基于上下文的餐饮推荐算法

唐东平,方民俊,吴邵宇
(华南理工大学,广东 广州 510640)

摘 要:互联网的发展,为餐饮用户的上下文信息获取提供了基础。在用户选择适合其餐饮模式的前提下,加入动态上下文因素以满足用户的需求。为改善传统的协同过滤方法应用于餐饮O2O推荐存在的稀疏矩阵、冷启动等问题,设计了基于上下文后过滤的协同过滤推荐方法。先通过转化基于项目属性效用的评分矩阵,计算出用户对项目评分的偏好相似度。根据用户的评分偏好和静态上下文信息构建相似组,结合上下文信息加权的贝叶斯模型,采用基于KL散度的加权方法进行动态偏好分析,解决上下文信息缺乏时难以构建概率模型以及推荐系统的用户冷启动问题。实验结果显示,随着邻居数目增加时,基于上下文的推荐算法与传统的协同过滤算法相比,能维持较高的准确率和召回率,验证了推荐算法的有效性。

关键词:餐饮;协同过滤;上下文后过滤;贝叶斯模型;KL散度
中图分类号:TP391 **文献标识码**:A **文章编号**:1673-629X(2021)04-0014-07
doi:10.3969/j.issn.1673-629X.2021.04.003

Algorithm on Catering Recommendation Based on Context

TANG Dong-ping,FANG Min-jun,WU Shao-yu
(South China University of Technology,Guangzhou 510640,China)

Abstract:The development of the Internet provides the basis for the acquisition of context information for users of catering. Under the premise that users choose catering modes which are appropriate,dynamic context factors are added to satisfy users' demands. To improve the property of traditional collaborative filtering algorithm which involves defects such as sparse matrix and cold-start,the catering O2O recommendation algorithm is designed based on post-context filtering. First,by converting the rating matrix based on the utility of the items' attribute,the users' preference similarity to the items rating are calculated. Based on the users' rating preference and static context,a similar group is constructed which is combined with the weighted context of Bayesian model. Then the dynamic preference analysis is performed utilizing the weighted method based on KL divergence. The research successfully figured out problems of users' cold-start and modeling probability when context is scarce. The experiment shows that as the number of neighbors increases,the context-based recommendation algorithm maintains a higher accuracy and recall rate than the traditional collaborative filtering algorithm,verifying the effectiveness of the recommendation algorithm.

Key words:catering;collaborative filtering;post-context filtering;Bayesian model;KL divergenc

0 引 言

“民以食为天”,随着互联网技术的推广,餐饮行业成功地实现了从线上到线上的转型,“互联网+餐饮”作为现阶段餐饮领域的热门应用在各种各样的场景,人们不仅可以利用此方式团购美食和订购外卖,而且可以从中找到适合自己的餐饮以及通过美食形成交友圈[1]。

餐饮O2O虽然提高了人们的生活效率,但人们经常无法正确选择符合其自身的食物。第一,面对种类繁多的食物,加上缺乏一定的营养健康理论基础,人们很难准确地判断出有利于其健康的食物[2];第二,人们比较容易受到外界条件的干扰从而做出饮食的选择,例如受到其他消费者的评分、天气、地理位置等因素的影响[3]。综上原因,为用户推荐出适合的餐饮很有必要,既能改善用户的餐饮O2O体验,又能推动餐饮O2O的发展,使其更加智能化、人性化。

在此背景下,该文从用户的心理认知出发,当系统把餐饮推荐给时用户时,用户的心理认知直接决定了

收稿日期:2020-05-27 **修回日期**:2020-09-28
基金项目:国家自然科学基金面上项目(71771091)
作者简介:唐东平(1965-),男,副教授,硕导,研究方向为项目管理、ERP、电子商务;方民俊(1997-),男,硕士研究生,研究方向为信息管理系统、电子商务。

该推荐信息是否有效、是否确切地实现个性化推荐。个性化推荐质量的提高,能有效地提高用户的使用频率,增加用户对推荐系统的信任程度。个性化推荐可定义为,A 用户在网站留下如购买、评价等行为数据,假如 B 用户与 A 用户具有相似的历史行为数据,则 B 用户为相似用户,并且可以将 A 用户具备而 B 用户不具备的爱好推荐给 B 用户。运用此原理,当目标用户打开餐饮网站或软件时,可以迅速推荐可能适合他们的食物[3]。

1　上下文后过滤推荐算法

基于协同过滤的推荐方法是个性化推荐方法的一种思路,由于互联网的餐饮推荐是动态的,因此该文加入上下文感知因素,研究基于上下文感知的协同过滤推荐算法。根据 Dey 对上下文感知的定义,用户和环境等上下文够为用户反馈出符合当前上下文的信息[4]。该文采用上下文后过滤模式在忽略上下文信息的前提下,先运用传统方法进行推荐,再匹配符合用户当前上下文的数据从而实现推荐。

该文区分项目资源的选择偏好和评分偏好,认为上下文信息与评分偏好之间的关系是松耦合的关系,即用户对项目的喜欢程度很大程度上影响着其对项目评分的高低,然而环境、地点等上下文信息的记录反映出了用户会在何种信息对该项目发出请求。换言之,在上下文信息的综合影响下,用户项目选择的感知不清晰,而上下文对项目选择的影响通常是根据概率的,即对于某种类型或具有某种属性的项目在不同的上下文条件下有相应的概率被触发。

基于上下文的推荐算法,一般需要结合基于用户或物品的协同过滤推荐算法进行改进[5]。根据传统的协同过滤算法,通常是依用户或物品之间的欧氏距离定义邻居集,进而在邻居数据中实现推荐过程。在传统的协同过滤算法思想的基础上,该文改进的基于上下文感知的协同过滤算法的主要流程如图 1 所示。

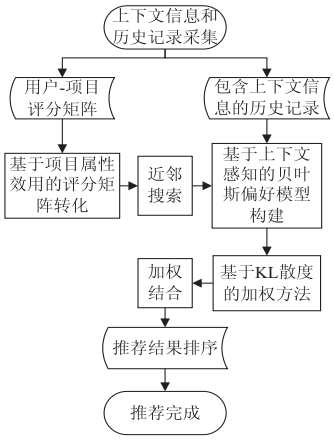

图 1　基于上下文的协同过滤算法流程

2　基于属性的近邻搜索

一般地,基于内容的推荐算法仅是简单地直接根据用户对项目的评分推算目标用户感兴趣的项目或目标用户的近邻用户并向目标用户推荐项目,这种通过评分信息把项目推荐给目标用户的方式可能会因为疏忽了用户偏好的具体细节而影响推荐的效果[6]。这类研究通常是基于分类的思想来挖掘用户对资源类别的粗粒度兴趣偏好,因此难以有效地反映用户对项目资源多个属性特征的偏好情况。若进一步划分和挖掘用户的粗粒度兴趣偏好,则能够更加准确地理解用户对项目资源的需求。朱磊等人研究了基于评分偏好和项目属性的协同过滤算法,物品的相似度公式改进后融合时间参数,结果表明,改进算法的准确率和召回率在 MovieLens－100K 数据集上提高了 9% ~ 27%,在 MovieLens－Latest－Small 数据集上提高了 16% ~ 28%[7]。该文认为,用户在不同项目资源间进行选择时主要从自身较重视的几个项目属性配置情况进行评估,对应地在项目使用结束后的评分也主要反映了用户对项目属性的评价。

该文提出将传统的用户–项目评分矩阵与项目属性结合以推算用户对项目属性的偏好信息,并基于用户对项目属性的偏好信息计算用户评分偏好的相似度,结合静态上下文信息确定协同过滤推荐近邻搜索。

2.1　评分矩阵的基本数据结构

在基于项目属性的推荐系统中要以用户–项目属性值评分矩阵为基础,在不加大用户反馈的工作量和能够共享传统推荐系统中用户信息数据的前提下,需要根据用户对项目资源的评分计算用户对项目属性值的评分信息进而得到用户–项目属性值评分矩阵。用户–项目属性值评分矩阵与用户–项目评分矩阵具有相同的结构,因此借助传统推荐系统中近邻查找算法或用户偏好模型建立方法以及目标用户对项目偏好的预测方法预测用户对项目资源的偏好程度,进而完成推荐过程。

在基于项目属性的推荐系统中,评分矩阵中的元素主要是用户对项目资源的显性评分,某用户对全体项目的评分构成单一的评分矢量,扩展到全体用户时则构成用户–项目评分矩阵。系统中包含 n 个用户和 m 个项目资源可表示的评分矩阵结构如表 1 所示。其中 r_{ij} 表示第 i 个用户对第 j 个项目资源的评分数值,可以用 1 或 0 分别表示喜欢和不喜欢,也可以选择具体的数值范围如 1 ~ 5 分、1 ~ 7 分、1 ~ 10 分当中某值表示用户对项目资源的喜好程度。矩阵的空缺元素表示用户没有对该项目评分,协同过滤推荐基于评分计算的相似用户或项目,预测这些空缺元素的评分值,而基于项目属性的推荐过程需要在用户–项目评分矩阵的

基础上,结合项目的属性信息转化为用户–项目属性值评分矩阵。

表 1 用户–项目评分矩阵

用户	项目					
	I_1	I_2	…	I_j	…	I_m
U_1	r_{11}	r_{12}	…	r_{1j}	…	r_{1m}
U_2	r_{21}	r_{22}	…	r_{2j}	…	r_{2m}
…	…	…	…	…	…	…
U_i	r_{i1}	r_{i2}	…	r_{ij}	…	r_{im}
…	…	…	…	…	…	…
U_n	r_{n1}	r_{n2}	…	r_{nj}	…	r_{nm}

项目资源的特征一般可以用一定数量的属性来描述,每个属性可以取若干个属性值,从而利用项目属性的取值情况对项目进行区分和描述,即项目资源 I 可以表示为元素属性值集的矢量:

$$\overrightarrow{I_j} = (C_{iP1}, C_{iP2}, \cdots, C_{iPo}) \tag{1}$$

其中, C_{iPj} 表示项目 I 的 p_j 属性的取值集。根据各个项目资源所具备的属性值的情况可以得到项目集的属性描述矩阵,如表 2 所示。

表 2 项目–属性矩阵

项目	属性					
	P_1	P_2	…	P_k	…	P_o
I_1	C_{1p1}	C_{1p2}	…	C_{1pk}	…	C_{1po}
I_2	C_{2p1}	C_{2p2}	…	C_{2pk}	…	C_{2po}
…	…	…	…	…	…	…
I_j	C_{jp1}	C_{jp2}	…	C_{jpk}	…	C_{jpo}
…	…	…	…	…	…	…
I_m	C_{mp1}	C_{mp2}	…	C_{mpk}	…	C_{mpo}

C_{iPj} 可能表示单一属性值或含有多个属性值的集合。比如菜式清蒸皖鱼的食材属性,取值可能为{皖鱼,葱,姜}。查找近邻用户和产生推荐结果是基于用户或项目资源之间的相似性计算,其中相似度计算方法的余弦相似度要求相关矢量包含的每个元素只能取单值,且取值必须量化,因此需要将项目–属性矩阵中的矢量转为只含有单个数值元素的矢量。

表 3 项目–属性值矩阵

项目	属性						
	P_1		…		P_O		
	p_{11}	…	p_{1g}	…	p_{o1}	…	p_{of}
I_1	$a_{1,11}$	…	$a_{1,1g}$	…	$a_{1,o1}$	…	$a_{1,of}$
…	…	…	…	…	…	…	…
I_m	$a_{m,11}$	…	$a_{m,1g}$	…	$a_{m,o1}$	…	$a_{m,of}$

对于离散型数值的项目属性,每个可能的属性取值转化为项目属性描述矢量中的一个元素,以 1 和 0 分别表示某个项目是否具备相应的属性。对于连续型数值的项目属性,首先将可能的属性取值范围进行离散化处理,然后按照属性具体的取值给相应矢量中的元素进行赋值,同样地以 1 和 0 分别表示项目是否具备该属性。处理后项目资源的每个属性 p 的取值可以表示为属性 p 所有可能离散取值的二进制矢量,项目–属性值矩阵如表 3 所示。

用户在使用项目结束后的评分反映了用户对项目属性的评价,整体上用 r 来描述用户对某项目资源的评分,而用 $s_{p1}, s_{p2}, \cdots, s_{po}$ 分别描述用户对项目中各属性的评分,最后以函数 f 表示用户对项目属性的评分与用户对项目的整体评分之间的转换函数:

$$r = f(S_{p1}, S_{p2}, \cdots, S_{po}) \tag{2}$$

在用户对项目资源的评分以及每个项目的属性取值数量足够多的前提下,则可以把函数 f 具体化,并运用该函数推算出用户–项目属性值评分矩阵。如果用 U 表示用户集, I 表示项目资源集, P 表示项目属性集, $\boldsymbol{M}$ 和 $\boldsymbol{M}'$ 分别表示用户–项目评分矩阵和用户–项目属性值评分矩阵,则有:

$$\boldsymbol{M}: U \times I \rightarrow \boldsymbol{M}': U \times P$$

矩阵的元素则为推算出的用户对项目属性值的评分,如表 4 所示。

表 4 用户–项目属性值评分矩阵

用户	属性						
	P_1			…	P_O		
	p_{11}	…	p_{1g}	…	p_{o1}	…	p_{of}
U_1	$s_{1,p11}$	…	$s_{1,p1g}$	…	$s_{1,po1}$	…	$s_{1,pof}$
…				…			
U_n	$s_{n,p11}$	…	$s_{n,p1g}$	…	$s_{n,po1}$	…	$s_{n,pof}$

2.2 基于 TF–IDF 算法的评分矩阵转化

项目属性值对系统分类结果影响的权重,反映了用户对项目属性的偏好程度。由此计算项目属性值对决定用户是否喜好某个项目资源所贡献的权重,以表示用户对于项目属性的偏好程度。

选取文本分类研究中计算文本特征项权重的 TF–IDF 算法来计算属性值的权重,在用户对项目资源的偏好分析中,如果在用户喜好的项目资源中某属性值出现的次数多,而该属性值出现在所有项目中的频率低,说明用户越偏好具有该属性值的项目,因此该属性值对决定用户喜好这个项目资源所贡献的权重越大[8]。

为了避免某个评分偏好对应的项目资源过少的问题,并考虑到推荐系统使用过程中消费者对商品只有选择和不选择这两种情况,可以根据用户对项目评分的高低将已评分的项目资源分为喜欢和不喜欢这两种偏好分类。假设用户 u 评分集共有 N 个项目资源,具有属性值 p_{ij} 的项目有 n_{pij} 个,喜欢的 N_a 个项目具有属性值 p_{ij} 的有 a_{pij} 个,不喜欢的 N_b 个项目具有属性值 p_{ij} 的有 b_{pij} 个,用户对属性值 p_{ij} 的初始偏好权重可以表

示为 $v_{u,pij}$:

$$v_{u,pij} = \left(\frac{a_{pij}}{N_a} - \frac{b_{pij}}{N_b}\right) \times \log(N/n_{pij}) \tag{3}$$

先求出用户对各个项目属性值的初始偏好权重,构成用户对项目资源各属性值的偏好权重矢量,再对每个用户的偏好矢量进行归一化处理后评分上限 top 相乘得到用户对各个属性值的偏好评分矢量。用户 u 对 p_{ij} 的偏好评分可以表示为 $s_{u,pij}$:

$$s_{u,pij} = \text{top} \times \frac{v_{u,pij} - \min(v_u)}{\max(v_u) - \min(v_u)} \tag{4}$$

2.3　协同过滤的用户偏好相似度计算

一般地,协同过滤推荐算法通常以两个用户对同一项目资源集给出的评分数值为基础,直接利用余弦相似性或 Pearson 相关相似性测量用户的相似度[9]。在餐饮 O2O 推荐的用户-商品评分矩阵中,具有明显的稀疏性,共同评分的菜式占目标用户评分集所有菜式的比例较小,且根据商品评分计算相似度忽略掉了用户偏好的具体细节。本研究将采用余弦相似性测量用户对于项目属性值偏好的相似度。用户 a 和 b 对项目属性值评分的相似度$\text{sim}_{\text{rating}}(a,b)$ 可以表示如下:

$$\text{sim}_{\text{rating}}(a,b) = \cos(\vec{s_a},\vec{s_b}) = \frac{\vec{s_a}\cdot\vec{s_b}}{\|\vec{s_a}\|\cdot\|\vec{s_b}\|} = \frac{\sum_{i=1}^{o}\sum_{j=1}^{f_i} s_{a,pij}\cdot s_{b,pij}}{\sqrt{\sum_{i=1}^{o}\sum_{j=1}^{f_i} s_{a,pij}^2}\cdot\sqrt{\sum_{i=1}^{o}\sum_{j=1}^{f_i} s_{b,pij}^2}} \tag{5}$$

其中,$\vec{s_a}$和$\vec{s_b}$表示用户 a 和 b 的项目属性值评分矢量,o 表示项目资源属性的个数,f_i 表示项目第 i 个属性的取值集的个数。

另外,该文采用静态上下文信息搜索的依据,同时考虑到用户评分记录较少时会存在用户冷启动问题[10]。如果两件事物之间存在很多相同的属性,即它们共享的信息越多,说明它们之间的相似度就越大;反之当两件事物之间没有共同的属性时,则认为它们是不相似的。为此,该文采用基于属性的语义相似性计算方法来衡量两个用户在基本信息上的相似度[11]。用户 a 和 b 在用户本体基本信息的语义相似性可以表示如下:

$$\text{sim}_{\text{profile}}(a,b) = \text{semsim}(\text{profile}_{ij}^{a},\text{profile}_{ik}^{b}) = \frac{|\ \text{profile}_{ij}^{a} \cap \text{profile}_{ik}^{b}\ |}{|\ \text{profile}_{ij}^{a} \cup \text{profile}_{ik}^{b}\ |} \tag{6}$$

其中,profile_{ij}^{a} 表示 a 用户本体基本信息的第 i 个属性的第 j 个取值,由相交的取值个数除以相并的取值个数得到用户 a 和 b 基本信息的语义相似性。

该文从用户对项目属性的评分偏好和用户本体基本信息两个方面综合计算用户偏好相似度。因此,基于上述两部分的相似度计算,本节采用线性加权的函数来计算综合相似度。用户 a 和 b 的综合相似度 $\text{sim}_{\text{combine}}(a,b)$ 可以表示如下:

$$\text{sim}_{\text{combine}}(a,b) = \gamma\text{sim}_{\text{rating}}(a,b) + (1-\gamma)\text{sim}_{\text{profile}}(a,b) \tag{7}$$

其中,γ 是可调系数,可用 Sigmoid 函数表示,其取值范围为[0,1]。当 γ 取值接近 1 时,表示主要通过用户对项目属性的评分偏好相似度来表示用户之间的相似度,当 γ 取值接近 0 时,表示主要通过用户本体基本信息的相似度来表示用户之间的相似度。

3　推荐生成过程

本研究融合了动态的情景因素,结合加权的贝叶斯方法进行基于上下文感知的项目属性动态偏好分析。根据相似组情景信息的历史记录,采用基于 KL 散度为情景信息加权的贝叶斯方法预测用户对于商品属性值的动态偏好,因而解决信息记录缺乏时难以构建概率模型以及推荐系统的用户冷启动问题。

3.1　基于贝叶斯方法的偏好分析

不同的上下文中各类型或具有某种属性特征的商品依概率来响应用户的选择偏好。本研究采用贝叶斯方法来预测用户在特定场所、活动等上下文信息对于商品属性值的动态偏好程度[12]。基于上下文感知的贝叶斯模型的构建需要一定数量训练集的支撑,但目标用户可能存在上下文信息记录缺乏而影响概率模型准确性的用户冷启动问题。该文对目标用户进行贝叶斯偏好模型构建时,认为偏好相似用户在相同或近似上下文中对于商品的选择偏好也具有相似性,根据用户对项目属性评分偏好以及静态上下文计算的综合相似度 $\text{sim}_{\text{combine}}(a,b)$ 进行排序,取排名较前且拥有足够数量历史行为记录的邻居用户作为相似组,并以所有相似用户的历史行为记录作为构建概率模型的训练数据集。用 c_i 来表示动态的关键上下文因素取值,则在用上下文因素的取值集为 $C=(c_1,c_2,\cdots,c_n)$ 的情况下,用户 u 选择具有 p_{ij} 属性的项目资源的概率可以表示如下:

$$P(p_{ij}\mid c_1,c_2,\cdots,c_n) = \frac{P(p_{ij},c_1,c_2,\cdots,c_n)}{P(c_1,c_2,\cdots,c_n)} = \frac{P(c_1,c_2,\cdots,c_n\mid p_{ij})P(p_{ij})}{P(c_1,c_2,\cdots,c_n)} \tag{8}$$

其中,$P(p_{ij}) = N_{pij}/N$ 表示训练数据集的记录中具有属性值 p_{ij} 的概率,N 为训练数据集中用户选择的项目资源总个数,N_{pij} 为用户选择具有属性值为 p_{ij} 的项目资源个数。式(8)通过计算每个属性值 p_{ij} 在训练数据中的特定上下文下的概率来预测用户对属性值的动态偏好。由于上下文具有多维特征,满足特定上下文的历

史记录样本量不足,因此使用贝叶斯公式进行转换。另外,由于计算 $P(c_1,c_2,\cdots,c_n \mid p_{ij})$ 需要一个较大的训练数据集,且上下文因素间的相互关系需要大量人为工作量,因此采用朴素贝叶斯方法假设在给定目标值时属性值之间互相条件独立,即在给定 p_{ij} 的情况下,观测到联合上下文的概率等于每个单独上下文因素取值的概率乘积:

$$P(c_1,c_2,\cdots,c_n \mid p_{ij}) = \prod_{k=1}^{n} P(c_k \mid p_{ij}) \tag{9}$$

考虑到由于部分项目记录较少而引起 $P(c_k \mid p_{ij})$ 为 0 的问题,定义在具有属性值 p_{ij} 的项目记录数 $N_{pij} < 10$ 的情况下,若 $P(c_k \mid p_{ij})$ 为 0,则设为 $P(c_k)$。将式(9)代入式(8),即可计算得基于上下文感知的贝叶斯模型:

$$p_{c,pij} = P(p_{ij} \mid c_1,c_2,\cdots,c_n) = \frac{P(p_{ij})\prod_{k=1}^{n} P(c_k \mid p_{ij})}{P(c_1,c_2,\cdots,c_n)} = \theta P(p_{ij}) \prod_{k=1}^{n} P(c_k \mid p_{ij}) \tag{10}$$

该模型在具有各个属性值的项目记录的条件概率下作比较,对于 $\forall p_{ij}$, $P(c_1,c_2,\cdots,c_n)$ 为恒定的,可设为与属性值 p_{ij} 无关的参数 θ,剩余部分具备条件概率比较特征[13]。由于贝叶斯偏好模型中上下文因素的取值规定为离散值,则上下文因素要考虑到对用户的动态偏好影响较大且其条件尽可能互相独立。

3.2 基于 KL 散度的加权方法

在基于上下文感知的贝叶斯模型中,上下文因素取值 c_k 对于各个属性值的动态偏好重要程度视为一致,权重均为 1。实际上不同 c_k 对于各个属性的选择偏好的影响效果存在差异,比如城市和健康状况上下文信息的某些实例对口味选择的影响可能比较大,但外卖或堂食时选择的口味可能与日常类似。因此,需要根据具体 c_k 对具体属性的影响程度来确定式(10)中上下文信息的权重。

此处引入 KL 散度用来衡量两个概率分布之间差异的指标。为了具体化不同上下文对于每个属性的重要程度,在计算用户在无上下文与具体上下文中对于单个属性的选择概率分布时,采用 KL 散度上下文加权方法[14],对于项目属性 P_i 来说,计算 c_k 值影响程度的 KL 散度可以描述为:

$$D_{Pi}(R_{ck} \mid\mid R) = \left(-\sum_{j=1}^{f_i} R_{ck,pij} \log \frac{1}{R_{ck,pij}}\right) + \sum_{j=1}^{f_i} R_{ck,pij} \log \frac{1}{R_{pij}} \tag{11}$$

其中,R_{pij} 表示用户选择具有属性值为 p_{ij} 的项目占属性 P_i 项目统计数的比率,$R_{ck,pij}$ 表示在上下文因素值 c_k 前提下的比率,f_i 表示项目属性 P_i 取值范围的属性值个数。当 KL 散度为 0 时,说明在某上下文下用户对属性 P_i 的选择与平常无差异,c_k 重要程度较低。而 KL 散度越大,则说明 c_k 对用户在属性 P_i 上的选择引起了较大的影响,应赋予 c_k 较大的权重。据此,本研究提出的计算上下文因素值 c_k 对于属性 P_i 重要程度的加权系数可以表示为:

$$W_{ck,P_i} = 1 + D_{P_i}(R_{ck} \mid\mid R) \tag{12}$$

式(10)中考虑加权系数后,即可计算得相应上下文信息加权的动态偏好模型:

$$t_{c,pij} = \theta P(p_{ij}) \prod_{k=1}^{n} P(c_k \mid p_{ij})^{W_{ck,Pi}} \tag{13}$$

3.3 推荐结果排序

推荐系统最后输出的结果是项目资源,因此项目属性的偏好值要转化为项目资源整体的偏好得分。基于属性效用叠加的思想,通过线性加权计算整体偏好得分,项目资源中同一属性有多个取值则取其平均值。用户 u 对于项目资源 v 的预测偏好得分 R_{uv} 可以表示如下:

$$R_{uv} = \sum_{i=1}^{o} w_i \cdot \overline{(s_{u,pij} \cdot t_{c,pij})} \tag{14}$$

其中,o 表示项目属性的个数,p_{ij} 为项目 v 所具有的属性值。w_i 为该项目第 i 个属性特征的权重,表示该属性对项目的重要程度,此权重值通过调查项目各属性的重要程度或咨询专家后得出。$s_{u,pij}$ 和 $t_{c,pij}$ 分别预测了用户 u 对于项目 v 所具有属性值 p_{ij} 的评分偏好和基于上下文感知的选择偏好,两者通过相乘结合,最后再取均值。

4 实验及结果分析

4.1 实验数据及处理

本研究设计一个用于收集用户上下文信息的 Web 页面,结合多种上下文信息获取方法来采集用户用餐时的记录。在考虑基于上下文感知的贝叶斯模型中上下文信息因素的取值约束和条件尽可能相互独立的原则,筛选出收集的数据。筛选后的属性包括 DayNight、Weather、Temperature(离散化后)、Holiday、AreaType、Acquaintance、Mode、Companion、Health,属性的取值主要通过手机浏览器和服务器从互联网等外部信息源直接获取,用户手动输入获取后 3 个属性。信息收集的页面如图 2 所示。

由于用户不易判断出菜式类的菜系等属性,而菜式的子类类型具有主观性且主要决定于食材和做法,因此不作为网页收集的菜式属性。实验选择便于用户判断的菜式属性:1 ~ 2 个食材、做法、口味、价格标签。为了方便用户分类并避免误判,部分属性值进行了合并和简化的处理,比如食材的选项按常识归纳缩减为

28 个类,包括畜肉类 4 项、禽肉类 4 项、蔬菜类 5 项、鱼虾蟹贝类 5 项、谷类、薯类、豆类、菌类等食材。做法选项有 10 个属性值,分别为炒、蒸、煲、煎炸类、烧烤类、炖煮类、灼烫类、烘焙、常温处理、其他做法。口味有 6 个属性值,分别为淡、咸、甜、苦、酸、辣。在加权结合用户评分偏好和基于上下文感知的选择偏好的式(14)中,属性效用权重 w_i 表示属性对菜式重要程度,食材取 0.4,做法、口味、价格标签各取 0.2。

图 2　信息收集页面

每次用餐记录可以保存 1 ~4 个菜式,信息获取错误等上下文信息缺失的情况不保存记录。通过输入菜式名保存的菜式在用户下次记录时可以直接通过下拉列表选择,将用户的记录中属性值相同的菜式设定为相同菜式,另外用户自定义菜式名的差异可忽略。根据上下文与项目评分的松耦合关系,评分表现的是用户对项目的固定偏好程度,因此同一用户在不同上下文中对相同菜式的重复评分记录取平均值。

在本研究中,邀请了各年龄段、各职业的用户进行用餐记录。在餐饮 O2O 平台注册时先对用户的静态信息进行采集,再为用户分配独立的页面链接,在移动设备的浏览器打开即可直接获取动态上下文信息,记录用餐内容信息以及评分。收集到的数据经过剔除和筛选,去除少于 5 条记录的用户后,共得到 34 位用户在 2019 年后三个季度期间的 2 170 条记录,合并属性值相同菜式后共得到不同的 356 个菜式。

4.2　结果分析

推荐预测准确度是评价电子商务推荐系统优劣的指标之一。针对用户的选择偏好程度进行基于上下文感知的预测,将用户对项目的选择偏好与评分偏好进行区分。为预测基于上下文感知的用户选择行为,实验以对比推荐列表和实际列表的准确率和召回率来评估个性化推荐的预测准确度。

实验数据划分为训练集和测试集,推荐系统根据训练集的数据得到一个在测试集条件下预测的推荐列表。通过准确率(Precision)表示推荐算法预测的推荐菜式结果符合用户偏好的程度,并通过召回率(Recall)表示用户偏好的菜式被推荐算法成功预测的程度[15]。假设 D 为实验的菜式列表,对于某目标用户,$R(d)$ 为推荐系统根据训练集的数据,采用不同算法得到的在测试集条件下预测的推荐列表,$T(d)$ 为测试集中该用户选择的菜式列表,准确率和召回率的计算公式可以表示如下:

$$\text{Precison} = \frac{\sum_{d\in D} | R(d) \cap T(d) |}{\sum_{d\in D} | R(d) |} \tag{15}$$

$$\text{Recall} = \frac{\sum_{d\in D} | R(d) \cap T(d) |}{\sum_{d\in D} | T(d) |} \tag{16}$$

从每位用户的记录中随机抽取 5 条包含上下文信息的用餐记录作为测试集,每条记录包含 1 ~4 个菜式,其余的所有记录作为训练集,计算平均的准确率和召回率。34 位用户各自的记录条数从 5 条至 183 条,因此实验的条件出现了传统协同过滤算法中用户冷启动的情况。菜式评分记录方面,同一用户对属性值相同菜式的重复评分记录取平均值后,以 34 位用户对不同菜式完整的 911 条评分作为近邻搜索的训练集,用户–菜式评分矩阵的缺失率为 92.5%,符合推荐系统实际使用中评分数据稀疏的情况。

本实验将对文中的上下文后过滤的协同过滤推荐方法 I,传统基于用户的协同过滤推荐方法 II,传统协同过滤近邻搜索的动态偏好预测推荐方法 III,基于上下文信息相似度的推荐方法 IV 进行比较。其中:

方法 II 根据对菜式的相同评分寻找邻居用户,再根据邻居用户的评分预测目标用户的缺失评分,其不融入上下文信息;

方法 III 根据传统协同过滤的方式寻找邻居用户,再按照本研究采用上下文信息加权的贝叶斯方法进行动态偏好分析,得到基于上下文感知的选择偏好;

方法 IV 按照本研究基于项目属性的方法搜索邻居,在目标用户以及邻居用户的历史上下文信息中,找出与当前上下文语义相似度最高的上下文,再推荐菜式[16]。

方法 I 和方法 IV 设定综合相似度表示式(7)中 $\gamma = \frac{1 - 1.1^{-n}}{1 + 1.1^{-n}}$,$n$ 为目标用户评分记录的条数,表示随着评分记录的增多,根据菜式评分计算得到的属性值评分可以更准确地代表用户偏好,γ 取值逐渐趋向于 1。

本实验对比了以上 4 种推荐方法在取不同邻居用户数量时得到的 TOP-10 推荐菜式的准确率和召回率,结果如图 3 和图 4 所示。

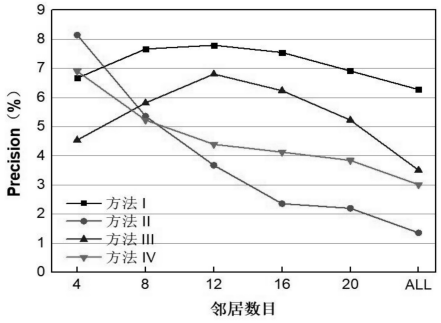

图 3 准确率

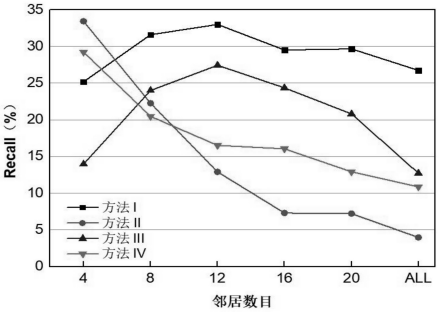

图 4 召回率

由图 3 和图 4 得,文中提出的上下文后过滤的协同过滤推荐方法的准确率和召回率整体上高于其他推荐算法,随着邻居数量的增加,准确率呈现先上升再下降的趋势。因为评分矩阵稀疏,传统基于用户的协同过滤推荐方法 II 在邻居数量较少时推荐结果较好,而待推荐菜式的已有评分记录菜式较少,因此在邻居数量上升时准确率和召回率下降较快,在邻居数目大于 12 时,其准确度均低于其他基于上下文感知的推荐方法,表明了融入上下文信息可以有效提高餐饮 O2O 推荐的准确度。方法 I 的准确度整体比方法 III 高,且相比以全体用户为邻居即不通过综合相似度选择邻居的情况表现更优,表明了基于项目属性的近邻搜索以及评分偏好预测方法的有效性。基于上下文相似度的推荐方法 IV 仅在邻居较少时有较高的准确率和召回率,邻居数目增大时准确度明显低于方法 I,表明用户在相似上下文中可能会选择相同菜式,但该方法没有考虑用户偏好信息,同时证实了采用上下文信息加权的贝叶斯方法进行动态偏好分析的有效性。

5 结束语

在为用户个性化地定制餐饮服务时,本研究融合了上下文后过滤的协同过滤推荐算法和基于贝叶斯模型的加权方法。经实验证明,所设计的模型能有效解决数据稀疏的问题,与其他算法进行对比,提高了推荐的准确率和召回率,表明更容易满足用户对个性化餐饮的要求。

此外,本研究还存在改善的空间:用户对菜品属性的偏好是随时间发生改变的,因此可考虑在推荐算法中引入与时间相关的因素,提高某时间段对用户的餐饮推荐;当用户群体较大时,可引入 K-means 聚类算法的思想进行算法改良;对特定的情景应该有更具体准确的分类,以提高推荐度。

参考文献:

[1] 朱伟权. 基于本体的餐饮 O2O 智能推荐方法研究[D]. 广州:华南理工大学,2019.

[2] PORTUNE K J, BENÍTEZ-PÁEZ A, DEL PULGAR E M G, et al. Gut microbiota, diet, and obesity - related disorders — the good, the bad, and the future challenges[J]. Molecular Nutrition & Food Research, 2017, 61(1): 1600252.

[3] 武慧娟,孙鸿飞. 基于认知计算与情境感知的个性化信息自适应推荐模式框架研究[J]. 情报科学,2018,36(5):114-118.

[4] DEY A K. Providing architectural support for building context-aware applications[D]. Atlanta: Georgia Institute of Technology, 2000.

[5] 史海燕,韩秀静. 情境感知推荐系统研究进展[J]. 情报科学,2018,36(7):163-169.

[6] 杨 凯,王 利,周志平,等. 基于内容和协同过滤的科技文献个性化推荐[J]. 信息技术,2019(12):11-14.

[7] 朱 磊,胡沁涵,赵 雷,等. 基于评分偏好和项目属性的协同过滤算法[J]. 计算机科学,2020,47(4):67-73.

[8] 王英杰. 基于 TF-IDF 的网络地理文本信息分类研究[J]. 科学技术创新,2020(10):76-77.

[9] 蒲鲜霖. 智能推荐系统中协同过滤算法综述[J]. 中国信通信,2018,20(23):31-32.

[10] 翁小兰,王志坚. 协同过滤推荐算法研究进展[J]. 计算机工程与应用,2018,54(1):25-31.

[11] ROBINSON P N, SCHULZ M H, BAUER S, et al. Methods for searching with semantic similarity scores in one or more ontologies:9002857[P]. 2015-04-07.

[12] 施 睿. 融合上下文信息的相关滤波跟踪算法研究[D]. 广州:华南理工大学,2017.

[13] 常志鹏,许 娟. 基于朴素贝叶斯算法的网络教学平台响应时间研究[J]. 数字技术与应用,2019,37(12):112-115.

[14] 胡 文,景玉海. 基于 KL 散度与 JS 散度相似度融合推荐算法[J]. 哈尔滨商业大学学报:自然科学版,2020,36(1):48-53.

[15] 陈 氢,冯进杰. 融合社交行为和社会化标签的移动情境感知服务研究[J]. 情报理论与实践,2019,42(2):114-119.

[16] 田雪筠. 基于情境感知的移动电子资源推荐技术研究[J]. 情报理论与实践,2015,38(5):86-89.