基于贝叶斯网络的上下文推荐算法

海本斋[1,2] 解瑞云[3]
(武汉理工大学信息工程学院 武汉 430070)[1] (河南师范大学计算机与信息工程学院 新乡 453007)[2]
(武汉理工大学计算机科学与技术学院 武汉 430070)[3]

摘 要 将上下文推荐系统同贝叶斯网络相结合,提出了一个上下文推荐算法,并设计了上下文资源推荐系统架构。首先利用贝叶斯网络,通过计算用户访问时间和资源信息的联合概率分布来取得用户在该环境下对资源的兴趣,然后比较当前用户所处环境所选取的资源与过去环境用户选取的资源的相似度,从而为用户提供合适的资源列表。最后将所提算法同其他常用的推荐系统算法进行了比较,系统架构按照 M/G/1 队列进行建模,对系统架构性能和稳定性进行了验证,取得较好结果。

关键词 推荐系统,上下文,贝叶斯网络,期望极大化
中图法分类号 TP399 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.07.057

Bayesian Network-based Context-aware Recommendation Algorithm

HAI Ben-zhai[1,2] XIE Rui-yun[3]
(College of Information Engineering, Wuhan University of Technology, Wuhan 430070, China)[1]
(College of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China)[2]
(College of Computer and Technology, Wuhan University of Technology, Wuhan 430070, China)[3]

Abstract Combining the context-aware recommender system with Bayesian networks, we proposed a context-aware recommendation algorithm, and designed a context-aware Resource Recommendation System Architecture. First using the Bayesian network, through computing the joint probability distribution of user access time and resource information, user interest of resources in the environment was obtained, and then the similarity of current user environment selected resources and past user environment selected resources was compared to provide appropriate resources list. Finally we compared the proposed algorithm with other common recommender system algorithms, and at the same time, the system architecture was modeled according to the M/G/1 queue. The result proves the superiority and stability of our system architecture and the algorithm are better.

Keywords Recommender system, Context, Bayesian network, Expectation-maximization

1 引言

随着互联网、物联网、普适计算和社会化网络的不断发展,人们已经进入"大数据"时代[1],"信息过载"问题变得更加严重。推荐系统能够从众多信息中为用户提供个性化推荐信息列表,从而满足用户的个性化需求,因此逐渐成为解决"信息过载"问题的重要方法。推荐系统主要有协同过滤推荐、基于物品过滤推荐和混合式推荐 3 种推荐方法[2],目前推荐系统已经在电子商务、旅游信息、社交网络以及信息检索等领域取得了巨大的成功。

近年来,上下文推荐系统(Context-Aware Recommender System)利用上下文信息在推荐方面的优势,通过获取和预测潜在用户偏好来过滤不相关的信息,为用户提供满足其个性化需求的结果,逐渐成为缓解"信息过载"的有效手段[3]。在上下文推荐系统中,如何利用上下文更准确地获取用户偏好并提供准确的个性化推荐成为目前主要的研究方向[4-7]。Werndl 等[8,9]提出了混合基于内容、基于知识和协同过滤 3 种常用推荐算法的上下文推荐方式;Wang 等[10]提出了将传统协同过滤方法与基于情绪特征上下文计算的协同过滤方法混合的新的推荐算法;Abbar 等人[11]从面向服务的角度提出一种上下文感知混合推荐系统框架。本文利用贝叶斯网络,提出了一种新的上下文推荐算法,通过计算用户访问时间和资源信息的联合概率分布来取得用户在该环境下对资源的兴趣,从而为用户提供合适的资源。通过测试表明,该算法在预测精度和推荐效率方面高于传统的推荐算法。

到稿日期:2013-10-16 返修日期:2013-11-07 本文受河南省教育厅科学技术研究重点项目:高校数字化教学资源整合与应用研究(12A520024),河南省教育厅科学技术研究重点项目:基于社会网络的个性化微博推荐技术研究(14A520085),河南省教育厅科学技术研究重点项目:基于网格的新型科技信息服务平台的关键技术研究(12B520016)资助。
海本斋(1980-),男,博士生,讲师,主要研究方向为非线性系统理论与智能检测技术,E-mail:haibenzhai@gmail.com;**解瑞云**(1980-),女,硕士,讲师,主要研究方向为网格技术。

2 基础知识

2.1 上下文感知推荐系统

对于上下文，采用 Dey 等人提出的定义："上下文是指可以描述实体(如人、地点)状态的任何信息。其中，实体是指与用户和应用程序交互相关的客体，包括用户与应用程序"[12]。上下文感知推荐系统既是一种推荐系统，也是一种上下文感知应用系统，Adomavicius 和 Tuzhilin 等人在文献[13]中指出，把上下文信息融入推荐系统将有利于提高推荐精确度。该系统将传统的用户-项目二维的评分效用扩展为多维的评分效用。因此上下文感知推荐系统可以被形式化为：

$$\forall (d_{j1},\cdots,d_{jl})\in D_{j1}\times D_{j2}\times\cdots D_{jl},$$
$$(d_{j1},\cdots,d_{jk})=\arg\max_{\substack{(d'_{j1},\cdots,d'_{jk})\in(D_{j1}\times D_{j2}\times\cdots D_{jk})\\(d'_{j1},\cdots,d'_{jl})=(d_{j1},\cdots,d_{jl})}} u(d_1',\cdots,d_n') \tag{1}$$

其中，$D_{j1}\times D_{j2}\times\cdots D_{jl}$ $(l<n)$ 为预测偏好的目标维度，$D_{j1}\times D_{j2}\times\cdots D_{jk}$ $(k<n)$ 是推荐结果维度，并且 $D_{j1}\times D_{j2}\times\cdots D_{jl}\cap D_{j1}\times D_{j2}\times\cdots D_{jk}=\varnothing$。

2.2 贝叶斯网络

贝叶斯网络是一组变量的联合概率分布，联合空间中每个变量在贝叶斯网络中称为一个结点。其定义如下：

给定一组变量集和 $X=(x_1,x_2,\cdots,x_m)$，若满足如下条件，则称(G,P)为贝叶斯网络。

1)由变量集中所有结点构成有向无环图 G。当结点 Y 到 X 存在一条有向的路径时，称 X 是 Y 的后继。

2)每个变量都有一个条件概率分布，用来描述该变量在给定其立即前驱时的概率分布。

则对变量组$(X_1,\cdots,X_n)$赋值为$(x_1,\cdots,x_n)$的联合概率分布可以表示为：

$$P(x_1,\cdots,x_m)=\prod_{i=1}^{n}P(x_i\mid Parents(X_i)) \tag{2}$$

其中，$Parents(X_i)$为 X_i 的直接前驱的集合。

2.3 期望极大化算法

期望极大化算法 EM(Expectation-maximization)也是一种未知参数的极大似然估计算法，通过假设隐变量的存在，简化似然方程，将比较复杂的似然函数极大似然估计问题转化为比较简单的极大似然估计问题。

令 $\theta=(\mu_1,\mu_2)$ 表示基准概率分布的参数，μ_1,μ_2 为任意初始值；$X=(x_1,x_2,\cdots,x_m)$代表被观察到的数据，$Y=X\cup Z$ 表示全部数据，Z 代表未被观测到的数据，把它看作一个随机变量，它的概率分布依赖于位置参数 θ 和已知数据 X。这里使用 h 代表参数 θ 的假设值，h'表示在 EM 算法的每次迭代中修改的假设。EM 算法通过寻找使 $E[\ln P(Y\mid h')]$最大的 h'来得到极大似然假设h'。

首先假设 h'下 Y 的似然度为 $P(Y\mid h')$，然后使 $\ln P(Y\mid h')$最大化从而使 $P(Y\mid h')$最大化。定义函数 $Q(h\mid h')$，该函数将 $E[\ln P(Y\mid h')]$作为 h'的一个函数给出，则当 $\theta=h$ 时且在全部数据 Y 中观察到的数据 X 已假定的情况下，就有

$$P(Y\mid h')=Q(h'\mid h)=E[\ln p(Y\mid h')\mid h,X] \tag{3}$$

然后重复执行以下两个步骤，直到收敛。

步骤 1　估计(E)步骤：使用当前假设 h 和观察到的数据 X 来估计 Y 的概率分布从而计算 $Q(h\mid h')$，就有：

$$Q(h\mid h')\leftarrow E[\ln P(Y\mid h')\mid h,X] \tag{4}$$

步骤 2　最大化(M)步骤：将假设 h 替换为使 Q 函数最大化的假设h'，就有：

$$h\leftarrow\arg\max_{h'}Q(h'\mid h) \tag{5}$$

当函数 Q 连续时，EM 算法收敛到似然函数 $P(Y\mid h')$的一个不动点。

3 上下文资源推荐系统架构

该系统具有数据预处理和个性化推荐算法实现功能。其中数据预处理和收集主要包括以下几个方面：

1)人口统计学方面信息：主要包括用户年龄、性别、年级、专业等，从用户注册信息中获取。

2)环境信息：主要包括用户所处位置、日期时间信息。

3)偏好信息：以往用户喜欢资源的类型等。

4)资源信息：主要有资源的类别、大小、添加时间、从属学科、作者等。

整个系统架构如图 1 所示。

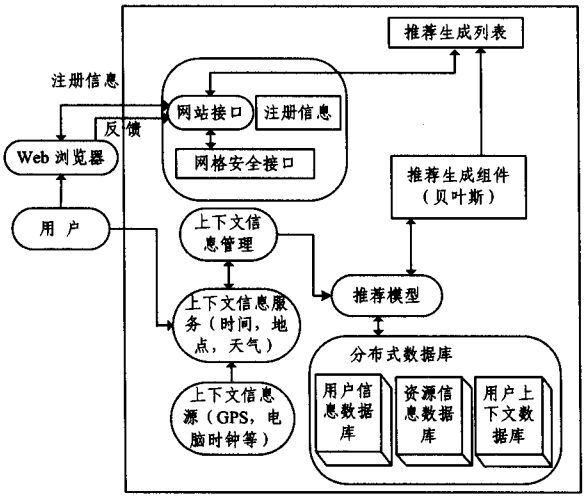

图 1　上下文资源推荐系统架构

整个系统包括用户接口、中间件、推荐服务和数据服务。主要的组件包括以下几个方面：

1)网络接口

通过浏览器访问系统的 Web 界面，包括防范非法用户和恶意用户的网格安全界面、用户和资源注册信息界面、用户提供资源信息界面以及用户提交查询的界面。

2)上下文管理组件

通过用户当前的环境信息来联系合适的上下文信息服务并将这些信息发送给推荐网络服务组件，获得相应的检索信息。地址信息的获取主要是通过 GPS 定位技术来取得，距离信息通过计算用户和资源之间的距离来获得，时间和日期信息通过计算机获取。

3)推荐模型

该组件是用于数据访问和集成的一个架构，主要用来解决上下文信息以及初始数据的预处理。该架构支持在一个结构下集成不同类型的数据库，如 XML 数据库和关系数据库等。它提供 3 种基本功能：查询数据、传输数据和参数学习，所有的信息(无论是用户、资源还是推荐结果等)都通过该组件存储在分布式数据库中。

4)推荐生成组件

这个组件使用贝叶斯网络产生推荐结果，一般会根据用户所处的环境信息为该用户推荐合适的前5个信息。

系统工作流程如下：

1)当需要推荐服务时，用户通过网络向系统发送推荐请求。

2)推荐网络服务激活系统数据库中关于该用户的信息，收集该用户的人口统计信息也就是注册信息，通过上下文管理组件获得用户所处环境信息，并通过推荐模型获得资源信息。服务代理开始部署不同的机器运行推荐服务，并通过网络接口将推荐算法生成资源信息列表传送到用户的手机。

3)浏览完资源后，用户需要提交关于该资源的评价，便于系统以后为用户提供更好的推荐服务。

4 推荐生成过程

本文采用贝叶斯网络来给出用户的兴趣模型，从而为用户提供合适的资源。

为用户在当前的环境下提供合适的资源推荐，需要对图2所示的各个信息之间的关系进行合适的处理。利用贝叶斯网络，通过计算用户访问时间和资源信息的联合概率分布来计算用户在该环境下对资源的兴趣[14]。考虑到用户的偏好，设计了下面的贝叶斯网络，如图2所示。从这个结构中可以获得用户 U 对于资源 R 的兴趣概率 $Perference(u,r)$。

$Perference(u,r)=p(interest|$用户年龄，用户职业，资源分类，访问类型，资源添加时间$)$

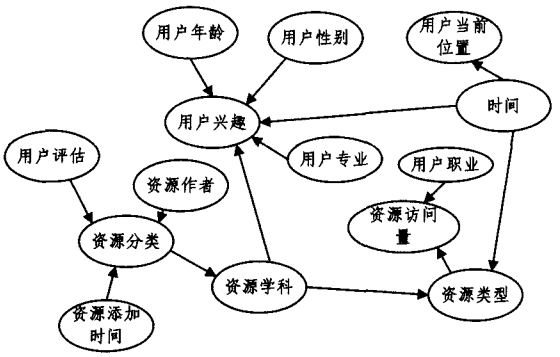

图2 获得用户兴趣概率的贝叶斯结构

为能反映用户环境的实时变化，算法引入了参数学习技术，在收集已有的数据基础上，我们使用期望最大化算法(EM)来获得条件概率列表。

V 代表变量集合$\{x_1,x_2,x_3,\cdots,x_n\}$，$S$ 代表贝叶斯网络结构，S 中每一个变量的值就是$\{x_i^1,x_i^2,x_i^3,\cdots,x_i^r\}$。$D=\{c_1,c_2,c_3,\cdots,c_m\}$是样本集。$\pi_i$ 是 x_i 的父集。那么我们就可以用 $p(x_i^k,\pi_i^j)$表示 $x_i=k,\pi_i=j$ 时的概率，记为 θ_{ijk}。给定贝叶斯机构 S、样本集 D，贝叶斯网络参数学习的过程就是计算条件概率 $p(\theta|D,S)$。

使用EM算法计算参数集合 D 和所有变量 x_i 的条件概率 $p(x_i,\pi_i|D_l,\theta^{(t)})$，给定集合 D，得到对数似然函数：

$$l(\theta|D)=\sum_l \ln p(D_l|\theta)=\sum_{ijk} f(x_i^k,\pi_i^j)\ln\theta_{ijk} \tag{6}$$

其中，$f(x_i^k,\pi_i^j)$表示当 $x_i=k,\pi_i=j$ 时的值，由此得到最大似然函数 θ：

$$\theta_{ijk}=\frac{f(x_i^k,\pi_i^j)}{\sum f(x_i^k,\pi_i^j)} \tag{7}$$

使用EM算法初始化 $\theta(0)$的值，并不断修正，从 $\theta(t)$到 $\theta(t+1)$有两步，也就是期望值和最大化。期望值的计算就是当 D 给定时，当前 θ 的期望值为：

$$l(\theta|\theta^{(t)})=\sum_l\sum_{x_i}\ln p(D_l,X_l|\theta)p(X_l|D_l,\theta^{(t)}) \tag{8}$$

对于所有的 θ，有 $l(\theta|\theta^{(t)})\leqslant l(\theta|\theta^{(t+1)})$，当 $f_i(x_i,\pi_i)=\sum_l p(x_i,\pi_i|D_l,\theta^{(t)})$时，由式(8)可得：

$$l(\theta|\theta^{(t)})=\sum_{i,j,k} f_i(x_i^k,\pi_i^j)\ln_{ijk} \tag{9}$$

通过最大化当前对数似然函数的期望值得到最大值：

$$\theta_{ijk}{}^{(t+1)}=\arg\max E[p(D|\theta)|D',\theta(t),S]=\frac{f(x_i^k,\pi_i^j)}{\sum_j f(x_i^k,\pi_i^j)} \tag{10}$$

式(8)得到期望值，式(10)得到最大值。

用户所处上下文信息 S 用时间、地点、网络关系3个特征表示，记为 $S=(F_t,F_l,F_s)$，由用户手机或其他移动设备时钟来取得时间信息 F_t，GPS获得用户的地址信息 F_l，用户的日历获得用户的网络关系 F_s。因此就有 $S^c=(F_t^c,F_l^c,F_s^c)$表示用户当前环境，$S^p=(F_t^p,F_l^p,F_s^p)$表示用户过去环境，UP 表示用户对资源的兴趣，给定当前用户环境 S^c，那么使用式(11)比较当前用户所处环境所选取的资源和过去环境用户选取的资源的相似度 S^r。

$$S^r=\arg\max_{j\in S^p}(\sum_j \partial_j.\ \mathrm{sim}_j(F_j^c,F_j^i)) \tag{11}$$

sim_j 表示当前环境第 j 个维度 F_j^c 和过去环境第 j 个维度 F_j^i 两个环境向量之间的相似矩阵，∂_j 表示第 j 个维度的权重。

结合式(10)和式(11)，取得当前环境下用户感兴趣的资源列表。整个系统的推荐流程如图3所示。

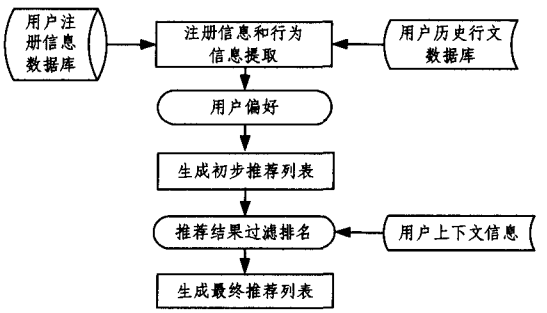

图3 推荐生成流程表

5 实验结果及分析

5.1 算法评测

算法评测使用的数据集采用的是上下文推荐系统领域通用的moviepolit数据集，它包括了从2008年1月到2010年3月的105131个用户的4544386个评分，数据稀疏度为0.96865，实验结果采用平均绝对误差(MAE)来评测算法预测的准确度。

$$MAE=\frac{\sum_{u,s\in S}|\overline{r_{u,s}}-r_{u,s}|}{|S|} \tag{12}$$

S 为测试集合，$\overline{r_{u,s}}$为用户 u 对资源 s 的预测评分，$r_{u,s}$为数据集 S 中用户 u 对资源 s 的实际评分。

在本次测试中，使用了圣诞周推荐任务的测试集，每次随机选取10%的用户预测其喜好，进行5次实验，取MAE的均值。其中KNN算法取近邻大小为5，SVD算法中正则化控制参数设置为0.005，特征数取25，迭代次数为20。所得结果如表1所列。

表 1　常用推荐系统算法与本文算法的 MAE 值

推荐算法	MAE
ItemCF	0.86158
UserCF	1.03740
本文算法	0.70187
Slope One	0.71978
KNN(k=5)	0.83184
SVD	0.70493

MAE 值越小，预测精度越高。因此从表 1 中可以看出 UserCF 算法的预测精度最差，本文提出的算法在预测准确度上相对于 ItemCF、UserCF 和 KNN 算法有较大优势，与 Slope One 算法相比预测误差由 0.71978 降低到 0.70187，降低了 0.01791，与 SVD 算法相比也降低了 0.00306，取得了较好的预测精度。

5.2　系统性能评价

将系统架构按照 M/G/1 队列进行建模，M/G/1 队列包含一个按照速率为 λ 的泊松进程的随机请求 FIFO 缓冲区和从队列中检索请求并提供服务的服务器。系统按照平均服务速率为 μ 的先到先服务的机制为队列中的所有请求提供服务。

网络负载具有重尾分布的特性，用函数表示为，当 $0\leqslant\partial\leqslant2$ 时，$\Pr\{X>x\}\sim x^{-\partial}$，在该系统中，用户请求所产生的推荐的多样性与资源在数据库中的编号相关。一个给定任务大小的资源推荐服务能力遵循有界 pareto 分布，有界 pareto(k, p,∂)的概率密度函数是：

$$f(x)=\frac{\partial k^{\partial}}{1-(k/p)^{\partial}}x^{-\partial-1} \tag{13}$$

其中，∂表示任务大小变化，k 表示可能的最小任务，p 表示最大的任务，$k\leqslant x\leqslant p$。通过∂的变化，观察呈现的中度可变性($\partial=2$)到高可变性($\partial=1$)的分布变化规律。

下面开始等待时间 $E(W)$的推导，W 表示用户获得服务前必须等待的时间，$E(N_q)$表示等待获得服务用户的数量，$E(X)$表示平均服务时间，由 little 法则知，平均队列长度 $E(N_q)$可以用等待时间来表示，因此，$E(N_q)=\lambda E(W)$，服务器上的负载 $\rho=\lambda E(W)$。设 $E(X^j)$表示在 j 时刻任务的服务分布，则有：

$$E(X^j)=\begin{cases}\dfrac{\partial pj((k/p)^{\partial}-(k/p)^j)}{(j-\partial)}, & \text{if } j\neq\partial\\ \dfrac{\partial k^{\partial}\ln(p/k)}{(1-(k/p)^{\partial})}, & \text{if } j=\partial\end{cases} \tag{14}$$

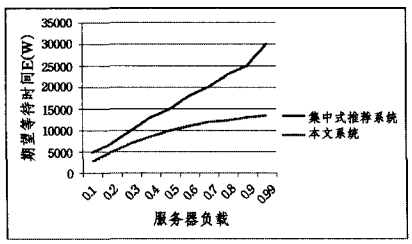

图 4　本文系统与集中式推荐系统反应时间的对比

因此，使用 P-K 公式，就可以得到系统中队列的预期等待时间 $E(W)=\lambda E(X^2)/2(1-\rho)$。集中式的推荐系统，当服务时间增加时，预期等待时间 $E(W)$会增加，服务器的负载也会增加。在网格环境下，网络负载会被分布到不同的服务器上，主服务器的新负载将会是$\bar{\rho}=(1-Predirect)$，新到达速率为$\bar{\lambda}=\lambda(1-Predirect)$。图 4 显示了在$\partial=1.5$，重定向概率 $Predirect=0.5$ 时，本文提出的推荐系统和传统的集中式推荐系统的预期等待时间的对比。可以看出，本文系统的预期等待时间有了较好的提升。

结束语　本文提出了一个基于贝叶斯网络的上下文推荐系统架构，并给出了相应的算法。通过研究发现，该算法取得了比传统的推荐算法更好的预测精度，同时在系统性能上也比传统的集中式推荐系统取得更好的推荐效率和稳定性。文中的算法评测使用了圣诞周推荐任务的测试集，将时间信息作为上下文信息，未来将继续研究将距离、用户评论等内容纳入到上下文信息中，进一步提高预测精度。

参 考 文 献

[1] Borkar V, Carey M, Li C. Inside Big Data Management: Ogres, onions, or parfaits? [C]// Proc. 15th International Conference on Extending Database Technology. Berlin, ACM, 2012: 3-14

[2] Linden G, Smith B, York J. Amazon. com recommendations: Item-to-item collaborative filtering[J]. IEEE Internet Computing, 2003, 7: 76-80

[3] Wang L C, Meng X W, Zhang Y J. Context-Aware recommender systems. Ruanjian Xuebao[J]. Journal of Software, 2012, 23(1): 1-20

[4] Xu F L, Meng X W, Wang L C. A collaborative filtering recommendation algorithm based on context similarity for mobile users[J]. Journal of Electronics & Information Technology, 2011, 33(11): 2785-2789

[5] Xie H T, Meng X W. Intelligent configuration recommendation of context-aware mobile application[C]// Proc. of the Globlecom 2011 Workshop on Ubiquitous Computing and Networks. IEEE Computer Society, Washington, 2011: 1263-1268

[6] Ricci F, Rokach L, Shapira B, et al. Recommender Systems Handbook[M]. Berlin: Springer-Verlag, 2011: 1-842

[7] Zheng Y, Zhang L, Ma Z, et al. Recommending friends and locations based on individual location history[J]. ACM Trans. on the Web(TWEB), 2011, 5(1): 1-44

[8] Woerndl W, Schueller C, Wojtech R. A hybrid recommender system for context-aware recommendations of mobile applications [C]// Proc. of the WPRSIUI IEEE Computer Society. Washington, 2007: 871-878

[9] Woerndl W, Brocco M, Eigner R. Context-Aware recommender systems in mobile scenarios[J]. Int'l Journal of Information Technology and Web Engineering, 2009, 4(1): 67-85

[10] Wang L C, Meng X W, Zhang Y J, et al. New approaches to mood-based hybrid collaborative filtering[C]// Proc. of the RecSys 2010 Workshop on CAMRa. New York, ACM Press, 2010: 28-33

[11] Abbar S, Bouzeghoub M, Lopez S. Context-Aware recommender systems: A service-oriented approach[C]// Lyon: Proc. of the VLDB Workshop on PersDB. 2009: 1-6

[12] Dey A K. Understanding and using context [J]. Personal and Ubiquitous Computing, 2001, 5(1): 4-7

[13] Adomavicius G, Sankaranarayanan R, Sen S, et al. Incorporating contextual information in recommender systems using a multidimensional approach[J]. ACM Trans. on Information Systems (TOIS), 2005, 23(1): 103-145

[14] Chu D C, Humphrey M. Mobile OGSI. NET: Grid Computing on Mobile Devices[C]// The 5th IEEE/ACM Int. Workshop on Grid Computing. Washington, Pittsburgh, PA, 2004: 452-460

[15] 张步良. 基于分类概率加权的朴素贝叶斯分类方法[J]. 重庆理工大学学报：自然科学版, 2012, 26(17): 81-83