

隐式反馈场景中融合社交信息的上下文感知推荐

俞春花 刘学军 李 斌

(南京工业大学电子与信息工程学院 南京 211816)

摘 要 作为解决信息过载问题的有效方式,推荐系统能够根据用户偏好对海量信息进行过滤,为用户提供个性化的推荐。对如何利用隐式反馈数据进行个性化推荐进行了研究,提出了一种融合上下文信息和用户社交信息的隐式反馈推荐模型(Implicit Feedback Recommendation Model Fusing Context-aware and Social Network Process,IFCSP)。首先从数据集中提取与用户兴趣相关的上下文信息的属性集合,并以此作为分裂属性,使用决策树分类算法对“用户-产品-上下文”集合进行分类,从而将历史选择集合分组。对于要推荐的用户,根据其选择产品时的上下文信息,匹配最相似的分组,再使用基于隐式反馈的推荐模型(Implicit Feedback Recommendation Model,IFRM)预测用户对未选择产品的偏好,并结合用户的社交信息,进而对用户进行产品推荐。实验表明,该模型在平均正确率均值(MAP)和平均百分百排序(MPR)评价指标上均优于其他 4 种算法,可以显著提高系统的预测和推荐质量。

关键词 推荐系统,隐式反馈,上下文感知推荐,社会化推荐,IFRM

中图法分类号 TP301 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.6.049

Implicit Feedback Personalized Recommendation Model Fusing Context-aware and Social Network Process

YU Chun-hua LIU Xue-jun LI Bin

(College of Electronic and Information Engineering, Nanjing Tech University, Nanjing 211816, China)

Abstract As a key solution to the problem of information overload, the recommender system can filter a large amount of information according to a user's preference and provide personalized recommendations for users. This paper explored the area of personalized recommendation based on implicit feedback and proposed a recommendation model, namely implicit feedback recommendation model fusing context-aware and social network process(IFCSP), which is a novel context-aware recommender system incorporating processed social network information. This model handles contextual information by applying a decision tree algorithm to classify the original user-item-context selections so that the selections with similar contexts are grouped. Then implicit feedback recommendation model (IFRM) was employed to predict the preference of a user for a non-selected item using the partitioned matrix. In order to incorporate social network information, a regularization term was introduced to the IFRM objective function to infer a user's preference for an item by learning opinions from his/her friends who are expected to share similar tastes. The study provides comparative experimental results based on the typical Douban and MovieLens-1M data sets. Finally, the results show that the proposed approach outperforms state-of-the-art recommendation algorithms in terms of mean average precision (MAP) and mean percentage ranking (MPR).

Keywords Recommender system, Implicit feedback, Context-awareness recommendation, Social recommendation, IFRM

1 引言

互联网技术与应用的快速发展在带来方便的同时,也产生了信息超载(Information Overload)问题,如何从海量信息中准确而快速地过滤出用户所需要的信息,已成为当代信息技术发展的一项挑战,也是目前学者们研究的热点问题。推荐系统(Recommender System)作为信息过滤的重要手段,是当前解决信息超载问题的有效方法^[1]。

构造个性化推荐系统的关键资源是用户历史行为数据,

具体可分为显式用户反馈(Explicit User Feedback)和隐式用户反馈(Implicit User Feedback)两类。其中,显式用户反馈是用户给出的显式倾向,如评分等。传统的协同过滤模型在推荐过程中仅仅对评分值本身进行一般性运算,不考虑其他任何信息,当信息量越来越大时,会带来稀疏性、扩展性、冷启动等问题,这在一定程度上限制了系统性能的进一步提高,推荐质量不高。隐式用户反馈不直接给出用户倾向,如购买商品、浏览网站等。隐式反馈只包含“已选择”与“未选择”两类(本文将点击、交易、转发等各种交互行为称为“选择”),收集

到稿日期:2015-05-08 返修日期:2015-08-02 本文受国家自然科学基金(61203072),国家公益性科研专项(201310162),江苏省重点研发计划(社会发展)(BE2015697)资助。

俞春花(1991-),女,硕士生,主要研究方向为数据挖掘、个性化推荐,E-mail:chunhuayu1991@163.com;刘学军(1971-),男,博士,教授,CCF高级会员,主要研究方向为数据库、数据挖掘、传感器网络等;李斌(1979-),男,硕士,讲师,主要研究方向为传感器网络、智能信息处理。

隐式反馈数据不需要用户额外的努力,不易引起用户反感,因此它具有收集成本更低、应用场景更广、数据规模更大的特点^[2]。然而,大多数推荐系统都只关注显式反馈信息,利用传统的协同过滤算法推荐,并进行优化,例如考虑用户评分时的上下文信息^[3,8,9-12],或解决特定的问题,如冷启动^[15-17]、稀疏性^[18,19]、扩展性^[7]等。但是这样浪费了宝贵的海量数据资源,因为用户愿意给出显式评分的数据量远小于隐式反馈的信息量,而且还限制了大数据时代推荐系统的发展。

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 等人较早就指出,把上下文信息融入推荐系统将有利于提高推荐精确度,并提出被广泛引用的“上下文感知推荐系统(Context-aware Recommender Systems, CARS)^[8-10]”的概念。Adomavicius^[8]等人提出多维度上下文用户偏好相似性计算模型,实验证明该模型在推荐精确度、多样性方面表现较好。文献^[12]则将 Factorization Machines 技术应用于快速上下文感知推荐生成,其能够处理不同类型的上下文,推荐精确度和实时性俱佳。此外,一些上下文感知协同过滤技术采用机器学习技术或者数学统计相关的模型(如张量分解^[11,12]),能够充分利用各种模型在处理高维数据方面的优势,成为近年研究热点。

在推荐系统领域,使用社交网络信息向目标用户推荐产品是另一个研究热点^[14,21]。Liu^[14]等人提出使用用户的社会化信息增强协同过滤的性能,证实了使用社会网络信息能提高推荐精度。文献^[17]提出了一种基于动态社会行为和用户背景的协同推荐方法,实验证明该方法不仅能较好地解决数据稀疏和冷启动的问题,还能有效提高推荐算法的精确度。

针对上述问题,本文在隐式反馈的场景中,提出了一种融合上下文信息和用户社交信息的隐式反馈推荐模型(Implicit Feedback Recommendation Model Fusing Context-aware and Social Network Process, IFCSPP)。该模型的贡献主要包括以下3个方面:1)首先从数据集中提取与用户兴趣相关的上下文属性集合,然后以此作为分类依据,使用决策树分类算法 C4.5 把历史“用户-产品-上下文”选择集合分组,则组内(即原集合的子集)包含相似的选择情景。换句话说,由于集预测目标用户对产品的选择倾向比使用原始的历史选择集合进行预测的推荐精度更高。并且 C4.5 算法可以处理各种类型的上下文信息,如离散值和连续值。2)对于要推荐的用户,根据其选择产品时的上下文信息,匹配最相似的分组,使用隐式反馈推荐模型(Implicit Feedback Recommendation Model, IFRM)^[2]预测用户对产品的选择倾向。为进一步提高推荐精度,在 IFRM 的基础上,增加一个社会正则化项。通过对比目标用户及其朋友的历史选择信息,找出与用户兴趣品味相似的朋友,利用朋友选择产品的信息改善对目标用户的推荐。3)在豆瓣和 MovieLens-1M 数据集上测试了算法在平均正确率均值(MAP)和平均百分比排序(MPR)评价指标上的性能,分析了用户的社会关系、推荐时的上下文信息以及算法中的参数设置对实验结果的影响。

本文第2节介绍隐式反馈推荐模型 IFRM;第3节提出模型 IFCSPP,其中3.1小节给出推荐问题的形式化定义,3.2小节在模型中融合用户选择产品时的上下文信息,使用 C4.5 算法生成最优决策树,3.3小节在模型中融合用户社交信息,

3.4小节具体描述了 IFCSPP 模型的推荐过程;第4节给出实验和结论,验证 IFCSPP 模型的推荐效果;最后对全文进行总结。

2 隐式反馈推荐模型(IFRM)

在介绍文章提出的模型之前,首先介绍隐式反馈推荐模型 IFRM,并讨论 IFRM 如何预测用户对产品的选择倾向。

基于隐式反馈进行推荐的难点在于缺乏显式的负例,即我们明确地知道用户喜欢什么但不清楚用户不喜欢什么。IFRM 通过建模用户与产品间的选择倾向,最大化可观察用户行为发生的可能性来构造和训练推荐模型,从而避免引入负例,提高预测准确率^[2]。算法设计如下:

将用户 i 与产品 j 表示为 K 维特征空间的一组向量; $U_i = (U_{i1}, U_{i2}, \dots, U_{iK})$, $V_j = (V_{j1}, V_{j2}, \dots, V_{jK})^T$, A_{ij} 表示用户 i 对产品 j 的选择倾向程度,则

$$A_{ij} = U_i \cdot V_j = \sum_{k=1}^K U_{ik} V_{jk} \quad (1)$$

用户 i 选择产品 j 的概率:

$$P_{ij} = \varphi(\Delta_{ij}) = \frac{\Delta_{ij}}{1 + \Delta_{ij}} \quad (2)$$

其中

$$\Delta_{ij} = \frac{A_{ij}}{A_i} = \frac{A_{ij}}{\frac{1}{M} \sum_{k=1}^M A_{ik}} \quad (3)$$

A_i 表示用户 i 对全体 M 个产品的平均选择倾向。

IFRM 模型根据可观察选择行为集合 O (即训练集 $O = \{(i, j) | \text{用户 } i \text{ 选择产品 } j\}$), 最大化潜在特征 U, V 的后验概率 $P(U, V | O)$, 求得潜在特征矩阵 U, V , 等价于最小化以下优化目标:

$$\arg \min_{U, V} L_1 := \sum_{(i, j) \in O} \ln(1 + \Delta_{ij}^{-1}) + \lambda (\|U\|_F^2 + \|V\|_F^2) \quad (4)$$

其中, $\|*\|_F^2$ 表示 2 范数, $\lambda = \frac{1}{2\sigma^2}$ 是调节参数,用于调节控制潜在特征矩阵 U, V 的复杂度,以防止过拟合。

在预测时,对任意给定的用户 i 以及候选产品集 $\{j_1, j_2, \dots, j_p\}$, 根据由式(1)计算出的 A_{ij} 由高到低对候选产品进行排序,即可产生推荐列表。

可以使用随机梯度下降法(SGD)对式(4)优化,对于观察到的选择行为 (i, j) , 分别对 U_i, V_j 求偏导作为梯度:

$$\frac{\partial L_1}{\partial U_i} = f(i, j) \bar{V} - g(i, j) V_j + 2\lambda U_i \quad (5)$$

$$\frac{\partial L_1}{\partial V_j} = -g(i, j) U_i + 2\lambda V_j \quad (6)$$

其中

$$f(i, j) = \frac{1}{U_i V_j + U_i \bar{V}} \quad (7)$$

$$g(i, j) = \frac{U_i \bar{V}}{U_i V_j} f(i, j) = \frac{U_i \bar{V}}{U_i V_j (U_i V_j + U_i \bar{V})} \quad (8)$$

其中, U_i 表示用户 i 的特征向量, V_j 表示产品 j 的特征向量, $\bar{V} = \frac{1}{M} \sum_{j=1}^M V_j$ 表示全部产品的平均特征向量。

与传统的基于显式反馈推荐方法不同,IFRM 并不去拟合具体评分值,它只关注选择行为本身,避免了引入噪声,IFRM 天然地适用于没有评分而只有选择交互行为的隐式反馈推荐场景。

3 隐式反馈场景中融合社交信息的上下文感知推荐 IFCSF

3.1 问题定义

在这样一个“已选择”与“未选择”二值推荐问题中,我们的目标就是利用观察到的历史选择和用户选择产品时的上下文信息及其社交网络去预测推荐列表中的缺失项(0或1),从而判断用户最终是否会选择某个推荐对象。传统的推荐系统通常只考虑用户和产品二维信息(见图1(a)),然而在很多系统中,丰富的上下文信息可以为推荐系统提供一个新的维度(见图1(b))。把上下文信息分成两类,一类是静态上下文,用于描述用户的特征,比如年龄、性别、工作等;另一类是动态上下文,用于描述情景发生时的条件,比如用户当时的情绪、时间、地点等。另外,在线社交网络为预测用户的偏好提供了另外一种方式,认为好友与用户有类似的选择(见图1(c))。因此,在这一部分把上下文信息和社交网络信息融入隐式反馈模型中,提出 IFCSF 模型。

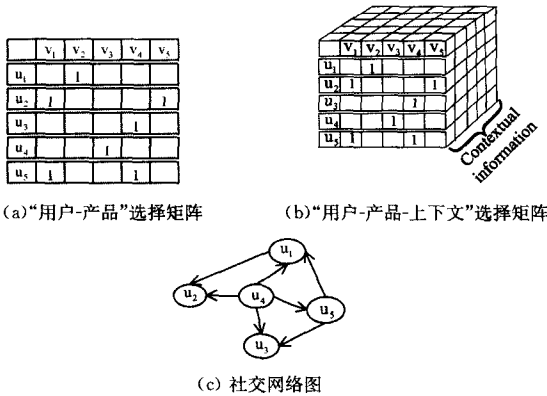


图1 上下文感知社会化推荐

将用户集合表示为 $U = \{U_1, U_2, \dots, U_m\}$, 产品集合表示为 $V = \{V_1, V_2, \dots, V_n\}$ 。任何一个用户可以根据自己的偏好选择产品,用户 U_u 选择产品 V_v , 记为 $S_{u,v} = 1$, 所有的选择记录 $S = \{S_{u,v} | U_u \in U, V_v \in V\}$ 组成一个选择列表。与选择 S_i 相关的上下文集合记为 $C_i = \{attribute_1, attribute_2, \dots, attribute_k\}$ 。值得注意的是,上下文信息的提取与具体的数据集有关,根据数据集中可获得的信息,提取上下文信息集合 C , 即同一个数据集中所有的选择 S 都提取相同的上下文信息集合。有向图 $G = (U, \epsilon)$ 表示用户的社交网络,边的集合 ϵ 表示用户的好友关系,用户 U_u 的好友集合记为 $F_u \subset U$ 。

3.2 上下文感知推荐

首先讨论如何利用上下文信息提高推荐精度。

从数据集中提取与用户兴趣相关的上下文属性集合 $attributelist$, $attributelist$ 的选取与具体的数据集相关,不同的数据集对应的上下文属性集合可能不同。为了有效处理不同类型的上下文信息,使用决策树分类算法 C4.5 生成一棵决策树,生成决策树所需的分裂属性集合就是提取的上下文属性集合。例如,其中一个上下文信息 $attribute_r$ 表示在一周中的哪一天用户选择了产品,选择集合就可以按照“周一到周日”或者“工作日和非工作日”进行分割。关于上下文属性的选取及其值的确定在 4.4 小节说明。

构建决策树时,使用信息增益率来选择决策树每层分裂属性。信息增益率为:

$$GainRatio(S,A) = \frac{Gain(S,A)}{SplitInfo(S,A)} \quad (9)$$

其中, $Gain(S,A)$ 为信息增益, $SplitInfo(S,A)$ 为分裂信息,代表了按照属性 A 分裂样本集 S 的广度和均匀性。

$$SplitInfo(S,A) = - \sum_{i=1}^c \frac{|S_i|}{|S|} \times \log_2 \left(\frac{|S_i|}{|S|} \right) \quad (10)$$

其中, S_1 到 S_c 是属性 A 的 c 个不同值分割 S 而形成的 c 个样本子集。

决策树构建完成后,“用户-产品-上下文”选择数据集 S 根据用户选择产品时的上下文属性集合 $attributelist$ 被分割为子集(即图2的叶子节点),由此,有相似上下文信息的选择被分成一组。由于这些子集的生成条件是相似的上下文,因此使用子集预测用户对产品的选择倾向程度并给出推荐比直接使用原始的选择集合进行推荐的精确度高很多。对于要推荐的用户,根据其选择产品时的上下文信息,匹配最相似的子集,利用子集预测用户对产品的选择倾向程度 S_m 。即 S_m 属于决策树中的一个“用户-产品-上下文”选择子集 $S_k \subset S$, 对于每个 S_k , 我们把它分解为用户的潜在特征向量 U_k 和产品的潜在特征向量 V_k , 使用这两个潜在特征向量可以预测用户对产品的选择倾向 S_m , 如式(11)和式(12)所示。

$$\begin{aligned} \arg \min_{U_k, V_k} L &:= \sum_{(i,j) \in O} \ln(1 + \Delta_{ij}^{-1}) + \lambda (\|U_k\|_F^2 + \|V_k\|_F^2) \\ &= \sum_{(i,j) \in O} \ln \left(1 + \frac{1}{N} \sum_{h=1}^N S_{h,i,j} \right) + \lambda (\|U_k\|_F^2 + \|V_k\|_F^2) \end{aligned} \quad (11)$$

$$S_m = (U_k^T)^T \cdot V_k \quad (12)$$

其中, $\frac{1}{N} \sum_{h=1}^N S_{h,i,j}$ 表示用户 i 对子集中全体 N 个产品的平均选择倾向。

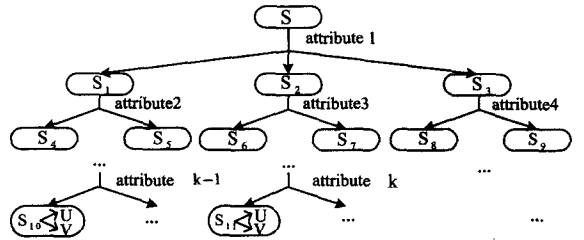


图2 用户-产品选择决策树

将上下文信息提取为 C4.5 算法中的分裂属性集合,构造最优决策树预测用户对产品的选择倾向程度,能够生成更加个性化的精准的推荐。更重要的是,通过移除一部分跟所选上下文相关性较弱的选择倾向而生成的子集远小于原始集合,大大降低了计算复杂度。

3.3 结合社交信息的推荐方法

在之前所提出的上下文感知推荐模型的基础上,结合目标用户的社交信息,进一步提高推荐质量。尽管朋友的意见能为用户提供有效的参考,可是现存的工作却只是直接利用好友关系,不加任何处理,比如,并不深入探讨如何衡量用户与好友之间的相似性^[14]。现实社会中,每个用户都有很多好友,如果考虑用户的所有好友既不现实也没意义,所以要找出与目标用户品味相似的用户,利用这些用户的历史选择信息,对目标用户进行推荐。为了解决上述问题,在式(11)的基础上,引入社会正则化项,见式(13)。

$$\alpha \sum_{i=1}^m \sum_{f \in F_i} sim(i,f) \|U_{k,i} - U_{k,f}\|_F^2 \quad (13)$$

其中, α 是一个常量, 用于控制模型中使用社交信息的程度。 $sim(i, f)$ 表示基于历史用户-项目选择集合计算得出的用户 u_i 和他的好友¹⁾ u_f 之间相似性, $sim(i, f)$ 的值越大, 表明两者喜好越相似。

从式(13)可以看出, 融合社交信息的一个有效方式就是通过喜好相似的好友向目标用户推荐产品, 而这种相似性通过他们过去共同选择过的产品进行计算。由于隐式反馈场景中只有“已选择”与“未选择”两种状态, 即不像显式场景中, 关心用户对产品的具体评分值是多少, 隐式反馈只需要关心用户与产品之间是否有选择关系。所以可以根据用户-产品历史选择集合, 由 Jaccard 系数衡量用户 u_i 和好友 u_f 之间的相似性:

$$sim(i, f) = \frac{V(i) \cap V(f)}{V(i) \cup V(f)} \quad (14)$$

其中, $V(i) \cap V(f)$ 表示用户 u_i 和他的好友 u_f 共同选择过的产品集合, $V(i) \cup V(f)$ 表示用户 u_i 和 u_f 所有选择过的产品集合。

将式(13)和式(14)代入式(11)中, 得到式(15):

$$\arg \min_{U_k^s, V_k^s} L_2 := \sum_{(i,j) \in O} \ln(1 + \frac{1}{N_{h=1}} \sum_{S_h} S_h) + \alpha \sum_{i=1} \sum_{f \in F_j} sim(i, f) \\ \|U_{k,i} - U_{k,f}\|_F^2 + \lambda (\|U_k\|_F^2 + \|V_k\|_F^2) \quad (15)$$

使用随机梯度下降法 SGD 求解 IFCSP 的基本算法。

算法 1 SGD 求解 IFCSP

输入: 可观察选择集合 O , 学习率 γ , 最大迭代次数 T

输出: 潜在特征矩阵 U_k^s, V_k^s

① 随机初始化 U_k^s, V_k^s

② for iter = 1, 2, ..., T do:

③ for each $(i, j) \in O$:

④ 根据式(5)所求梯度更新 $U_{k,i}^s \leftarrow U_{k,i}^s - \gamma \frac{\partial L_2}{\partial U_{k,i}^s}$

根据式(6)所求梯度更新 $V_{k,j}^s \leftarrow V_{k,j}^s - \gamma \frac{\partial L_2}{\partial V_{k,j}^s}$

⑤ end for

⑥ end for

3.4 IFCSP 推荐过程描述

输入: 用户-产品选择集合, 上下文信息集合, 用户的社交信息。

输出: 用户在当前上下文下的推荐产品列表。

① 选取分裂属性 *attributelist*。根据数据集中可获得的上下文信息, 以及这些信息对用户做出选择的影响程度, 应用 ChiMerge 算法选取上下文属性集合作为决策树的分裂属性。

② 生成决策树。根据式(9)计算的信息增益率, 选择决策树每层分裂属性, 利用决策树分类算法 C4.5 构建决策树, 从而根据用户选择产品时的上下文属性集合 *attributelist* 将“用户-产品-上下文”选择数据集分割为子集。由于子集预测目标用户对产品的选择倾向比使用原始的历史选择集合预测的精度更高。

③ 利用子集预测用户对产品的选择倾向程度。对于要推

荐的用户, 根据其选择产品时的上下文信息, 匹配最相似的子集, 根据式(11)和式(12)预测用户对产品的选择倾向。

④ 结合社交信息, 进一步提高推荐精度。对比目标用户及其朋友的历史选择信息, 利用式(14)找出与用户兴趣品味相似的朋友, 在式(11)的基础上, 增加式(13)所示的社会正则化项, 利用这些朋友选择产品的信息改善对目标用户的推荐。

⑤ 生成推荐列表。根据式(15)得到用户的潜在特征向量和产品的潜在特征向量, 再根据式(12)预测用户对产品的选择倾向, 将候选产品按照选择倾向程度高低排序, 并将其推荐给目标用户。

4 实验

4.1 实验数据集描述及实验设置

豆瓣是中国最大的社交平台之一, 提供了书目推荐和以共同兴趣交友等多种服务功能。用户可以对图书、电影、音乐等提供评论, 也可以通过“喜欢”来表达他对某个物品的偏好, 比如想读某本书。豆瓣数据集还提供了用户评论物品的时间日期等相关的上下文信息, 而且用户还能在豆瓣上关注感兴趣的内容和人, 所以豆瓣数据集中包含了用户的社交信息, 因此选择豆瓣数据集²⁾来衡量文中提出的 IFCSP 模型的推荐效果。另外, 由于很多数据集中不包含用户的社交信息, 如 MovieLens, Netflix 等, 因此选择 MovieLens-1M³⁾数据集来验证不考虑社交信息时 IFCSP 模型的推荐效果。本文采用两个数据集, 详细统计信息如表 1 所列。

表 1 数据集描述

			user	item	selection
Douban	Book	training set	6879	135986	712037
		test set	1719	33996	158620
	Movie	training set	4182	38705	936484
		test set	1045	9676	214721
Music	training set	19058	148459	887216	
	test set	4764	37115	166895	
MovieLens-1M	training set	1007	78325	2197822	
	test set	316	9844	723657	

为了拟合真实的推荐场景, 按时间将数据切分为训练集与测试集, 对测试集中的每个用户, 随机抽样 800 个待推荐产品作为候选集, 其中包括用户在测试集中选择的产品以及用户在训练集与测试集中从未选择过的产品。各推荐模型的目标是根据切分时间点之前的隐式反馈信息推测未来的用户选择行为, 将候选产品集根据被用户选择的可能性由高到低进行排序, 生成推荐列表。

4.2 评价指标

采用平均正确率均值 (Mean Average Precision, MAP) 度量算法的推荐精度。MAP 是对所有测试用户的平均正确率 (Average Precision, AP) 的再一次平均。给定一个用户 i 以及他的长度为 M 的推荐列表 $\langle j_1, j_2, \dots, j_M \rangle$, 假设用户 i 选择了其中 N 个, 则可以计算出平均正确率:

¹⁾ 在不同的社交圈中, 好友关系有对称和不对称之分。例如 QQ 好友就是对称的, 而豆瓣的好友则不对称。当好友关系不对称时, 我们此处的好友是指目标用户愿意与之结交的朋友。比如豆瓣中, 用户去关注的人, 而不是指用户的粉丝。

²⁾ 该数据集引用自文献[13]。

³⁾ <http://www.grouplens.org/node/12>。

$$AP_i = \frac{\sum_{k=1}^M \text{precision}(k) \times \text{ref}(k)}{N} \quad (16)$$

其中, $\text{precision}(k)$ 是 Top-K 的命中率, 如果 j_k 命中, 则 $\text{ref}(k)=1$; 否则 $\text{ref}(k)=0$ 。MAP 这一评价指标特别看重推荐列表的前几位是否命中, MAP 越高, 表示算法的推荐准确度越高。

选取平均百分百排序 (Mean Percentage Ranking, MPR) 衡量用户对推荐列表的满意程度。MPR 是一种面向召回率的评价指标, 把为用户 i 提供的已排序推荐列表中产品 j 的排序位置记为 $\text{rank}_{i,j}$, 如果 j 排在推荐列表首位, 则 $\text{rank}_{i,j}=0$; 如果排在末位, 则 $\text{rank}_{i,j}=100\%$ 。通过平均所有被用户选择的产品的排序位置, 可得:

$$\text{MPR} = \frac{1}{|\text{Users}|} \sum_i \frac{\text{ind}(i,j) \times \text{rank}_{i,j}}{\sum_j \text{ind}(i,j)} \quad (17)$$

其中, $|\text{Users}|$ 表示测试集中用户的总数; $\text{ind}(i,j)$ 是指示函数, 如果用户 i 选择了产品 j 则为 1, 否则为 0, MPR 看重预测平均召回率。完全随机的推荐列表的 MPR 值接近 50%。MPR 越低, 表示算法在预测召回率上的表现越好。

4.3 对比算法

为了展示 IFCSPP 算法在推荐性能上的提升, 本文将它同以下方法进行比较。

- 1) 基于用户的协同过滤^[4] (User-based Collaborative Filtering, UBCF): 根据历史选择信息为每个用户计算出相似用户集, 再根据这些相似用户的共同选择行为进行推荐;
- 2) 基于产品的协同过滤^[6] (Item-based Collaborative Filtering, IBCF): 根据历史选择信息为每个产品计算出相似产品集, 再通过用户对这些相似产品的平均选择情况来估计推荐产品被用户选择的可能性;
- 3) 基于大规模隐式反馈的个性化推荐^[2] (IFRM): 将推荐任务转化为选择行为发生概率的优化问题, 关注选择行为本身, 计算用户选择倾向程度, 产生推荐列表;
- 4) 基于隐式反馈的矩阵分解^[5] (iMF): 将可观察的选择设为 1, 未选择设为 0, 并设置一个相对较小的权值, 将传统 0-1 矩阵分解成功应用于隐式反馈推荐场景。

4.4 上下文信息的选取

根据豆瓣及 MovieLens-1M 数据集中可获得的上下文信息, 以及这些信息对用户做出选择的影响程度, 应用 ChiMerge 算法^[20] 选取了 5 个上下文属性集合作为决策树的分裂属性 $\text{attributelist} = \{\text{hour-of-day}, \text{day-of-week}, \text{number-of-like}, \text{size}, \text{category}\}$ 。其中, hour-of-day 表示一天中用户选择产品的时间; day-of-week 表示用户周几选择产品; number-of-like 表示产品的关注度, 产品被用户“喜欢”的次数越多, 意味着该产品受关注度越高, 因而文中选取产品被“喜欢”的次数表示产品的关注度 (MovieLens-1M 数据集不考虑此上下文属性); size 表示用户的活跃程度, 用户选择产品的次数越多, 意味着该用户越活跃, 因而文中选取用户选择产品的数量表示用户活跃程度; category 表示用户偏爱选择什么类型的产品。因为这 5 个属性数值较大, 故运用 ChiMerge 算法^[20] 离散化后, 得出 hour-of-day 分为 5 段, day-of-week 分为 2 段, number-of-like 分为 11 段, size 分为 12 段, category 分为 6 段。

4.5 实验结果与分析

实验 1 参数设置对实验结果的影响

IFCSPP 中需要设定如下参数: 潜在特征维数 K 、社会化信息在模型中占的比重 α 、正则化参数 λ 以及随机梯度下降法的学习速率 γ 。本文使用豆瓣数据集验证不同的参数设定对 IFCSPP 模型的推荐效果的影响。

首先固定迭代次数为 10 次, 由图 3 可以看出, 随着特征维数 K 增大, MAP 小幅度增大, MPR(%) 逐渐减小, 但是当 K 取 30 到 50 时, 模型过分拟合训练数据, 反而对结果产生不良影响。由图 3 可以看出, 推荐效果对特征维数不敏感。特征维数 K 的建议取值为 10~20 之间, 因为优化过程耗时会随着维数的增加而显著增加, 故文中取值为 10。

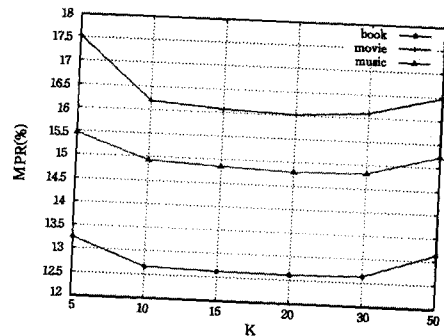
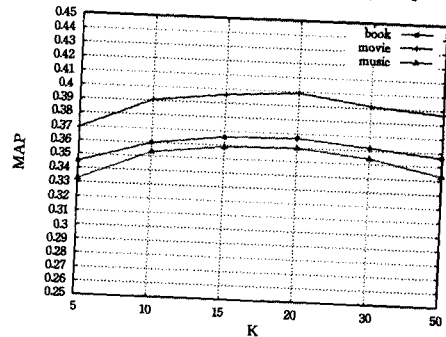


图 3 潜在特征维数 K 的影响 ($\lambda=10^{-5}$, $\gamma=0.01$, $\alpha=0.01$, $T=10$)

由图 4 可以看出, 当将 λ 设置为 0 时, IFCSPP 模型的推荐效果依然较好。这是因为 IFCSPP 模型优化了被选产品与一般产品之间的相对比例, 而不像传统的矩阵分解模型去拟合一个绝对数值, 从而避免了潜在特征矩阵过于复杂, 防止过拟合。正则化参数 λ 应该设置一个较小的值, 文中取值为 10^{-5} 。

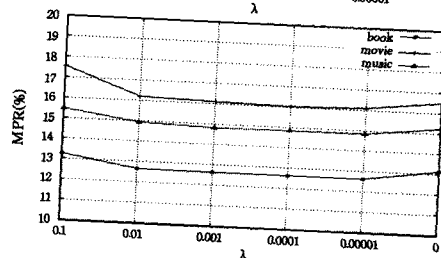
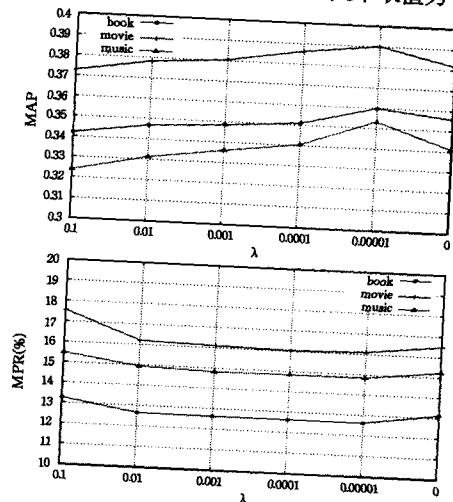


图 4 正则化参数 λ 的影响 ($K=10$, $\gamma=0.01$, $\alpha=0.01$, $T=10$)

由图5可以看出,当 $\alpha=0.01$ 时,MAP值达到最大,当 α 再稍有减小时,MAP的值也随之减小;而MPR(%)先减小,当 $\alpha=0.01$ 时,MPR(%)的值趋于稳定。由此可见, α 取值0.01时,可以平衡并协调用户-产品选择信息和用户的社会化信息,使推荐效果最佳。

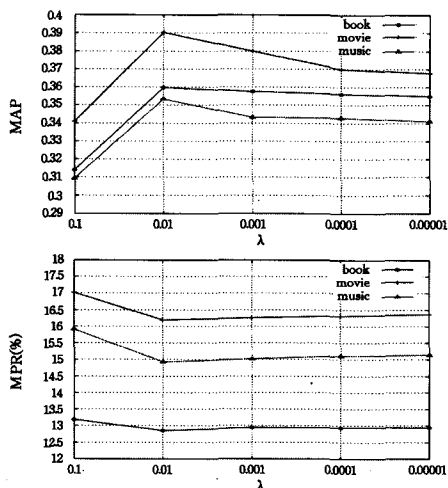


图5 社会化信息在模型中占的比重 α 的影响($K=10, \lambda=10^{-5}, \gamma=0.01, T=10$)

对于随机梯度下降法SGD来说,学习速率 γ 是一个重要参数,显著影响收敛速度以及优化效果。由图6可以看出,将 γ 设置为0.01左右较为合适。若将 γ 设置过大如0.05,则会导致模型无法收敛,经过4次迭代推荐效果就已经达到顶峰而无法继续提升,并且在经过第10次迭代之后,推荐效果有明显的下降趋势;过小的学习率会导致收敛速度过慢,当将 γ 设置为0.001时,需要经过30次迭代(图6中只画出了前12次)才可以达到理想的推荐效果。

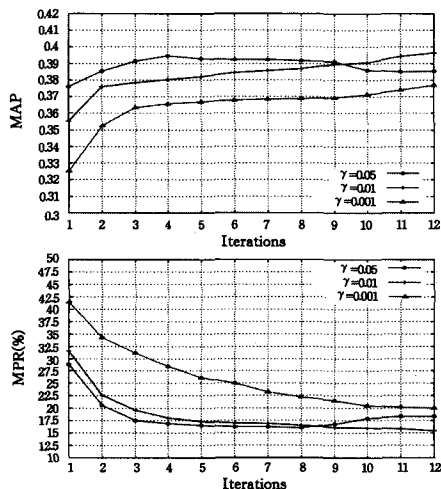


图6 学习速率 γ 的影响($K=10, \lambda=10^{-5}, \alpha=0.01, T=10$)

实验2 IFCSP与其他算法的推荐效果比较

该实验通过将IFCSP与4种算法进行对比来证明IFCSP算法的有效性。为了获得更可靠的实验结果,我们将实验重复5次,每次为目标用户重新随机生成800个候选产品。由图7可以看出,IFCSP和IFRM在MAP评价指标上表现较好,但是由于IFCSP在IFRM的基础上考虑了上下文信息和用户社交信息,因此其比IFRM有更大优势,推荐准确率更高,这说明IFCSP模型可以显著提高Top-N推荐的命中率。

在MPR评价指标上,各个算法在MovieLens-1M数据集中表现出了一致的优越性。在豆瓣数据集中,与传统协同过滤方法相比,IFCSP取得了显著的效果,与IFRM和iMF相比,IFCSP在预测召回率上的表现较好。

在实际应用中,Top-N推荐命中率尤其重要,因为用户不可能耐心浏览完所有推荐产品。与iMF以及IFRM相比,IFCSP可以产生更高质量的推荐,因为它不仅直接对用户选择行为进行建模,而且充分利用用户的社会化信息以及上下文信息,从而显著提高了预测准确度。

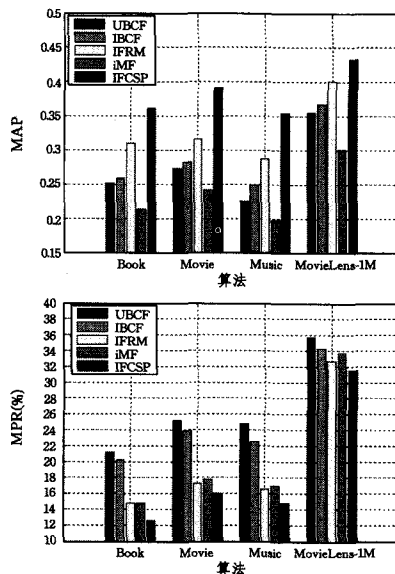


图7 IFCSP与对比算法的推荐效果比较

结束语 针对如何在隐式反馈场景下提高个性化推荐的精度和用户满意度问题,本文提出了一种新颖的基于隐式反馈的推荐模型IFCSP,将上下文信息引入推荐系统,并利用用户社交信息进一步改善对目标用户的推荐结果。实验表明,利用IFCSP模型为用户提供个性化的推荐,适用范围更广泛,推荐精度更高。然而IFCSP模型通过分割“用户-产品-上下文”选择集合,生成的决策树子集中可能没有目标用户或者目标产品的部分历史选择信息,从而导致冷启动问题。这个问题可以通过以下方法解决:从原始选择集合中引入一小部分跟要预测的选择的上下文类似的历史用户选择信息,与当前要预测的选择的上下文信息最相似的历史选择信息优先选择。但如果目标用户或者目标项目是全新的,没有任何历史选择信息,这就需要设计专门的算法解决冷启动问题,这将是我们下一步的研究工作。

参考文献

- [1] Xu Hai-ling, Wu Xiao, Li Xiao-dong, et al. Comparison study of internet recommendation system[J]. Journal of Software, 2009, 20(2): 350-362 (in Chinese)
许海玲, 吴潇, 李晓东, 等. 互联网推荐系统比较研究[J]. 软件学报, 2009, 20(2): 350-362
- [2] Yin Jian, Wang Zhi-sheng, Li Qi, et al. Personalized recommendation based on large-scale implicit feedback[J]. Journal of Software, 2014, 25(9): 1953-1966 (in Chinese)
印鉴, 王智圣, 李琪, 等. 基于大规模隐式反馈的个性化推荐[J]. 软件学报, 2014, 25(9): 1953-1966

- [4] Bennett K P, Demiriz A. Semi-supervised support vector machines[M]. Cambridge: MIT Press, 1998; 10-20
- [5] Zhang X D, Ma J. A general cubic spline smooth semi-supervised support vector machine[J]. Chinese Journal of Engineering, 2015, 37(3): 386-389 (in Chinese)
张晓丹, 马菁. 一个广义三次样条光滑半监督支持向量机[J]. 工程科学学报, 2015, 37(3): 386-389
- [6] Chapelle O, Zien A. Semi-supervised classification by low density separation[C]//10th International Workshop on Artificial Intelligence and Statistics. Barbados, 2005; 57-68
- [7] Liu Y Q, Lin S Y. Polynomial Smooth Semi-supervised Support Vector Classifier[J]. Computer Science, 2009, 36(7): 179-181 (in Chinese)
刘叶青, 刘三阳. 一种多项式光滑的半监督支持向量机[J]. 计算机科学, 2009, 36(7): 179-181
- [8] Bennett K, Demiriz A. Semi-supervised support vector machines [C]//Proceedings of the 1998 Conference Advances in Neural Information Processing Systems. 1998; 368-374
- [9] Joachims T. Transductive inference for text classification using support vector machines[C]//International Conference on Machine Learning, Pittsburgh. 1999; 200-209
- [10] Dijkstra E W. A note on two problems in connection with graphs [J]. Numerische Mathematics, 1959, 1(1): 269-271
- [11] Cox T F, Cox M A. Multidimensional Scaling[J]. Journal of the Royal Statistical Society Series A(Statistics in Society), 1994, 45(2): 271-281
- [12] Bengio Y, Grandvalet Y. Semi-supervised learning by entropy minimization[J]. Neural Information Processing Systems, 2005, 5(2): 529-536
- [13] Nene S A, Nayar S K, Murase H. Columbia object image library (coil-20) [R]. CUCS-005-96, Columbia Univ., USA, February 1996

(上接第 253 页)

- [3] Liu X, Aberer K. SoCo: A social network aided context-aware recommender system[C] //Proc of the 22nd Int Conf on World Wide Web. Brazil, IW3C2, 2013; 781-802
- [4] Herlocker J, Konstan J, Borchers A, et al. An algorithmic framework for performing collaborative filtering [C] // Proc of the 22nd Annual Int ACM SIGIR Conf on Research and Development in Information Retrieval. New York; ACM, 1999; 230-237
- [5] Hu Y, Koren Y, Volinsky C. Collaborative filtering for implicit feedback datasets[C]//Proc of the 8th IEEE Int Conf on Data Mining. Pisa, Italy: ICDM, 2008; 263-272
- [6] Sarwar B, Karypis G, Konstan J, et al. Item-Based collaborative filtering recommendation algorithms[C]//Proc of the 10th Int Conf on World Wide Web. Hong Kong, China; ACM Press, 2001; 285-295
- [7] Yang Xing-yao, Yu Jiong, Ibrahim T, et al. Collaborative filtering model fusing singularity and diffusion process [J]. Journal of Software, 2013, 24(8): 1868-1884 (in Chinese)
杨兴耀, 于炯, 吐尔根·依布拉音, 等. 融合奇异性和扩散过程的协同过滤模型[J]. 软件学报, 2013, 24(8): 1868-1884
- [8] Adomavicius G, Tuzhilin A. Context-Aware recommender systems[M]. Recommender Systems Handbook. Berlin: Springer-Verlag, 2011; 217-253
- [9] Adomavicius G, Tuzhilin A. Context-Aware recommender systems[C]//Proc of the RecSys 2008. New York: ACM Press, 2008; 335-336
- [10] Adomavicius G, Ricci F. Workshop on context-aware recommender systems[C]//Proc of the RecSys 2009 (CARS 2009). New York; ACM Press, 2009; 423-424
- [11] Karatzoglou A, Amatriain X, Baltrunas L, et al. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering[C]//Proc of the fourth ACM Conference on Recommender systems. Barcelona, Spain; ACM, 2010; 74-86
- [12] Rendle S, Gantner Z, Freudenthaler C, et al. Fast context-aware recommendations with factorization machines[C]//Proc of the 34th Int ACM SIGIR Conference on Research and Development In Information Retrieval. Beijing, China; SIGIR, 2011; 635-644
- [13] Zhong E, Fan W, Wang L, et al. Comsoc: adaptive transfer of user behaviors over composite social network[C]//Proc of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2012; 696-704
- [14] Liu F, Lee H. Use of social network information to enhance collaborative filtering performance [J]. Expert Systems with Applications, 2010, 37(7): 4772-4778
- [15] Zhou K, Yang S, Zha H. Functional matrix factorizations for cold-start recommendation[C]//Proc of the 34th Int ACM SIGIR Conference on Research and Development In Information Retrieval. Beijing, China; SIGIR, 2011; 69-78
- [16] Zhang M, Tang J, Zhang X, et al. Addressing cold start in recommender systems: A semi-supervised co-training algorithm [C]//Proc of the 37th Int ACM SIGIR Conference on Research & Development in Information Retrieval. New York; ACM, 2014; 73-82
- [17] Jiang Sheng, Wang Zhong-qun, Xiu Yu, et al. Collaborative Filtering Recommendation Method Based on Dynamic Social Behavior and Users' Background Information [J]. Computer Science, 2015, 42(3): 252-255, 265 (in Chinese)
蒋胜, 王忠群, 修宇, 等. 基于动态社会行为和用户背景的协同推荐方法[J]. 计算机科学, 2015, 42(3): 252-255, 265
- [18] Wang Peng, Wang Jing-jing, Yu Neng-hai. A kernel and user-based collaborative filtering recommendation algorithm [J]. Journal of Computer Research and Development, 2013, 50(7): 1444-1451 (in Chinese)
王鹏, 王晶晶, 俞能海. 基于核方法的 User-Based 协同过滤推荐算法[J]. 计算机研究与发展, 2013, 50(7): 1444-1451
- [19] Hu Xun, Meng Xiang-wu, Zhang Yu-jie, et al. Recommendation algorithm combing item features and trust relationship of mobile users [J]. Journal of Software, 2014, 24(8): 1817-1830 (in Chinese)
胡勋, 孟祥武, 张玉洁, 等. 一种融合项目特征和移动用户信任关系的推荐算法[J]. 软件学报, 2014, 25(8): 1817-1830
- [20] Liu H, Setiono R. Chi2: Feature selection and discretization of numeric attributes[C]//ICTAI. 1995; 388-391
- [21] Yang X, Steck H, Liu Y. Circle-based recommendation in online social networks[C]//Proc of the 18th ACM SIGKDD Int Conference on Knowledge Discovery and Data Mining. 2012; 1267-1275