度不平衡。

针对上述问题,本文对概念生成的启发式方式进行了改进,提出了一种基于模糊概念集启发式构造的推荐方法。主要贡献有以下2个方面:1)构建用户子背景,采用启发式方法生成模糊概念集合。根据用户之间的相似度,为每个用户提取子背景,然后在子背景上采用新的启发式信息,分别以用户和项目为线索生成模糊概念,降低时间复杂度。2)利用模糊概念集合实现用户推荐。利用模糊概念内部信息,设计了融入用户权重的推荐置信度,实现了对用户的个性化推荐。通过试验验证,本方法在大规模数据集上具有良好的性能与较好的推荐效果。

1 相关工作

本节主要介绍形式概念分析与模糊形式概念分析中的相关理论与知识,以及模糊形式概念分析在推荐系统中的相关应用。主要符号见表1。

表1 主要符号
Table 1 Main notations

符号	含义	符号	含义
U	用户集	C	概念集合
M	属性集	$H_u = (U_u, M_u, \tilde{I}_u)$	用户 u 子背景
I	二元关系	$D(u_1, u_2)$	用户相似度
$K = (U, M, I)$	形式背景	$S(c)$	模糊概念大小
\tilde{I}	模糊关系	$w_c(u_1, u_2)$	用户权重
$H = (U, M, \tilde{I})$	模糊形式背景	$R_c(u, m)$	推荐置信度
$\mu(*)$	隶属度	A_u	用户 u 线索集合
$\varphi(*)$	模糊集	B_m	项目 m 线索集合
$c = (\varphi(A), B)$	模糊概念	L	推荐矩阵

1.1 形式概念

形式概念分析的基础是形式背景,通过将形式背景转化为概念格,从而进行知识发现与信息提取。相关理论定义如下。

定义1 形式背景^[23]。一个形式背景是一个三元组 $K = (U, M, I)$,其中 U 为用户集, M 为项目集, $I \subseteq U \times M$ 为 U 与 M 之间的二元关系。对任意的 $u \in U, m \in M, (u, m) \in I$ 或 ulm 表示用户 u 拥有项目 m 。

定义2 f/g 算子^[23]。在形式背景 $K = (U, M, I)$ 中,令 $A \subseteq U, B \subseteq M$,则算子 f 与 g 分别定义为

$$f(A) = \{m \in M \mid \forall u \in A, ulm\}, \tag{1}$$

$$g(B) = \{u \in U \mid \forall m \in B, ulm\}. \tag{2}$$

上述 $f(A)$ 表示的是 A 中用户共同项目的集合, $g(B)$ 表示的是拥有 B 中所有项目的用户集合。

定义3 形式概念^[23]。设 $K = (U, M, I)$ 为形式背景, $A \subseteq U, B \subseteq M$,若满足 $f(A) = B$ 且 $g(B) = A$,则称二元组 (A, B) 为形式概念,简称概念,其中 A 与 B 分别称为概念的外延与内涵。

表2为一个形式背景示例,共记录了5个用户对5个项目的拥有情况。表中1表示拥有,0表示不拥有,如用户 u_1 拥有项目 m_2 ,用户 u_4 不拥有项目 m_3 。

表2 形式背景
Table 2 Formal context

U	M				
	m_1	m_2	m_3	m_4	m_5
u_1	1	1	1	0	0
u_2	1	1	1	1	1
u_3	1	0	1	0	1
u_4	1	1	0	1	0
u_5	0	0	1	0	0

从表2可以导出一个概念($\{u_1, u_2\}, \{m_1, m_2, m_3\}$),其中外延为 $\{u_1, u_2\}$,内涵为 $\{m_1, m_2, m_3\}$ 。

1.2 模糊形式概念

模糊形式概念分析的基础是模糊形式背景,在模糊形式背景中,用户与项目之间的关系不再是确定的0或1,而是通过处于 $[0, 1]$ 的一个隶属度来描述。这种方式使其能够用于连续型数据当中,扩展了形式概念的应用场景。

定义4 模糊集^[24]。设 U 是一个用户集合,则集合 U 上的模糊集 $\varphi(U)$ 定义为

$$\varphi(U) = \{(x, \mu_{\varphi(U)}(x)) \mid x \in U\}, \quad (3)$$

其中 $\mu_{\varphi(U)}(x): U \rightarrow [0, 1]$,且 $\mu_{\varphi(U)}(x)$ 被称为用户 $x \in U$ 关于 $\varphi(U)$ 的隶属度。

定义5 模糊形式背景^[9]。一个模糊形式背景是一个三元组 $H = (U, M, \tilde{I})$,其中 U 为用户集, M 为项目集, \tilde{I} 为集合 U 与 M 之间的模糊关系,即存在一个映射 $\mu: U \times M \rightarrow [0, 1]$,对任意的 $u \in U, m \in M$,均有一个隶属度 $\mu(u, m) \in [0, 1]$ 。

模糊概念是模糊形式概念分析应用的基础,目前模糊概念主要分为2种:一种是外延与内涵都为模糊集;另一种是外延或内涵为模糊集,这种模糊概念也称为单边模糊概念^[25-26]。本文选择外延为模糊集,内涵为经典集的模糊概念应用到推荐系统。相关定义如下。

定义6 ($*$, $*$)算子^[9]。给定一个模糊形式背景 $H = (U, M, \tilde{I})$ 以及隶属度阈值 $T, A \subseteq U, B \subseteq M$,则一对($*$, $*$)算子定义为

$$A^* = \{m \in M \mid \forall u \in A, \mu(u, m) \geq T\}, \quad (4)$$

$$B^* = \{u \in U \mid \forall m \in B, \mu(u, m) \geq T\}, \quad (5)$$

其中 $T \in [0, 1]$ 。

定义7 模糊形式概念^[9]。给定一个模糊形式背景 $H = (U, M, \tilde{I})$ 与隶属度阈值 $T, A \subseteq U, B \subseteq M$,若满足 $A^* = B$ 和 $B^* = A$,则二元组 $(\varphi(A), B)$ 是一个模糊形式概念,简称模糊概念。其中 $\varphi(A)$ 是定义在 A 上的一个模糊集,每个用户 $u \in A$ 均有一个隶属度 $\mu_{\varphi(A)}(u)$,定义为

$$\mu_{\varphi(A)}(u) = \min_{m \in B} \mu(u, m). \quad (6)$$

如果 $B = \emptyset$,则对任意 u 均有 $\mu_{\varphi(A)}(u) = 1$ 。

表3是一个模糊形式背景 $H = (U, M, \tilde{I})$,其中 $U = \{u_1, u_2, u_3, u_4, u_5\}$ 和 $M = \{m_1, m_2, m_3, m_4, m_5\}$ 。表中的数值为隶属度代表了用户拥有项目的程度。

表3 模糊形式背景
Table 3 Fuzzy formal context

U	M				
	m_1	m_2	m_3	m_4	m_5
u_1	1.0	0.2	0.6	0.0	0.0
u_2	0.8	0.4	0.8	0.6	0.8
u_3	0.6	0.0	0.2	0.0	0.4
u_4	0.6	0.2	0.0	0.6	0.0
u_5	0.0	0.0	0.6	0.0	0.0

设置 $T=0.2$,可以从表3导出一个模糊概念($\{(u_1, 0.2), (u_2, 0.4)\}, \{m_1, m_2, m_3\}$),其中 $(u_1, 0.2)$ 表示

用户 u_1 对 $\{m_1, m_2, m_3\}$ 拥有的程度不低于 0.2。

1.3 模糊形式概念的推荐应用

推荐系统能够为用户进行资源筛选,缓解信息量过载问题,因此得到了广泛关注与发展。目前主流的推荐算法包括基于内容的推荐算法、协同过滤算法、社会化推荐算法和基于深度学习的推荐算法等^[27]。

协同过滤算法是利用具有相同兴趣的群体偏好进行推荐,而形式概念可以看作是通过内涵对用户的聚类,因此协同过滤算法可以与形式概念很好地结合起来。Boucher-Ryan 等^[28]将形式概念分析用于协同过滤推荐系统,利用形式概念描述用户与项目之间关系,通过概念格为用户寻找邻居,从而实现推荐。朵琳等^[29]为解决协同过滤推荐算法中的数据稀疏问题,利用概念格为用户兴趣建模,将近邻用户分为直接“最近邻”和间接“最近邻”两类,通过相似度计算为目标用户进行评分预测。

模糊形式概念分析能够适应模糊环境,利用更多的信息,因此在推荐系统领域也有相关研究。Fang 等^[20]对多值形式背景进行模糊化,利用属性标尺将用户评分分解为 3 个模糊集(不喜欢、中立和喜欢),并据此构建模糊概念格,最后利用格上模糊概念进行协同推荐。张喜征等^[21]构建了带有相似度的模糊概念格,通过概念之间的层次关系,实现了对用户的个性化知识推荐。Mezni 等^[30]用模糊概念格表示云环境,通过从模糊概念格上进行信息提取对用户进行云服务推荐。

目前利用模糊概念进行推荐的研究仍偏少,主要是利用格进行推荐,而格的构建需要生成所有概念复杂度高,不利于应用,因此为降低复杂度,通过生成部分模糊概念,利用概念的内部信息实现用户推荐。

2 问题描述

利用模糊概念集合进行推荐,主要包含 2 个子问题:一是模糊概念集合构造问题;二是基于模糊概念集合的推荐问题。

2.1 模糊概念集合构造

定义 8 用户相似度。给定一个模糊形式背景 $H=(U, M, \tilde{I})$ 以及一个隶属度阈值 T , 令用户 $u_1, u_2 \in U$, 它们之间的相似度定义为

$$D(u_1, u_2) = |\{m \in M \mid \mu(u_1, m) \geq T, \mu(u_2, m) \geq T\}|, \quad (7)$$

其中 $|\cdot|$ 表示集合的基数。

用户相似度描述了用户之间的关系,其直观意义上表示 2 个用户在隶属度阈值 T 下共同拥有项目数量。

定义 9 用户子背景。给定一个模糊形式背景 $H=(U, M, \tilde{I})$ 以及隶属度阈值 T 与相似度阈值 θ , 令 u 为 U 中的一个用户, 则一个关于用户 u 且具有阈值 θ 的子背景为 $H_u=(U_u, M_u, \tilde{I}_u)$ 。其中 $U_u = \{x \in U \mid D(u, x) \geq \theta\}$, $M_u = \{m \in M \mid \mu(u, m) \geq T\}$, $\tilde{I}_u \subseteq \tilde{I}$ 是集合 U_u 与 M_u 之间的模糊关系。

用户子背景是由与当前用户相似度大于等于 θ 的用户集和隶属度大于等于 T 的项目集构成。利用子背景进行模糊概念生成,避免使用较为庞大的原始背景,同时过滤低质量用户,帮助加快模糊概念生成,提高模糊概念的质量。

为了得到内涵与外延平衡的概念,在模糊概念启发式生成时需要设计相关指标对生成概念进行判断。

定义 10 概念大小。设 $H=(U, M, \tilde{I})$ 是一个模糊形式背景, $c=(\varphi(A), B)$ 是其上的一个模糊概念, 则模糊概念 c 定义为

$$S(c) = \min(\alpha * |A|, (1-\alpha) * |B|), \quad (8)$$

其中 $\alpha \in [0, 1]$ 是一个权重因子。

概念大小指标的设计利用了外延与内涵的长度信息,可以有效保证生成的概念外延与内涵数量保持一定平衡,使其包含更多信息,提高后续推荐效果。

问题 1 模糊概念集合构造。

输入:模糊形式背景 (U, M, \tilde{I}) , 隶属度阈值 T 。

输出:模糊概念集合 C 。

约束条件 1: $\forall u \in U, \exists (\varphi(A), B) \in C, \mu_{\varphi(A)}(u) \geq T$ 。

约束条件 2: $\forall (\varphi(A), B) \in C, |A| > 1$ 。

优化目标: $\max(S(c)), c \in C$ 。

其中,隶属度阈值 T 约束了用户与项目之间的关系,需要根据具体背景进行选择。约束条件 1 表示模糊形式背景中的每个用户都应该至少存在于一个模糊概念中,是后续推荐的基础。约束条件 2 表示生成的概念中不能只有一个用户,作用是过滤对推荐无效的概念。优化目标是最大化概念大小,其目的是使希望生成的概念包含更多的信息,提高后续推荐效果。

2.2 基于模糊概念集合的推荐

概念本身实际上可以看作是通过项目对用户的聚类,在同一概念中的用户可以看作具有相同偏好,因此基于协同过滤的思想,可以利用单个概念本身信息实现推荐。

模糊概念外延用户隶属度代表了用户对项目的拥有情况,它们之间的差值可以衡量用户的相似性,因此可以利用隶属度差值设计用户权重来表示不同用户对推荐的影响。

定义 11 用户权重。设 $H=(U, M, \tilde{I})$ 是一个模糊形式背景, $c=(\varphi(A), B)$ 是其上的一个模糊概念, 给定用户 $u_1, u_2 \in A$, 则在概念 c 中用户 u_2 关于用户 u_1 的权重定义为

$$w_c(u_1, u_2) = \frac{1 - |\mu_{\varphi(A)}(u_1) - \mu_{\varphi(A)}(u_2)|}{\sum_{x \in A, x \neq u_1} (1 - |\mu_{\varphi(A)}(u_1) - \mu_{\varphi(A)}(x)|)} \quad (9)$$

对用户进行推荐时,为了判断项目是否给用户推荐,定义了一个融入用户权重的推荐置信度来衡量。

定义 12 推荐置信度。设 $H=(U, M, \tilde{I})$ 是一个模糊形式背景, $c=(\varphi(A), B)$ 是其上的一个模糊概念, 给定用户 $u \in A$ 与项目 $m \in M$, 则在模糊概念 c 上项目 m 关于用户 u 的推荐置信度定义为

$$R_c(u, m) = \frac{\sum_{x \in A, x \neq u} (\lceil \mu(x, m) \rceil * w_c(u, x))}{|A| - 1}, \quad (10)$$

其中 $\lceil \cdot \rceil$ 为向上取整。

推荐置信度定义在单个模糊概念上,其值范围为 $[0, 1]$ 。计算推荐置信度融入了用户权重,使其判定信息更加丰富。

问题 2 基于模糊概念集合的推荐。

输入:模糊形式背景 (U, M, \tilde{I}) , 模糊概念集合 C , 推荐阈值 λ 。

输出:推荐矩阵 L 。

约束条件: $\forall u \in U, \forall m \in M$, 且 $\mu(u, m) = 0, \exists c \in C$, 使 $R_c(u, m) \geq \lambda$, 则向 u 推荐 m 。

优化目标: $\max(F_1)$ 。

其中,输入的模糊概念集合由问题 1 构建。约束条件表示为目标用户进行推荐时,待推项目的推荐置信度达到 λ 才可以推荐。优化目标为最大化 F_1 , 即使推荐效果更优。

3 算法设计

本节设计了模糊概念集构造 (fuzzy concept set construction, FCSC) 算法与基于模糊概念集合的推荐 (recommendation based on fuzzy concept set, RFCS) 算法, 解决问题 1 与问题 2, 并给出了算法描述与伪代码, 分析了算法时间复杂度, 最后给出运行实例。

3.1 模糊概念集构造算法

模糊概念集构造算法先利用用户相似度为每个用户生成子背景, 再调用模糊概念生成 (fuzzy concept generation, FCG) 算法生成模糊概念, 实现模糊概念集合的构造, FCG 算法为单个用户在其子背景下分别以用户和项目为线索进行启发式模糊概念生成。

在每个用户子背景上, 为了让生成概念的信息更丰富, 即使概念大小 $S(c)$ 尽可能大, 采用启发式方式从用户与项目 2 个角度进行模糊概念生成。

设 H_u 为用户 u 的子背景, 从用户角度进行模糊概念生成: 依次选取相似度最大的用户加入用户线索集合 A_u , 即 $A_u = A_u \cup \{\arg \max_{x \in U_u} D(u, x)\}$, 利用用户线索集合生成模糊概念 $c_0 = (\varphi(A_u^{**}), A_u^*)$, 生成的模糊概念

利用定义 10 计算 $S(c_o)$ 。当概念大小不再增大时将 c_o 加入模糊概念集合。

从项目角度生成模糊概念类似:依次选取覆盖用户最多的项目加入到项目线索集合 B_m , 即 $B_m = B_m \cup \{\arg \max_{m \in M_u} |m|^*|\}$, 利用项目线索集合生成模糊概念 $c_a = (\varphi(B_m^*), B_m^{**})$, 同样计算 $S(c_a)$ 。当概念大小不再增大时将 c_a 加入模糊概念集合。

算法 1 FCG 算法。

输入:用户子背景 $H_u = (U_u, M_u, \tilde{I}_u)$, 隶属度阈值 T , 权重因子 α , 用户 u 。

输出:模糊概念集 C_u , 其包含以用户为线索生成的概念 c_o 和以项目为线索生成的概念 c_a 。

1. $A_u \leftarrow \emptyset, c_o \leftarrow (\emptyset, \emptyset)$;
2. $U_t \leftarrow U_u, s \leftarrow 0$;
3. while (true) do
4. $t = \arg \max_{x \in U_t} D(u, x)$;
5. $A_t = A_u \cup \{t\}$;
6. $s_t = S((\varphi(A_t^{**}), A_t^*))$;
7. if ($s_t > s \vee |A_t^{**}| \leq 1$) then
8. $A_u = A_t, s = s_t$;
9. $U_t = U_t - \{t\}$;
10. else
11. $c_o = (\varphi(A_u^{**}), A_u^*)$;
12. break;
13. end if
14. end while
15. $B_m \leftarrow \emptyset, c_a \leftarrow (\emptyset, \emptyset)$;
16. $M_t \leftarrow M_u, s \leftarrow 0$;
17. while (true) do
18. $t = \arg \max_{m \in M_t} |m|^*|$;
19. $B_t = B_m \cup \{t\}$;
20. $s_t = S((\varphi(B_t^*), B_t^{**}))$;
21. if ($s_t > s \vee |B_t^*| \leq 1$) then
22. $B_m = B_t, s = s_t$;
23. $M_t = M_t - \{t\}$;
24. else
25. $c_a = (\varphi(B_m^*), B_m^{**})$;
26. break;
27. end if
28. end while
29. return $C_u = \{c_o, c_a\}$;

算法 1 在用户 u 子背景上分别以用户和项目为线索,利用启发式方法进行模糊概念生成。第 1—14 行代码是在用户子背景 H_u 上以用户为线索为用户 u 进行模糊概念生成。其主要思路是逐一将与 u 相似度最大的用户添加到用户线索集合 A_u , 通过 A_u 来计算概念大小,判别是否满足条件,最终得到以用户为线索生成的模糊概念。第 15—28 行代码则是以项目为线索进行模糊概念生成,与以用户为线索进行模糊概念生成类似,其中项目线索集合 B_m 中逐一添加的是当前覆盖用户最多的项目。

算法 2 FCSC 算法。

输入:模糊形式背景 $H = (U, M, \tilde{I})$, 隶属度阈值 T , 相似度阈值 θ , 权重因子 α 。

输出:模糊概念集合 C 。

```

1.  $C \leftarrow \emptyset$ ;
2. for (each  $u \in U$ ) do
3.    $U_u \leftarrow \emptyset, M_u \leftarrow \emptyset$ ;
4.   for (each  $o \in U$ ) do
5.     if ( $D(u, o) \geq \theta$ ) then
6.        $U_u = U_u \cup \{o\}$ ;
7.     end if
8.   end for
9.   for (each  $m \in M$ ) do
10.    if ( $\mu(u, m) \geq T$ ) then
11.       $M_u = M_u \cup \{m\}$ ;
12.    end if
13.  end for
14.   $H_u = (U_u, M_u, \tilde{I}_u)$ ;
15.   $C_u = \text{FCG}(H_u, T, \alpha, u)$ ;
16.   $C = C \cup C_u$ ;
17. end for
18. return  $C$ ;

```

算法 2 在模糊形式背景上进行模糊概念集合生成。通过遍历模糊形式背景中每个用户,为其生成子背景,再利用算法 1 对每个用户进行模糊概念的生成,最终合并得到模糊概念集合。第 3—14 行代码为用户 u 提取子背景,子背景中的用户与用户 u 相似度不低于 θ ,项目则为用户 u 在阈值 T 下所拥有的项目。第 15—16 行代码调用算法 1 在子背景上为用户 u 生成模糊概念,再加入到模糊概念集合中。

3.2 基于模糊概念集合的推荐算法

RFCS 算法是基于算法 2 所生成的模糊概念集合。通过计算推荐置信度为每个用户进行项目推荐。利用模糊概念集合对用户的推荐可以形式化描述为:

设 U 为用户集, M 为项目集,形成的模糊形式背景为 $H = (U, M, \tilde{I})$, C 为从背景 H 生成的模糊概念集合, $R_c(u, m)$ 是在概念 c 中项目 m 对用户 u 的推荐置信度, λ 为推荐阈值,则对用户 $u \in U$ 推荐的结果是使 $R_c(u, m) \geq \lambda$ 且 $\mu(u, m) = 0$ 的项目 s , 即 $s = \{m \in M \mid \exists c \in C, R_c(u, m) \geq \lambda \wedge \mu(u, m) = 0\}$ 。

算法 3 RFCS 算法。

输入:模糊形式背景 $H = (U, M, \tilde{I})$, 模糊概念集合 C , 隶属度阈值 T , 推荐阈值 λ 。

输出:推荐矩阵 L 。

```

1.  $L_{|U| \times |M|} \leftarrow 0$ ;
2. for (each  $u \in U$ ) do
3.    $C_t = \{c \in C \mid \mu_{\varphi(A)}(u) \geq T\}$ ;
4.   for (each  $m \in M$ ) do
5.     if ( $\mu(u, m) = 0 \wedge R_{c \in C_t}(u, m) \geq \lambda$ ) then
6.        $L_{(u, m)} = 1$ ;
7.     end if
8.   end for
9. end for
10. return  $L$ 

```

算法 3 基于模糊概念集合为每一个用户推荐其所未拥有的项目。第 3 行是为用户 u 选择隶属度大于 T 的模糊概念。第 4—8 行基于模糊概念为用户 u 进行项目 m 推荐,根据推荐置信度是否满足阈值 λ ,确定是否推荐,最后更新至推荐矩阵。

3.3 算法时间复杂度分析

接下来进行算法1—3的时间复杂度分析。假设形式背景大小为 $n \times m$, 其中 n 为用户数。算法1中以用户为线索生成模糊概念最坏情况需要比较 n 次, 因此时间复杂度为 $O(n^2m)$, 以项目为线索生成模糊概念最坏情况需要比较 m 次, 时间复杂度为 $O(nm^2)$ 。算法2中用户子背景生成主要计算用户相似度时间复杂度为 $O(n^2m)$, 然后需要利用算法1对形式背景中的每个用户生成模糊概念, 因此时间复杂度为 $O(n^3m + n^2m^2)$ 。算法3在进行推荐时, 需要遍历每个用户与项目, 最坏情况时需要遍历每个模糊概念及其外延用户, 模糊概念个数最大为 $2n$, 其外延长度为 n , 因此算法3时间复杂度为 $O(n^3m)$ 。

模糊概念格通过先生成经典概念格再加入模糊集运算而得到。经典的概念格生成算法需要生成所有概念复杂度较高, 如 Next Neighbor 算法时间复杂度为 $O(nm^2|C|)$, Object Intersection 算法时间复杂度为 $O(nm|C|)$, Ganter Next Closure 算法时间复杂度为 $O(n^2m|C|)$ ^[31]。这里 $|C|$ 表示概念的数量, 在极端情况下会有 $2^{\min(n,m)}$ 个概念。本方法由于只生成部分概念, 相较于其他生成格的算法具有更低的时间复杂度。

3.4 运行实例

根据表3的模糊形式背景, 以用户 u_1 为例, 描述模糊概念的生成及推荐过程, 其中 $T=0.2, \theta=2, \alpha=0.3, \lambda=0.7$ 。

首先为用户 u_1 生成模糊概念。计算 u_1 与用户 u_2, u_3, u_4, u_5 的相似度分别为 3、2、2、1, 再根据 θ 提取子背景 H_{u_1} 见表4。接下来在 H_{u_1} 上分别以用户和项目为线索进行模糊概念生成, 过程如见表5、6。以用户为线索进行概念生成, 依次根据相似度加入到用户线索集合中, 当加入到 u_4 时, 概念减小, 于是停止计算, 得到模糊概念 $c_1 = \{ \{ (u_1, 0.6), (u_2, 0.8), (u_3, 0.2) \}, \{ m_1, m_3 \} \}$ 。以项目为线索进行概念生成方法类似, 得到模糊概念 $c_2 = \{ \{ (u_1, 0.2), (u_2, 0.4), (u_4, 0.2) \}, \{ m_1, m_2 \} \}$ 。

表4 u_1 的子背景
Table 4 The sub-background of u_1

U_{u_1}	M_{u_1}		
	m_1	m_2	m_3
u_1	1.0	0.2	0.6
u_2	0.8	0.4	0.8
u_3	0.6	0.0	0.2
u_4	0.6	0.2	0.0

表5 在 u_1 子背景下以用户为线索生成概念过程
Table 5 The process of concept generation by taking user as the clue under H_{u_1}

A_u	$\varphi(A_u^{**})$	A_u^*	$S(c)$
$\{u_1\}$	$\{(u_1, 0.2), (u_2, 0.4)\}$	$\{m_1, m_2, m_3\}$	0.6
$\{u_1, u_2\}$	$\{(u_1, 0.2), (u_2, 0.4)\}$	$\{m_1, m_2, m_3\}$	0.6
$\{u_1, u_2, u_3\}$	$\{(u_1, 0.6), (u_2, 0.8), (u_3, 0.2)\}$	$\{m_1, m_3\}$	0.9
$\{u_1, u_2, u_3, u_4\}$	$\{(u_1, 1.0), (u_2, 0.8), (u_3, 0.6), (u_4, 0.6)\}$	$\{m_1\}$	0.7

表6 在 u_1 子背景下以项目为线索生成概念过程
Table 6 The process of concept generation by taking item as the clue under H_{u_1}

B_u	$\varphi(B_u^*)$	B_u^{**}	$S(c)$
$\{m_1\}$	$\{(u_1, 1.0), (u_2, 0.8), (u_3, 0.6), (u_4, 0.6)\}$	$\{m_1\}$	0.7
$\{m_1, m_2\}$	$\{(u_1, 0.2), (u_2, 0.4), (u_4, 0.2)\}$	$\{m_1, m_2\}$	0.9
$\{m_1, m_2, m_3\}$	$\{(u_1, 0.2), (u_2, 0.4)\}$	$\{m_1, m_2, m_3\}$	0.6

基于模糊概念 c_1 与 c_2 对用户 u_1 进行推荐, u_1 的待推荐项目为 $\{m_4, m_5\}$, 在 c_1 中根据定义11可求得 u_2 与 u_3 的权重分别为 0.57 与 0.43, 根据定义12可求得 $R_{c_1}(u_1, m_4) = 0.57, R_{c_1}(u_1, m_5) = 1$ 。同样地, 可以得到 $R_{c_2}(u_1, m_4) = 1, R_{c_2}(u_1, m_5) = 0.44$ 。由于推荐阈值 $\lambda = 0.7$, 因此模糊概念 c_1 和 c_2 , 向用户 u_1 推荐项目 m_4 与 m_5 。

4 实验与分析

4.1 数据集

为了验证算法的有效性, 试验采用6个真实数据集, 其中 movielens-100k 和 movielens-1m 数据集^[32]由 GroupLens 项目组收集, 分别包含了10万和100万条电影评分记录; eachmovie 数据集^[33]由 DEC 研究中心收集, 共包含72916个用户对1628部电影的2811983条评分记录, 随机选择了2000、3000个用户对1628部电影的评分, 分别记作 eachmovie-2ku 与 eachmovie-3ku; filmtrust 数据集^[34]是从 FilmTrust 网站抓取的数

数据集,共包含1508个用户对2071部电影的评分;jester数据集^[35]是从Jester Online Joke Recommender System抓取的笑话评分数据集,包含了73421个用户对100个笑话的评分情况,选择了8000个用户的评分数据,记为jester-s。数据集具体信息如表7所示。

表7 试验数据集
Table 7 Experimental datasets

数据集	用户数	项目数	评分数	稠密度
movielens-100k	943	1682	100000	0.0630
movielens-1m	6040	3952	1000000	0.0419
eachmovie-2ku	2000	1628	38571	0.0118
eachmovie-3ku	3000	1628	59247	0.0121
filmtrust	1508	2071	35497	0.0113
jester-s	8000	100	577379	0.7217

由于后续处理的基础是模糊形式背景,因此需要将数据集进行归一化处理,采用线性归一化方法,将评分值归一化至 $[0,1]$ 。

4.2 评价指标

对推荐质量的评估采用推荐系统中常用的3个指标:准确率 $P_{\text{precision}}$ 、召回率 P_{recall} 与 F_1 。假设 $L(U)$ 为对所有用户的推荐矩阵, $E(U)$ 为验证结果的测试矩阵。

准确率表示的是推荐正确的个数占总推荐数的比例:

$$P_{\text{precision}} = \frac{|L(U) \cap E(U)|}{|L(U)|} \quad (11)$$

召回率表示的是推荐正确的个数占测试集总项目的比例:

$$P_{\text{recall}} = \frac{|L(U) \cap E(U)|}{|E(U)|} \quad (12)$$

F_1 是综合准确率与召回率的评价指标:

$$F_1 = \frac{2P_{\text{precision}}P_{\text{recall}}}{P_{\text{precision}} + P_{\text{recall}}} \quad (13)$$

4.3 参数分析

算法中有3个比较重要的参数,即相似度阈值 θ 、权重因子 α 和推荐阈值 λ 。为了分析它们对算法的影响,现从movielens-100k数据集上随机抽样4组大小为 200×420 的数据,记为data1-data4,并按照8:2进行训练集与测试集划分。

4.3.1 相似度阈值 θ 的影响

θ 决定了子背景中的用户个数,影响子背景的大小,因此设计实验测试 θ 对概念生成时间以及推荐结果的影响。设置 θ 范围为 $[0,40]$,步长为2, $T=0.2$, $\alpha=0.2$, $\lambda=0.7$ 。如图1(a)所示,在不同数据集上,概念的生成时间与 θ 总体呈负相关, θ 增大可以有效缩短概念生成时间。如图1(b)所示,随着 θ 的增大, F_1 先增大再减小,当 $\theta=10$ 时,具有较短的生成时间与较好的推荐效果,因此设置合理的 θ ,可以缩短概念生成时间并提高推荐效果。

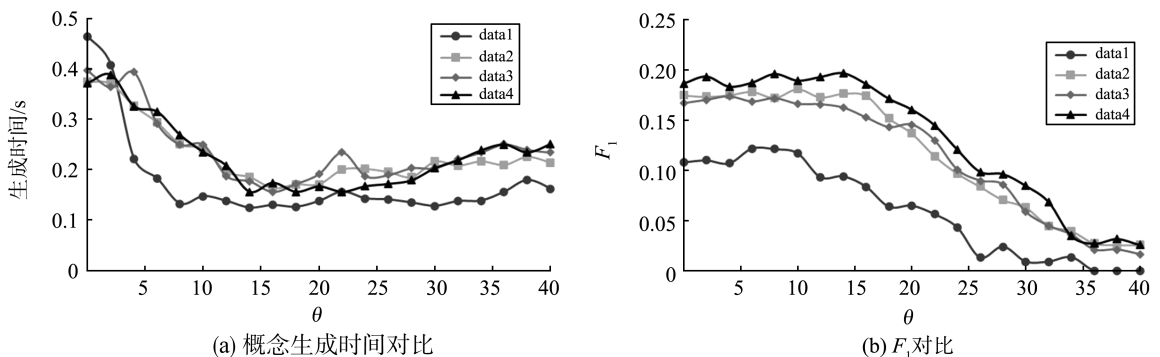


图1 不同 θ 对概念生成时间与 F_1 的影响

Fig.1 Influence of different θ on concept generation time and F_1

4.3.2 权重因子 α 的影响

α 是概念大小中的权重因子,它可以表示概念外延与内涵的重要程度。为了测试其对推荐结果的影响,在4个测试数据集上,设置 $\alpha \in [0, 1]$,步长为 0.05, $T=0.2$, $\theta=1$, $\lambda=0.7$,实验结果如图2所示。随着 α 的增大,4个数据集上的 F_1 均是先增大后减小。在后续的实验,为不同的数据集选择了最优的 α ,以提高推荐效果。

4.3.3 推荐阈值 λ 的影响

由于 λ 决定了向用户推荐哪些项目,因此设计实验测试 λ 对推荐结果的影响。设置 $T=0.2$, $\alpha=0.3$, $\theta=1$, $\lambda \in [0, 1]$,步长为 0.05,在4个测试数据集上进行实验,实验结果如图3所示。随着推荐阈值 λ 增大,在4个测试数据集上, F_1 均先增大后减小,在 α 为 0.6 左右达到最大值,因此设置合理的 λ ,可以有效提升推荐效果。

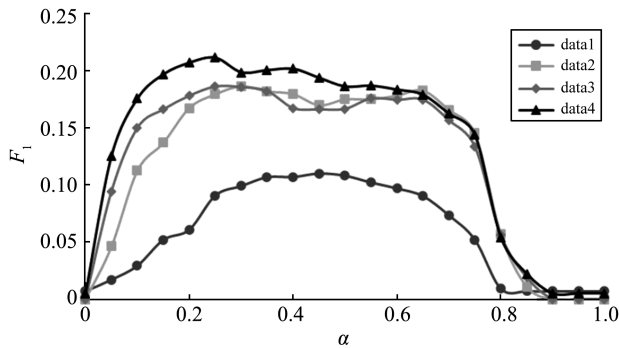


图2 不同 α 对 F_1 的影响

Fig.2 Influence of different α on F_1

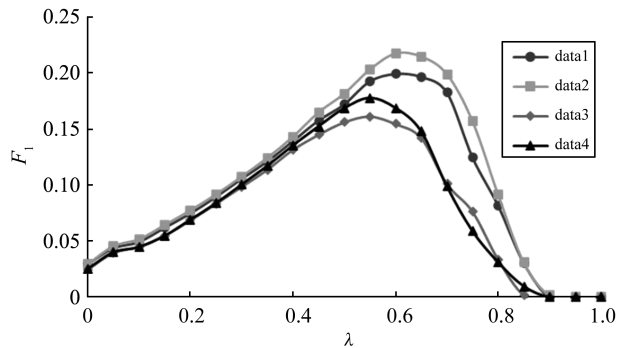


图3 不同 λ 对 F_1 的影响

Fig.3 Influence of different λ on F_1

4.4 算法对比

为了验证提出的算法,选择了推荐领域经典的 k 最近邻 (k -nearest neighbor, kNN) 与基于项目的协同过滤 (item-based collaborative filtering, IBCF) 算法、基于概念集合的推荐 (concept set based recommendation, CSBR)^[22] 算法、利用形式概念分析进行矩阵分解并结合 kNN 的 GreConD-kNN^[36] 算法以及基于神经网络的协同过滤推荐 (neural network-based collaborative filtering, NCF)^[37] 算法进行对比。

kNN 算法是基于用户的协同过滤算法,主要通过计算用户之间的相似度,为目标用户寻找 k 个相似用户,再利用相似用户的项目偏好进行推荐。本文中选择了 Jaccard 距离进行用户相似度的计算。

IBCF 算法是基于项目的协同过滤算法,通过计算项目之间的相似度,根据待推荐用户的历史项目偏好,将类似项目推荐给用户。本文中在计算项目相似度时也采用了 Jaccard 距离。

CSBR 算法是基于概念集合的推荐算法,利用概念面积作为启发式信息,通过不断添加项目为用户生成概念面积最大的强概念,之后根据概念中邻居用户的项目偏好进行推荐。

GreConD-kNN 是结合形式概念分析与 kNN 的推荐算法,利用形式概念分析进行矩阵分解,将分解的用户因子矩阵作为 kNN 的输入,对用户进行协同推荐。

NCF 算法是基于神经网络的协同过滤算法,通过多层感知机学习用户与物品的交互函数,利用神经网络代替矩阵分解中的内积运算,实现协同过滤推荐。

在实验中各数据集按照 8:2 的比例进行训练集与测试集划分,并随机重复 10 次取平均得到各实验指标。在 RFCS 算法中, $\theta \in [1, 20]$, $\alpha \in [0.2, 0.3]$, $\lambda \in [0.6, 0.8]$ 。

表8展示了 RFCS 算法与其他算法在不同数据集上的实验结果,在准确率方面,RFCS 算法在大多实验数据集上都优于其他算法;在召回率方面,RFCS 算法不占优,要低于 kNN、IBCF 与 NCF 算法,但在大部分数据集上要优于 CSBR 算法;在 F_1 方面,RFCS 算法在 eachmovie-2ku、eachmovie-3ku 与 filmtrust 上要优于其他算法,在 movielens-100k 与 jester-s 上要高于 CSBR 算法与 GreConD-kNN 算法相当。

从实验结果可以看出,RFCS 算法在稀疏的数据集上比其他算法效果更好。eachmovie-2ku、eachmovie-3ku 与 filmtrust 数据集比其他3个数据集更为稀疏,RFCS 算法在3个数据集上的推荐效果要优于其他推荐算法。jester-s 数据集稠密度最高,RFCS 算法效果与其他推荐算法相比不占优势,说明 RFCS 算法更适用于稀

疏度高的数据集。

表8 RFCS与其他算法推荐效果对比
Table 8 Comparison of RFCS recommendation effect with other algorithms

数据集	算法	$P_{\text{precision}}$	P_{recall}	F_1
movielens-100k	RFCS	0.248 7	0.243 9	0.246 2
	CSBR	0.192 5	0.239 3	0.213 2
	kNN	0.199 4	0.344 2	0.252 5
	IBCF	0.203 8	0.406 1	0.271 4
	GreConD-kNN	0.199 6	0.342 4	0.252 1
	NCF	0.194 6	0.335 2	0.246 1
movielens-1m	RFCS	0.216 0	0.186 4	0.200 0
	CSBR	0.142 1	0.204 0	0.167 6
	kNN	0.188 3	0.339 0	0.242 0
	IBCF	0.200 1	0.394 2	0.265 5
	GreConD-kNN	0.188 5	0.338 0	0.242 1
	NCF	0.139 5	0.359 3	0.200 8
eachmovie-2ku	RFCS	0.223 6	0.251 8	0.236 8
	CSBR	0.187 5	0.227 2	0.205 2
	kNN	0.201 4	0.272 9	0.231 8
	IBCF	0.148 6	0.352 0	0.208 9
	GreConD-kNN	0.199 0	0.269 8	0.229 1
	NCF	0.129 7	0.317 8	0.184 1
eachmovie-3ku	RFCS	0.231 5	0.271 3	0.249 7
	CSBR	0.203 0	0.236 2	0.218 3
	kNN	0.112 3	0.276 8	0.159 6
	IBCF	0.151 0	0.389 4	0.217 7
	GreConD-kNN	0.104 3	0.247 8	0.146 6
	NCF	0.131 9	0.363 2	0.193 4
filmtrust	RFCS	0.486 2	0.477 6	0.481 4
	CSBR	0.360 2	0.641 7	0.461 2
	kNN	0.438 6	0.441 8	0.441 0
	IBCF	0.182 6	0.532 8	0.271 7
	GreConD-kNN	0.467 9	0.406 6	0.435 0
	NCF	0.207 9	0.742 2	0.324 7
jester-s	RFCS	0.714 8	0.846 9	0.775 3
	CSBR	0.572 7	0.839 7	0.681 0
	kNN	0.880 3	0.902 9	0.891 5
	IBCF	0.891 0	0.851 7	0.870 9
	GreConD-kNN	0.895 2	0.614 0	0.728 4
	NCF	0.971 0	0.853 4	0.908 4

5 结语

本文提出了一种模糊概念集合的启发式构建方法,并将生成的模糊概念集合用于推荐。在用户子背景上将概念大小作为启发式信息,以用户与项目为线索生成模糊概念。将外延用户的隶属度融入了推荐置信度的计算,有效提升了推荐效率。模糊概念相较于传统概念具有更丰富的信息,因此具有广阔的发展空间,本文提出的算法拓宽了其在推荐领域中的应用。由于现实中动态数据较多,因此进一步将考虑在动态环境

下研究模糊概念的生成与推荐,同时设计并行算法,进一步提升推荐效率。

参考文献:

- [1] WILLE R. Restructuring lattice theory: an approach based on hierarchies of concepts[C]//Formal Concept Analysis; 7th International Conference. Berlin: Springer, 2009:314-339.
- [2] KUZNETSOV S O. Machine learning and formal concept analysis[C]//Concept Lattices; Second International Conference on Formal Concept Analysis. Berlin: Springer, 2004:287-312.
- [3] KANG Xiangping, LI Deyu, WANG Suge. A multi-instance ensemble learning model based on concept lattice[J]. Knowledge-Based Systems, 2011, 24(8):1203-1213.
- [4] KUMAR C A. Fuzzy clustering-based formal concept analysis for association rules mining[J]. Applied Artificial Intelligence, 2012, 26(3):274-301.
- [5] TALLAM S, GUPTA N. A concept analysis inspired greedy algorithm for test suite minimization[J]. ACM SIGSOFT Software Engineering Notes, 2005, 31(1):35-42.
- [6] ZOU Caifeng, ZHANG Daqiang, WAN Jiafu, et al. Using concept lattice for personalized recommendation system design[J]. IEEE Systems Journal, 2015, 11(1):305-314.
- [7] 陈昊文,王黎明,张卓. 基于概念邻域的 Top-N 推荐算法[J]. 小型微型计算机系统, 2017, 38(11):2553-2559.
CHEN Haowen, WANG Liming, ZHANG Zhuo. Top-N recommendation algorithm based on conceptual neighborhood[J]. Journal of Chinese Computer Systems, 2017, 38(11):2553-2559.
- [8] ZADEH L A. Fuzzy sets[J]. Information and Control, 1965, 8(3):338-353.
- [9] QUAN T T, HUI S C, CAO T H. A fuzzy FCA-based approach for citation-based document retrieval[C]//IEEE Conference on Cybernetics and Intelligent Systems, 2004. Piscataway: IEEE, 2004, 1:578-583.
- [10] BURUSCO J A, FUENTES-GONZÁLEZ R. The study of the L-fuzzy concept lattice[J]. Mathware and Soft Computing, 1994, 1(3):209-218.
- [11] 刘宗田,强宇,周文,等. 一种模糊概念格模型及其渐进式构造算法[J]. 计算机学报, 2007, 30(2):184-188.
LIU Zongtian, QIANG Yu, ZHOU Wen, et al. Fuzzy concept lattice model and its incremental construction algorithm[J]. Chinese Journal of Computers, 2007, 30(2):184-188.
- [12] KRAJCI S. Cluster based efficient generation of fuzzy concepts[J]. Neural Network World, 2003, 13(5):521-530.
- [13] YAHIA S B, AROUR K, SLIMANI A, et al. Discovery of compact rules in relational databases[J]. Information Science Journal, 2000, 4(3):497-511.
- [14] ZHANG Wenxiu, MA Jianmin, FAN Shiqing. Variable threshold concept lattices[J]. Information Sciences, 2007, 177(22):4883-4892.
- [15] BOFFA S, DE MAIO C, DI NOLA A, et al. Unifying fuzzy concept lattice construction methods[C]//2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). Piscataway: IEEE, 2016: 209-216.
- [16] SHAO Mingwen, LEUNG Y, WANG Xizhao, et al. Granular reducts of formal fuzzy contexts[J]. Knowledge-Based Systems, 2016, 114:156-166.
- [17] MI Yunlong, SHI Yong, LI Jinhai, et al. Fuzzy-based concept learning method: exploiting data with fuzzy conceptual clustering[J]. IEEE Transactions on Cybernetics, 2020, 52(1):582-593.
- [18] XU Weihua, GUO Doudou, QIAN Yuhua, et al. Two-way concept-cognitive learning method: a fuzzy-based progressive learning[J]. IEEE Transactions on Fuzzy Systems, 2022, 31(6):1-15.
- [19] SINGH P K, KUMAR C A, LI Jinhai. Knowledge representation using interval-valued fuzzy formal concept lattice[J]. Soft Computing, 2016, 20:1485-1502.
- [20] FANG Peici, ZHENG Siyao. A research on fuzzy formal concept analysis based collaborative filtering recommendation system [C]//2009 Second International Symposium on Knowledge Acquisition and Modeling. Piscataway: IEEE, 2009: 352-355.
- [21] 张喜征,蔡月月,罗文. 基于模糊概念格的领先用户个性化知识推荐研究[J]. 科技管理研究, 2019, 39(7):183-189.
ZHANG Xizheng, CAI Yueyue, LUO Wen. Research on personalized knowledge recommendation for leading users based on fuzzy concept lattice in innovation community[J]. Science and Technology Management Research, 2019, 39(7):183-189.
- [22] LIU Zhonghui, ZHAO Qi, ZOU Lu, et al. A heuristic concept construction approach to collaborative recommendation[J]. International Journal of Approximate Reasoning, 2022, 146:119-132.
- [23] GANTER B, WILLE R. Formal concept analysis: mathematical foundations[M]. Berlin: Springer, 1999.

- [24] ZADEH L A. Fuzzy sets and information granularity[M]. Amsterdam: World Scientific Publisher, 1979.
- [25] 李金海, 魏玲, 张卓, 等. 概念格理论与方法及其研究展望[J]. 模式识别与人工智能, 2020, 33(7):619-642.
LI Jinhai, WEI Ling, ZHANG Zhuo, et al. Concept lattice theory and method and their research prospect[J]. Pattern Recognition and Artificial Intelligence, 2020, 33(7):619-642.
- [26] POELMANS J, IGNATOV D I, KUZNETSOV S O, et al. Fuzzy and rough formal concept analysis: a survey[J]. International Journal of General Systems, 2014, 43(2):105-134.
- [27] 于蒙, 何文涛, 周绪川, 等. 推荐系统综述[J]. 计算机应用, 2022, 42(6):1898-1913.
YU Meng, HE Wentao, ZHOU Xuchuan, et al. Review of recommendation system[J]. Journal of Computer Applications, 2022, 42(6):1898-1913.
- [28] BOUCHER-RYAN P D, BRIDGE D. Collaborative recommending using formal concept analysis[J]. Knowledge-Based Systems, 2006, 19(5):309-315.
- [29] 朵琳, 杨丙. 一种基于用户兴趣概念格的推荐评分预测方法[J]. 小型微型计算机系统, 2020, 41(10):2104-2108.
DUO Lin, YANG Bing. Recommendation rating prediction based on user interest concept lattice[J]. Journal of Chinese Computer Systems, 2020, 41(10):2104-2108.
- [30] MEZNI H, ABDELJAOUED T. A cloud services recommendation system based on fuzzy formal concept analysis[J]. Data & Knowledge Engineering, 2018, 116:100-123.
- [31] KUMAR C A, SINGH P K. Knowledge representation using formal concept analysis: a study on concept generation[M]. Hershey: IGI Global, 2014: 306-336.
- [32] HARPER F M, KONSTAN J A. The movielens datasets: history and context[J]. ACM Transactions on Interactive Intelligent Systems, 2015, 5(4):1-19.
- [33] PHUONG N D, PHUONG T M. Collaborative filtering by multi-task learning[C]//2008 IEEE International Conference on Research, Innovation and Vision for the Future in Computing and Communication Technologies. Piscataway: IEEE, 2008: 227-232.
- [34] GUO Guibing, ZHANG Jie, YORKE-SMITH N. A novel Bayesian similarity measure for recommender systems[C]//Proceedings of the Twenty-third International Joint Conference on Artificial Intelligence. Menlo Park: AAAI Press, 2013: 2619-2625.
- [35] GOLDBERG K, ROEDER T, GUPTA D, et al. Eigentaste: a constant time collaborative filtering algorithm[J]. Information Retrieval, 2001, 4:133-151.
- [36] IGNATOV D I, NENOVA E, KONSTANTINOVA N, et al. Boolean matrix factorisation for collaborative filtering: an FCA-based approach[C]//International Conference on Artificial Intelligence: Methodology, Systems, and Applications. Cham: Springer, 2014: 47-58.
- [37] HE Xiangnan, LIAO Lizi, ZHANG Hanwang, et al. Neural collaborative filtering[C]//Proceedings of the 26th International Conference on World Wide Web. Republic and Canton of Geneva: International World Wide Web Conferences Steering Committee, 2017: 173-182.

(编辑:陈丽萍)