计算机科学
COMPUTER SCIENCE

基于用户覆盖及评分差异的多样性推荐算法

陈壮，邹海涛，郑尚，于化龙，高尚

引用本文

陈壮，邹海涛，郑尚，于化龙，高尚. 基于用户覆盖及评分差异的多样性推荐算法[J]. 计算机科学, 2022, 49(5): 159-164.

CHEN Zhuang, ZOU Hai-tao, ZHENG Shang, YU Hua-long, GAO Shang. Diversity Recommendation Algorithm Based on User Coverage and Rating Differences[J]. Computer Science, 2022, 49(5): 159-164.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210300263

基于用户覆盖及评分差异的多样性推荐算法

陈 壮 邹海涛 郑 尚 于化龙 高 尚
江苏科技大学计算机学院 江苏 镇江 212003
(just_chenzhuang@163.com)

摘 要 传统的推荐算法着重于提高推荐结果的精确度,对于推荐列表的多样性则有所忽略。但很多研究表明,用户对产品的多样性需求也是影响用户体验的重要因素之一。针对该问题,在用户覆盖定义的基础上,提出了一个基于产品评分差异的用户覆盖模型。在生成用户兴趣域(用户覆盖)的过程中,该模型一方面通过构建评分差异矩阵,将不同用户对同一产品评分上的差异与用户覆盖模型有效地结合起来,从而计算得到更精准的用户兴趣域;另一方面,该模型将用户和推荐列表的兴趣域分别映射到两个 m 维向量上(分别称为用户向量和产品集向量),并将推荐的目标函数向量化,有效地减少了计算过程中迭代的次数。此外,通过探讨用户向量和产品集向量之间的相似度关系,提出了一种新的推荐列表选取策略。基于该方法构建得到的模型可以在一定程度上兼顾推荐精确度与多样性。由于根据用户向量构建匹配的产品集向量是一个 NP-hard 问题,因此使用贪心算法来求解该问题,贪心算法的有界性有着严谨的理论依据。在两个真实的数据集上进行实验,结果表明,与多样性推荐方面相关的先进算法相比,所提算法具有一定的优势。

关键词:推荐系统;多样性;用户覆盖;评分差异;相似度
中图法分类号 TP181

Diversity Recommendation Algorithm Based on User Coverage and Rating Differences

CHEN Zhuang,ZOU Hai-tao,ZHENG Shang,YU Hua-long and GAO Shang
College of Computer,Jiangsu University of Science and Technology,Zhenjiang,Jiangsu 212003,China

Abstract Traditional recommender systems usually focus on improving recommendation accuracy while neglecting the diversity of recommendation lists. However,several studies have shown that,users' diversity needs also take an important part of their satisfaction. In this paper,a user-coverage model based on item rating differences is proposed. During generating user's interest domain(user coverage),on the one hand,the model combines rating differences between users across an item with user-coverage model effectively,thus obtaining a more precise interest domain of the user. On the other hand,objective function is constructed in the form of vector by mapping a user's and an itemset's interest domain to two m-dimensional vectors (called user vector and itemset vector respectively),which can reduce the number of iterations in the calculation process. In addition,a new items selection strategy is provided by similarity relationship between those two m-dimensional vectors. The proposed model has superior performance in both accuracy and diversity. User vector for a specific user is a constant,however,finding the most matching itemset vector will be an NP-hard problem. During the implementation of the proposed model,a greedy algorithm is chosen to solve the optimization problem based on critical theoretical boundary. Experimental comparisons with the state-of-the-arts related to diversity recommendation in recent years on two real-world data sets demonstrate that the proposed algorithm can effectively improve the diversity of the recommendation.

Keywords Recommender systems,Diversity,User coverage,Rating differences,Similarity

1 引言

为了更好地迎合用户偏好,传统推荐算法往往只关注推荐结果的精确度,算法的主要目的是提高准确率、召回率等精确度指标。Nahta 等[1]将推荐系统与深度学习相结合,提出了一种广义推荐模型-元嵌入深度协同过滤算法,提高了推荐的精确度。Xu 等[2]以及 He 等[3]通过考虑用户在不同产品上的预测评分与实际评分之间的差值,提出了一种高阶评分距离模型,有效地降低了预测评分与实际评分的均方误差。Zhang 等[4]为了更准确地度量社交关系对推荐预测的影响,提出了一种基于领域信任及不信任的社会化奇异值分解推荐算法。然而,高精确度的推荐并不一定意味着用户满意度较高[5-6]。Cheng 等[7]也通过实验证明,精确度导向的推荐系统往往给用户推荐高相似度的产品,这会减少用户接触到的产品类别数目,不利于销量的增长。因此,以推荐精确度为基础,多样性导向的推荐系统越来越受到重视。

多样性导向的推荐系统根据不同的目标函数大体上可以分为两大类。第一类是平衡模型[8-10],这类模型分别使用

到稿日期:2021-03-25 返修日期:2021-08-07
通信作者:邹海涛 (zouhaitao@126.com)

不同的函数去量化地定义精确度和多样性，在目标函数中使用参数来平衡两者的关系。Sha 等[8]将多样性转化为向量空间上产品之间的欧氏距离。这种类型的模型所在的向量空间大都是基于矩阵分解算法[11]得到的低维矩阵。但由于评分矩阵的稀疏性，矩阵分解算法得到的低维产品矩阵具有较大的不确定性，使得这类模型的表现往往差强人意。Liu 等[10]同时考虑近邻用户集合对目标用户的兴趣覆盖比例和相似程度，将这两项整合到目标函数中，并使用参数 α 和 β 进行平衡。但是，对于不同的训练集，该模型需要重新训练参数 α 和 β，模型的适应性差。第二类则是同时考虑精确度和多样性的统一模型[12-13]。Shameem 等[12]在相似图上使用集合覆盖的概念，通过最大化列表内项目的差异和最大化外部相似构建了目标函数。由于该模型需要计算系统中每对产品之间的相似度，因此系统开销大。He 等[13]基于用户覆盖的概念，通过最大化覆盖用户的兴趣域，提出了用户覆盖模型。但用户覆盖模型在计算用户兴趣点频率时忽略了不同用户对同一产品评分的差异，导致计算得到的用户覆盖不够精准。

基于文献[13]的研究，本文在用户覆盖模型的基础上，通过构建评分差异矩阵，将用户对同一产品评分的差异与用户对该产品的兴趣点频率相结合。将推荐任务向量化，最终得到更为精准高效的基于评分差异的用户覆盖模型。此外，为了更好地支持多样化推荐指标，在将目标函数向量化后，本文通过用户向量和产品集向量之间相似度的关系，提出了一个新的推荐列表选取策略。

2 相关工作

本文提出的多样性推荐算法的原型是基于课题组已研发的用户覆盖模型[13]。在推荐过程中，用户覆盖模型通过最大化覆盖用户的兴趣域，来达到兼顾相关性和多样性的目的。

假设推荐系统中包含 m 个用户、n 个产品，其分别属于集合 U 和集合 I。用户评分矩阵 $\boldsymbol{R}$ 为 $m\times n$ 维，其中 $r_{ui}\in\boldsymbol{R}$，表示用户 u 对产品 i 的评分值，该值越大说明 u 对 i 的接受程度越高。若用户 u 没有对产品 i 进行评价，则 r_{ui} 为空。推荐系统的最终目的是通过构建目标函数来预测用户对候选产品的满意度，进而返回精确度和多样性高的推荐列表。

2.1 用户覆盖模型

用户覆盖模型[13]（为了简便，本文称其为 EGA 模型）。基于用户覆盖的概念实现了最大化覆盖用户的兴趣域，并且由于模型无参、高效的特点，在实验仿真中表现优秀，灵活性较好。其核心概念的定义如下。

给定产品 i，其用户覆盖定义如式(1)所示：

$$C(i)=\{u\mid r_{ui}\in\mathbb{R}\} \tag{1}$$

其中，$\mathbb{R}$ 为 $\boldsymbol{R}$ 中非零评分值组成的集合。

产品集合 S 的用户覆盖定义如式(2)所示：

$$C(S)=\bigcup_{i\in S}C(i) \tag{2}$$

对于任意用户 u，其兴趣点 P_u 定义为 $P_u=\{i\mid r_{ui}\in\mathbb{R}\}$，则 u 的用户覆盖定义如式(3)所示：

$$C(P_u)=\bigcup_{i\in P_u}C(i) \tag{3}$$

因此，对目标用户 u 进行推荐时，需优化式(4)。

$$\arg\max_{S\subseteq\Omega}\frac{|C(S)\bigcap C(P_u)|}{|C(P_u)|} \tag{4}$$

其中，Ω 为候选产品集合，S 为最终推荐列表。

由于不同用户对 $C(P_u)$ 起着不同的作用，可使用兴趣点频率的概念去量化不同用户对 $C(P_u)$ 所起作用的大小。用户 v 对 u 的兴趣点频率的定义如式(5)所示：

$$Freq(v)=\sum_{i\in P_u}H(v,C(i)) \tag{5}$$

其意义是量化 v 对 $C(P_u)$ 构建所起的作用。

基于式(5)，将在 $C(P_u)$ 构建过程中起相同作用的用户划分到同一层，且假设 $C(P_u)$ 被分为 $1\sim top$ 层。其中，l 层上的用户集合表示为 L_l，其权重值为 w_l（$w_l=\sqrt{Freq(v)}$，$v\in L_l$），那么式(4)将被改写为：

$$\arg\max_{S\subseteq\Omega}\sum_{l=1}^{top}w_l\frac{|C(S)\bigcap L_l|}{|L_l|} \tag{6}$$

2.2 贪心理论

对于一些 NP-hard 问题，很难在多项式时间内得到问题的最优解，而使用贪心思想求取问题的近似解也成了常用的手段[14]。贪心算法在求解问题时，不从整体上考虑问题的最优解，而是从当前情况出发，做出最好的选择。其得到的结果是整体的近似最优解，相比使用穷举法去找寻最优解，贪心的策略可以节省大量时间。在推荐系统领域，其一般步骤如下：

(1)构建推荐的目标函数 F。

(2)初始化候选产品列 Ω、推荐产品数 K 及推荐列表 S。

(3)遍历 Ω，选取本次循环中使 F 值达到最大的产品 i（产品 i 则为本次循环的最优解）。将 i 加入到 S 中，并从候选列表 Ω 中将其移除。

(4)重复 K 次步骤(3)，返回长度为 K 的推荐列表 S。

2.3 用户覆盖模型的不足之处

用户覆盖模型在计算用户 v 对用户 u 的兴趣点频率时，仅仅从集合角度统计了用户 v 出现在 $C(i)(i\in P_u)$ 中的次数，而忽略了 u 和 v 在同一产品评分上的差异，这使计算得到的用户覆盖向量不足以准确地衡量 v 对 $C(P_u)$ 构建所起作用的大小；另外，用户覆盖模型没有探讨用户向量和产品集向量之间的相似度关系对推荐结果的影响，这些都使得推荐结果的精确度和多样性受到影响。

3 问题陈述

本文要解决的问题来自以下两个方面：

(1)如何改进计算用户兴趣点频率的函数，使其可以将 u 和 v 在同一产品评分上的差异与式(5)有效地结合。

(2)基于用户覆盖的概念，如何合理地构建用户向量和产品集向量，并利用这两个向量之间的相似度关系构建推荐列表。

4 基于评分差异的用户覆盖模型设计

为解决因忽略用户对产品的评分差异而带来的精确度和多样性损失的问题，在计算用户兴趣点频率时，本文将用户评分的差异考虑在内，提出并构建了基于评分差异的用户覆盖模型，其具体设计与相关公式的描述如下。

4.1 基于评分差异的兴趣点频率

为了得到更加精准的用户覆盖向量，本文将重新定义兴趣点频率，把目标用户 u 和其他用户 v 针对 P_u 中所有产品的评分差异考虑在内。使用式(7)替换式(5)。

$$Freq(v)=\sum_{i\in P_u}H(v,C(i))\frac{1}{|r_{ui}-r_{vi}|+1} \tag{7}$$

其中，H 为指示函数，若 $v\in C(i)$，则 $H=1$，否则 $H=0$。

从式(7)可以看出：如果 u 和 v 对 P_u 中所有产品的评分差异越小，且 v 出现在集合 $C(i)(i\in P_u)$ 中的次数越多，那么在构建 $C(P_u)$ 的过程中，用户 v 起到的作用就越大，即 $Freq(v)$ 的值越大。

基于评分差异计算得到的用户兴趣点频率大多数为小数，使得很少会有用户对 $C(P_u)$ 构建起着相同比重的作用。因此，本文不再对用户覆盖进行分层操作，而是为组成 $C(P_u)$ 的每个用户设置对应权重。对于用户 $v\in C(P_u)$，其权重值 w_v 定义如式(8)所示：

$$w_v=\frac{Freq(v)}{\sqrt{Freq(v)}\times\sqrt{|P_u|}} \tag{8}$$

对于组成 $C(P_u)$ 的用户来说，$\sqrt{|P_u|}$ 为公共项，若将权重值修改为式(9)的形式，并不会对最终的推荐结果产生影响。

$$w_v=\sqrt{Freq(v)} \tag{9}$$

最终，推荐任务的目标函数可以改写为：

$$\arg\max_{S\subseteq\Omega}\sum_{v\in C(P_u)}w_vH(v,C(S)) \tag{10}$$

4.2　目标函数的向量法表示

4.1 节主要从集合的角度对模型进行表述。但在实际运算过程中，集合运算繁琐，迭代频繁。为了减少计算量，精简计算步骤，本节将给出模型的向量法表示。

假设待推荐用户为 u，其兴趣点为 P_u。首先构建一个 $m\times n$ 的矩阵 $\boldsymbol{D}$，用于存储不同用户对同一产品的评分差异。对于 $\forall v\in C(i)$，$d_{vi}\in\boldsymbol{D}$，$d_{vi}=\frac{1}{|r_{ui}-r_{vi}|+1}$，表示用户 v 和 u 在产品 i 上的评分值的接近程度。d_{vi} 越大，v 和 u 在产品 i 上的评分就越接近。若 $d_{vi}=0$，则表示 v 和 u 没有同时对产品 i 进行评分，即 $r_{ui}=0$ 或者 $r_{vi}=0$。

得到评分差异矩阵 $\boldsymbol{D}$ 后，u 的用户覆盖向量可以使用式(11)计算得到：

$$C_P=\sqrt{\sum_{i\in P_u}\boldsymbol{D}_i} \tag{11}$$

其中，$\boldsymbol{D}_i$ 为矩阵 $\boldsymbol{D}$ 的第 i 列，其构建及计算演示过程如图1(a)所示。

对于产品集合 S，其用户覆盖向量如式(12)所示：

$$C_S=\bigvee_{i\in S}Binary(\boldsymbol{R}_i) \tag{12}$$

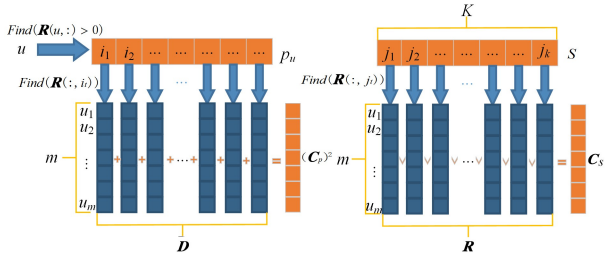
其中，符号“$\vee$”为向量上的按位或运算，$\boldsymbol{R}_i$ 为评分矩阵 $\boldsymbol{R}$ 的第 i 列。$Binary(\boldsymbol{R}_i)$ 将向量 $\boldsymbol{R}_i$ 二值化。若 $\boldsymbol{R}_i[j]>0$，则 $\boldsymbol{R}_i[j]=1$，否则 $\boldsymbol{R}_i[j]=0$。其构建及计算演示过程如图1(b)所示。

从上述定义不难看出，$\boldsymbol{C}_P$ 和 $\boldsymbol{C}_S$ 均为 m 维向量。$\boldsymbol{C}_P$ 中的每个元素代表着对应用户对 $\boldsymbol{C}(P_u)$ 所起作用的大小。$\boldsymbol{C}_S$ 为二值向量，只包含 0 和 1 两种值，且向量 $\boldsymbol{C}_S$ 的值不为 0 位置所对应的用户组成了产品集合 S 的用户覆盖，即 $\boldsymbol{C}(S)$。本文约定向量 $\boldsymbol{C}_S$ 中第 j 个元素使用 $\boldsymbol{C}_S[j]$ 进行表示，类似的约定也适用于其他向量。

最终，式(10)的向量法表示如下：

$$\arg\max_{S\subseteq\Omega}Dot(\boldsymbol{C}_P,\boldsymbol{C}_S) \tag{13}$$

其中，$Dot(\boldsymbol{C}_P,\boldsymbol{C}_S)$ 返回向量 $\boldsymbol{C}_P$ 和 $\boldsymbol{C}_S$ 的内积。



(a) 用户向量构建及计算过程　　(b)产品集向量构建及计算过程

图1　向量构建及计算过程

Fig. 1　Vector construction and calculation process

4.3　使用贪心算法求解基于产品评分差异的用户覆盖模型

从 n 个产品中找到最优的长度为 K 的推荐列表 S，显然这是一个 NP-hard 问题[15]。本文使用贪心算法近似地解决这个问题，具体流程如算法 1 所示。

算法 1　使用贪心算法求解基于产品评分差异的用户覆盖模型

Input：I，R，u，K

Output：S

1. $P_u=Find(\mathbf{R}(u,:)>0)$
2. $\mathbf{D}=Cal(\mathbf{R},P_u)$
3. $\mathbf{C}_P=\sqrt{\sum_{i\in P_u}\mathbf{D}(:,i)}$
4. $S=\emptyset,\Omega=I\backslash P_u$
5. while $|S|<K$ do
6. $\quad i^*\leftarrow\arg\max_{i\in\Omega}Dot(\mathbf{C}_P,\bigvee_{j\in S\cup\{i\}}Binary(\mathbf{R}_j))$
7. $\quad S=S\cup\{i^*\}$
8. $\quad\Omega=\Omega\backslash\{i^*\}$
9. end while
10. return S

通过计算得到用户 u 的兴趣点 P_u，计算系统中其余相关用户与 u 在 P_u 中的每个产品评分之间的差异，并将计算得到的结果存储在矩阵 $\boldsymbol{D}$ 中(算法 1 中的第 1－3 行)；接着遍历候选列表 Ω 中的每个产品 i，在每一次循环中选取使目标函数值达到最大的产品 i，并将其加入到推荐列表 S 中，并从候选列表 Ω 中去掉该产品(算法 1 中的第 5－9 行)。经过 K 次循环，返回长度为 K 的推荐列表 S。

接下来，本文将基于 *sub-modular*[16] 函数的定义及其性质来证明贪心算法的合理性。

定义 1　$\forall S,N$ 为有限集，若 $f(S)+f(N)\geqslant f(S\cup N)+f(S\cap N)$，则函数 f 为 *sub-modular*。

定理 1　在 *sub-modular* 函数中，若 $\hat{S}$ 为使用贪心算法得到的解，S^* 为算法的最优解，则：

$$\frac{f(\hat{S})}{f(S^*)}\geqslant1-\left(1-\frac{1}{K}\right)^K\geqslant1-\frac{1}{e} \tag{14}$$

其中，$K=|S^*|=|\hat{S}|$。

该定理的具体证明细节可参见文献[13]。由 *sub-modular* 函数的定义可知，式(13)中的目标函数 $Dot(\boldsymbol{C}_p,\boldsymbol{C}_S)$ 也为 *sub-modular* 函数，因此，当 K 足够大时，使用贪心算法求解 $Dot(\boldsymbol{C}_p,\boldsymbol{C}_S)$ 将逼近与该函数的最优解。

为了便于描述，后文将该模型简称为 RGA 模型。

4.4 基于用户向量和产品集向量相似度的用户覆盖模型设计

本文在用户覆盖的基础上，基于用户向量和产品集向量之间的相似度关系，构建了一个基于相似度的用户覆盖模型。同样地，为了方便表示，后文用 SGA 表示该模型。

SGA 模型使用的概念和定义大体上与 RGA 模型相同。不同的是，在 SGA 模型中，需要重新定义推荐列表 S 的用户覆盖向量，其定义如式(15)所示：

$$C_S=\sqrt{\sum_{i\in S}\boldsymbol{D}_i} \tag{15}$$

其中，$\boldsymbol{D}$ 为评分差异矩阵。

由于用户向量 $\boldsymbol{C}_P$ 和产品集向量 $\boldsymbol{C}_S$ 在一定程度上反映了 u 和 S 的兴趣点分布，因此，如果 $\boldsymbol{C}_P$ 和 $\boldsymbol{C}_S$ 在其所处的 m 维向量空间中比较相似，那么说明 u 和 S 的兴趣点分布较为相似，即 S 中的产品被目标用户 u 接受的可能性较大。基于上述假设，SGA 模型将构建式(16)所示的目标函数。

$$\arg\max_{S\subseteq\Omega} sim(\boldsymbol{C}_P,\boldsymbol{C}_S) \tag{16}$$

其中，$\boldsymbol{C}_P$ 和 $\boldsymbol{C}_S$ 分别由式(11)和式(15)计算得到，而两者的相似度则由式(17)计算得到。

$$sim(\boldsymbol{C}_P,\boldsymbol{C}_S)=\frac{\boldsymbol{C}_P\cdot\boldsymbol{C}_S}{\|\boldsymbol{C}_P\|\times\|\boldsymbol{C}_S\|} \tag{17}$$

其中，$\|\cdot\|$ 为向量的二范数。

为用户向量找到最相似的产品集向量仍是一个 NP-hard 问题，本文依旧使用贪心算法去解决这个问题。算法流程如算法 2 所示。

算法 2 使用贪心算法求解基于相似度的用户覆盖模型

Input: I, R, u, K
Output: S
1. P_u = Find($\mathbf{R}$(u, :) > 0)
2. $\mathbf{D}$ = Cal($\mathbf{R}$, P_u)
3. $\mathbf{C}_P=\sqrt{\sum_{i\in P_u}\mathbf{D}(:,i)}$
4. S = Ø, Ω = I\P_u
5. while |S| < K do
6. max = 0, indice = 0
7. for i in Ω do
8. $C_s=\sqrt{\sum_{j\in S\cup\{i\}}D_j}$
9. temp_max = sim ($\mathbf{C}_P$, $\mathbf{C}_S$)
10. if temp_max > max do
11. max = temp_max
12. indice = i
13. S = S∪{indice}
14. Ω = Ω\{indice}
15. end while
16. return S

5 实验结果及分析

5.1 数据集描述

本文涉及的所有实验均在运行 Microsoft Windows10 的四核 CPU (Intel i5, 1.8 GHz)、8.0 GB 内存和 512 GB 硬盘的 PC 上实现，主要从算法的多样性、精确性两方面对本文提出的 RGA，SGA 模型进行考查。实验中选定了两个稳定的数据集 MovieLens1M[1] 和 MovieLens-Latest-Small 进行测试。这两个数据集都包含用户对不同产品的评分信息和产品所属类别信息。每个用户的评分个数不少于 20，且用户对产品的评分值为 1～5 之间的整数。数据集中的每个产品都至少属于某个类别且可以属于多个类别。在实验中，随机地对数据集进行划分，其中训练集占 50%，剩余的 50%作为测试集。数据集的详细信息如表 1 所列(由于受篇幅限制，此处数据集名称做了简写，用 1M 表示 MovieLens1M，Latest-Small 代表 MovieLens-Latest-Small)。

表 1 数据集信息描述

Table 1 Datasets description

数据集	用户数	电影数	评分数	类别数	稀疏度/%
1M	6040	3952	1000209	18	4.19
Latest-Small	610	9742	100836	18	1.69

5.2 测试指标

本文将在精度(Precision)、类别覆盖(Genre Coverage, GC)、测试集类别覆盖(Test Genre Coverage, TGC)、列表内产品平均距离[8](Intra-List Distance, ILD)、累积折损信息增益(Discounted Cumulative Gain, DCG)和兴趣域覆盖比率[13](Proportion of Interest Domain, PID)这 6 个指标上进行对比实验。其中，Precision 和 DCG 为精确度指标，其余 4 个为多样性指标。

Precision 的定义如下：

$$Precision=\frac{1}{|U|}\sum_{u\in U}\frac{|S_u\cap T_u|}{|T_u|} \tag{18}$$

其中，U 为系统中所有用户所在的集合；S_u 为呈现给用户的推荐列表；T_u 为用户在测试集上交互过的产品集。

GC(TGC)指标测量的是推荐列表中产品的类别对用户在训练集(测试集)中交互过的产品类别的覆盖比率，其定义如式(19)、式(20)所示：

$$GC=\frac{1}{|U|}\sum_{u\in U}\frac{|\bigcup_{i\in S_u}genres(i)\cap\bigcup_{i\in P_u}genres(i)|}{|\bigcup_{i\in P_u}genres(i)|} \tag{19}$$

$$TGC=\frac{1}{|U|}\sum_{u\in U}\frac{|\bigcup_{i\in S_u}genres(i)\cap\bigcup_{i\in T_u}genres(i)|}{|\bigcup_{i\in T_u}genres(i)|} \tag{20}$$

其中，$genres(i)$ 表示得到产品 i 所属的类别集合。

ILD 测量的是推荐列表中产品之间的平均距离，ILD 值越大，说明推荐列表中的产品在其所处的向量空间中的距离越远，产品之间也就越不相似，从而说明推荐列表的多样性程度高。其定义如式(21)所示：

$$ILD=\frac{1}{|U|}\sum_{u\in U}\frac{1}{K(K-1)}\sum_{i\in S_u}\sum_{j\in S_u,j\neq i}dist(i,j) \tag{21}$$

其中，$K=|S_u|$；$dist(i,j)=1-\frac{\boldsymbol{q}_i\boldsymbol{q}_j}{\|\boldsymbol{q}_i\boldsymbol{q}_j\|}$ 为产品 i 和 j 之间的距离或者称为不相似的程度；$\boldsymbol{q}_i$ 和 $\boldsymbol{q}_j$ 为产品 i 和 j 在评分矩阵 $\boldsymbol{R}$ 中所属的列向量。

DCG 为考虑排序的精确度指标，定义如式(22)所示：

$$DCG=\frac{1}{|U|}\sum_{u\in U}\sum_{i\in S_u}\frac{2^{r_i}-1}{\log_2(i+1)} \tag{22}$$

其中，若 $i\in S_u\wedge i\in T_u$，则 $r_i=1$，否则 $r_i=0$。

[1] https://grouplens.org/datasets/movielens/

此外，为了更好地对比本文提出的 RGA 和 SGA 与基础的 EGA 这 3 个基于用户覆盖概念的模型对用户兴趣域覆盖的性能，本文还对各个模型的 *PID* 指标进行了对比。通过 *PID* 指标计算得到推荐列表中产品对用户兴趣域的覆盖比率。若 *PID* 值越大，则说明该模型的目标函数可以在更大范围内以更快的速度覆盖用户的兴趣域。其定义如式(23)所示：

$$\mathrm{PID}=\frac{nozeros(\boldsymbol{C}_S \wedge \boldsymbol{C}_P)}{nozeros(\boldsymbol{C}_P)} \tag{23}$$

其中，$\boldsymbol{C}_S$ 和 $\boldsymbol{C}_P$ 分别为产品集向量和用户向量；nozeros 返回向量非零元素的个数；符号“∧”为向量上的按位与运算，仅当 $\boldsymbol{C}_S[i]\neq 0$ 且 $\boldsymbol{C}_P[i]\neq 0$ 时，$\boldsymbol{C}_S[i]\wedge\boldsymbol{C}_P[i]=1$；否则 $\boldsymbol{C}_S[i]\wedge\boldsymbol{C}_P[i]=0$。

5.3　实验结果

本文选取目前较为新颖的 SUB 模型[12]、EGA 模型[13]，与本文提出的 RGA 模型、SGA 模型进行对比实验。首先，将各种算法在 Precision，GC，TGC 和 ILD 这 4 个指标上进行对比。总体上说，这 4 项指标的数值越大，模型性能就越优秀。

此外，为了使实验更具有普遍性和说服性，本文将根据不同的推荐列表长度进行多次实验。实验中，依次将推荐列表的长度设置为 5，10，15，20，其具体实验的结果如表 2 和表 3 所列。

表 2　MovieLens1M 上的实验结果

Table 2　Results on MovieLens1M

K	Metric	SUB	EGA	RGA	SGA
5	*Precision*	0.0409	0.0422	0.0435	**0.0534**
	GC	0.5381	0.5727	**0.5751**	0.5331
	TGC	0.5327	0.5758	**0.5780**	0.5303
	ILD	0.4108	0.4284	**0.4288**	0.4133
10	*Precision*	0.0749	0.0752	0.0795	**0.0898**
	GC	0.6803	0.7136	**0.7167**	0.6814
	TGC	0.6812	0.7173	**0.7200**	0.6754
	ILD	0.4097	0.4137	**0.4145**	0.4088
15	*Precision*	0.1034	0.1002	0.1021	**0.1167**
	GC	0.7681	**0.7968**	0.7959	0.7696
	TGC	0.7577	0.8046	**0.8055**	0.7649
	ILD	0.4001	0.4044	**0.4060**	0.3988
20	*Precision*	0.1169	0.1096	0.1112	**0.1418**
	GC	0.8231	0.8481	**0.8483**	0.8207
	TGC	0.8218	0.8500	**0.8562**	0.8136
	ILD	0.3994	0.4013	**0.4021**	0.3982

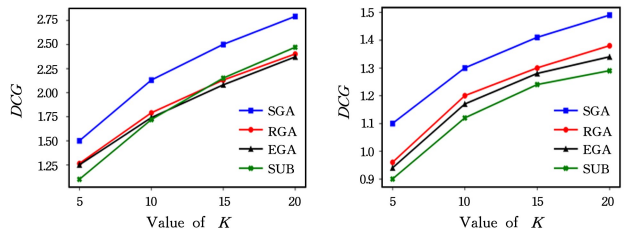
表 3　MovieLens-Latest-Small 上的实验结果

Table 3　Results on MovieLens-Latest-Small

K	Metric	SUB	EGA	RGA	SGA
5	*Precision*	0.0209	0.0224	0.0250	**0.0258**
	GC	0.6032	0.6352	**0.6359**	0.6068
	TGC	0.6075	**0.6395**	0.6383	0.5954
	ILD	0.4287	0.4652	**0.4684**	0.4202
10	*Precision*	0.0305	0.0315	0.0360	**0.0400**
	GC	0.7456	0.7763	**0.7774**	0.7399
	TGC	0.7430	0.7735	**0.7747**	0.7331
	ILD	0.4201	0.4538	**0.4570**	0.4156
15	*Precision*	0.0372	0.0398	0.0415	**0.0508**
	GC	0.8279	0.8411	**0.8425**	0.8145
	TGC	0.8302	0.8401	**0.8508**	0.8285
	ILD	0.4171	0.4476	**0.4502**	0.4109
20	*Precision*	0.0419	0.0448	0.0468	**0.0573**
	GC	0.8545	0.8786	**0.8822**	0.8532
	TGC	0.8519	0.8700	**0.8834**	0.8512
	ILD	0.4093	0.4382	**0.4441**	0.4021

由表 2 和表 3 可以看出：EGA 模型和 RGA 模型在多样性指标 *GC*，*TGC* 和 *ILD* 上表现优异，但在精确度指标 *Precision* 上落后于 SGA 模型。EGA 模型总体上优于 SUB 模型，而 RGA 模型全面优于 EGA。但当 $K\geqslant 15$ 时，在 MovieLens1M 数据集上，SUB 模型的 *Precision* 指标超过 EGA 和 RGA。而 SGA 模型在 *Precision* 指标上有着绝对的优势，大幅度领先 EGA 模型和 RGA 模型；但在 GC，TGC 等多样性指标上，落后于 EGA 模型和 RGA 模型，与 SUB 模型不相上下。总体上，RGA 全面优于 EGA，在多样性方面表现突出；SGA 具有一定的精确度优势。经过对 RGA 与 SGA 的进一步分析发现，上述情况与模型构造的目标差异紧密相关。RGA 模型的构造旨在根据评分差异实现用户覆盖的最大化，其计算公式与 *ILD* 评测指标直接相关；且由于用户覆盖趋于最大化，将使得 *GC* 和 *TGC* 评测效果极佳。SGA 模型通过寻找最相似产品向量来实现用户覆盖最大化，其潜在指标与推荐精确性密切相关，且并未直接参考产品之间的类别差异，因此在精确度指标上表现较好。总体而言，RGA 与 SGA 在推荐精确性以及多样性指标上，其数值差异仅在 0.01 左右，且各项指标相对优于 SUB 及 EGA，因此本文构建的 RGA 与 SGA 模型在推荐的精确性与多样性上都有一定程度的优势。

由于 SUB 和 SGA 在多样性指标上相差不大，为了更好地比较这两个具有一定精确度优势的模型，本文使用 DCG 指标对其精确度进行对比，其实验结果如图 2 所示。从图中可以看出，在 *DCG* 指标上，SGA 全面优于 RGA，EGA 和 SUB；RGA 优于 EGA。在 MovieLens 数据集上，当 $K<15$ 时，SUB 模型的 *DCG* 指标落后于 RGA；当 $K\geqslant 15$ 时，SUB 的 *DCG* 指标超过 RGA。而在 MovieLens-Latest-Small 数据集上，RGA 的 *DCG* 指标始终优于 SUB 模型。

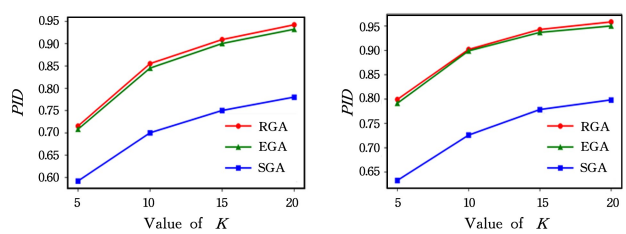


(a) MovieLens1M　(b) MovieLens-Latest-Small

图 2　累积折损信息增益关于 *K* 的变化趋势

Fig. 2　Variation of the discounted cumulative gain about *K*

此外，为了更好地检验目标函数最大化覆盖用户兴趣域的性能，本文还测量了 EGA，RGA 和 SGA 这 3 个模型在不同 *K* 值下的兴趣域覆盖比率(PID)，实验结果如图 3 所示。



(a) MovieLens1M　(b) MovieLens-Latest-Small

图 3　兴趣域覆盖比率关于 *K* 的变化趋势

Fig. 3　Variation of the proportion of interest domain about *K*

具体来说，这3个模型的兴趣域覆盖比率都随着 K 值的增大而增大。当 K 值较小时，EGA 和 RGA 的 *PID* 就可以达到较大的值。当 $K=5$ 时，在这两个数据集上，EGA 和 RGA 的 *PID* 值都可达 70%；当 $K=15$ 时，EGA 和 RGA 的 *PID* 值都已超过 90%。相比之下，SGA 在 *PID* 上表现一般，需要更大的 K 值才可以达到较高的 *PID* 值。

结束语 本文在基于用户覆盖的基础上，将推荐列表的生成过程转化为用户向量和产品集向量之间关联关系的计算过程，提高了计算的效率。并且，在计算用户兴趣点频率时，把用户在产品评分上的差异考虑在内，从而计算得到更加精准的用户向量。在不同数据集上的实验表明，本文提出的模型可以有效地提高推荐列表的精确性和多样性。

本文提出的两个模型都是通过贪心算法实现的。虽然算法的有效性有着严格的理论依据，但是使用贪心算法依然有着迭代次数多、计算量大等缺点。下一阶段将在用户覆盖的基础上着重解决效率方面的问题。

参 考 文 献

[1] NAHTA R, MEENA Y K, GOPALANI D, et al. Embedding metadata using deep collaborative filtering to address the cold start problem for the rating prediction task[J]. Multimedia Tools and Applications, 2021, 80(12): 18553-18581.

[2] XU J, YAO Y, TONG H, et al. HoORaYs: High-order optimization of rating distance for recommender systems[C]// Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '17). 2017: 525-534.

[3] HE Y, ZOU H, YU H, et al. Adaptive and efficient high-order rating distance optimization model with slack variable[J]. Knowledge-Based Systems, 2020, 205: 106228.

[4] ZHANG Q, LIU L, WEN J H. Recommendation Algorithm with Field Trust and Distrust Based on SVD[J]. Computer Science, 2019, 46(10): 27-31.

[5] MCNEE S M, RIEDL J, KONSTAN J A. Being accurate is not enough: How accuracy metrics have hurt recommender systems[C]// Extended Abstracts Proceedings of the 2006 Conference on Human Factors in Computing Systems. 2006: 1097-1101.

[6] CREMONESI P, F GARZOTTO, NEGRO S, et al. Looking for "Good" Recommendations: A Comparative Evaluation of Recommender Systems[M]// Human-computer Interaction-INTERACT, 2011. Berlin: Springer, 2011: 152-168.

[7] CHENG P, WANG S, MA J, et al. Learning to Recommend Accurate and Diverse Items[C]// Proceedings of the 26th International Conference on World Wide Web. 2017: 183-192.

[8] SHA C, WU X, NIU J. A Framework for Recommending Relevant and Diverse Items[C]// Proceedings of the 25th International Joint Conference on Artificial Intelligence. 2016: 3868-3875.

[9] PANTELI A, BOUTSINAS B. Improvement of similarity-diversity trade-off in recommender systems based on a facility location model[J]. Neural Computing and Applications, 2021(5): 1-13.

[10] LE W, LIU Q, CHEN E H, et al. Relevance Meets Coverage: A Unified Framework to Generate Diversified Recommendations[J]. ACM Transactions on Intelligent Systems and Technology, 2016, 7(3): 1-30.

[11] SALAKHUTDINOV R. Probabilistic Matrix Factorization[C]// Proceedings of the 20th International Conference on Neural Information Processing Systems. 2007: 1257-1264.

[12] SHAMEEM A P P, NICOLAS U, YVES G. A Coverage-Based Approach to Recommendation Diversity On Similarity Graph[C]// Proceedings of the 10th ACM Conference on Recommender Systems. 2016: 15-22.

[13] HE Y, ZOU H, YU H, et al. Diversity-Aware Recommendation by User Interest Domain Coverage Maximization[C]// 2019 IEEE International Conference on Data Mining (ICDM). 2019: 1084-1089.

[14] AZIN A, BRANISLAV K, SHLOMOB, et al. Optimal Greedy Diversity for Recommendation[C]// IJCAI. 2015: 1742-1748.

[15] HOCHBA, DORIT S. Approximation Algorithms for NP-Hard Problems[J]. ACM SIGACT News, 1997, 28(2): 40-52.

[16] FISHER M L, NEMHAUSER G L, WOLSEY L A. An Analysis of Approximations for Maximizing Submodular Set Functions I[J]. Mathematical Programming, 1978, 14(1): 265-294.

CHEN Zhuang, born in 1995, postgraduate. His main research interests include recommender system and so on.

ZOU Hai-tao, born in 1984, Ph.D, lecturer. His main research interests include data mining and information retrieval.

（责任编辑：柯颖）