基于双向主题模型的协同过滤算法*

李 改[1,2,3]，李 磊[2,3]
（1. 顺德职业技术学院电子与信息工程系，广东 顺德 528333；
2. 中山大学信息科学与技术学院，广东 广州 510006；
3. 中山大学软件研究所，广东 广州 510275）

摘 要：主题模型可以学习用户和推荐项目的潜在主题分布。提出了一种基于双向主题模型的协同过滤算法，分别学习用户和推荐项目的潜在主题分布用于推荐服务。在真实的数据集上实验验证，该算法的性能均优于几个经典的协同过滤算法。

关键词：推荐系统；协同过滤；主题模型；潜在狄利克雷分布
中图分类号：TP301 **文献标志码**：A **文章编号**：0529－6579（2013）05－0068－05

Dual Collaborative Topic Regression for Recommendation Systems

LI Gai[1,2,3], *LI Lei*[2,3]
（1. School of Electronics and information engineering, Shunde Polytechnic, Shunde 528333, China;
2. School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, China;
3. Software Institute, Sun Yat-sen University, Guangzhou 510275, China）

Abstract: Topic model can be used to learn the latent topic distribution. A new collaborative filtering algorithm based on dual collaborative topic regression to learn the user's latent topic distribution and the item's latent topic distribution for recommendation is proposed. On a large real-world dataset, the experiment results illustrate that the approach achieves a better performance than the state-of-the art collaborative filtering methods.

Key words: recommended systems; collaborative filtering; topic model; latent Dirichlet allocation

推荐系统通过收集和分析用户的各种信息来学习用户的兴趣和行为模式，根据分析得到的用户的兴趣和行为模式，来为用户推荐他所需要的服务。这些系统的例子包括：CiteULike 论文推荐系统（www. citeulike. org）为用户推荐各种其可能感兴趣的论文；Netflix 电影出租系统（www. netflix. com）为用户推荐各种其可能喜欢的电影。Google、Baidu、Yahoo 等为用户提供个性化的新闻推荐和搜索服务。推荐系统中运用最广泛的是基于协同过滤的推荐算法[1－4]。

协同过滤的算法核心是分析用户兴趣，在用户群中找到与指定用户的相似（兴趣）用户，综合这些相似用户对某一信息的评价，形成系统对该指定用户对此信息的喜好程度预测。近年来协同过滤的算法在国内外得到了广泛研究。传统的协同过滤算法面临数据稀疏性问题和评分数据的不平衡问题。许多模型提出通过引入额外的信息来解决这些问题以增强推荐效果：如推荐项目的内容信息[4]，用户的社交网络信息[5－6]，用户本身的属性信息[7]等。协同主题模型（CTR）就是这类模型中最新的一种模型[4－5]。CTR 模型通过在传统的基于矩阵分解的协同过滤算法中引入潜在狄利克雷分布（LDA）来学习文本形式的推荐项目的潜在主题分布[8]，从而增强推荐效果。文本形式的推荐项目

* **收稿日期**：2012－12－24
基金项目：国家自然科学基金资助项目（61003140，61033010）；中山大学高性能与网格计算平台资助项目
作者简介：李改（1981 年生），男；**研究方向**：机器学习与数据挖掘；E-mail：ligai999@126. com

在现实生活中广泛存在，如科学文献，网页，微博等。如何有效的推荐这类文本形式的对象给所需要的用户是当前协同过滤领域的一个重要研究课题。

本文的主要贡献是：

①在 CTR 模型的基础上，提出了一种新的基于双向主题模型的协同过滤算法（DCTR）；

②提出了两种学习用户的主题分布向量的方法，并实验验证了两种方法的优劣。

③实验验证了 DCTR 算法的有效性。

本文具体内容安排如下：第 1 节介绍基本定义、LDA 模型和 CTR 模型简介；第 2 节详细介绍本文所提出的基于双向主题模型的协同过滤算法。第 3 节针对所提出的算法进行实验验证，并对实验结果进行分析；最后第 4 节是本文的总结。

1　基本定义、LDA 模型和 CTR 模型简介

1.1　基本定义

在本文中矩阵用斜体大写字母表示（如：R），标量用小写字母表示（如：i，j）。给定一个矩阵 R，R_{ij} 表示它的一个元素，$R_{i\cdot}$ 表示矩阵 R 的第 i 行，$R_{\cdot j}$ 表示矩阵 R 的第 j 列，R^{T} 表示矩阵 R 的转置。R^{-1} 表示矩阵 R 的逆。在本文中给定的矩阵 R 表示具有 I 个用户、J 个对象的评分矩阵，矩阵 U、V 分别表示用户和推荐对象的特征矩阵。

1.2　LDA 模型简介

潜在狄利克雷分布（LDA）是一种最简单的主题模型，图 1 是 LDA 模型的概率图。

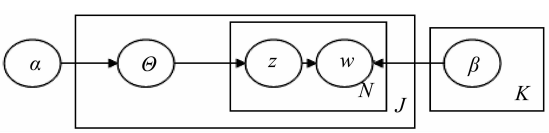

图 1　LDA 模型

Fig. 1　LDA model

假定有 K 个主题，即 $\beta = \beta_{1:K}$，是一个向量，其中的每个元素值表示一个词表的分布。这里的参数 α 是一个超参数，用于控制主题分布 θ。LDA 模型的运行过程如以下算法 1 所示。

算法 1　LDA 模型的运行过程

输入：超参数 α，向量 β。

输出：每个文本形式的推荐项目的主题分布 θ_j。对于每个文本形式的推荐项目 j：

① 得到主题分布 θ_j，θ_j 满足参数为 α 的狄利克雷分布，即 $\theta_j \sim \mathrm{Dirichlet}(\alpha)$。

② 对推荐项目中的每个单词 w_{jn}

（i）得到该单词的主题 z_{jn}，z_{jn} 服从参数为 θ_j 的多项式分布，即 $z_{jn} \sim \mathrm{Mult}(\theta_j)$。

（ii）得到单词 w_{jn}，w_{jn} 服从参数为 $\beta_{z_{jn}}$ 的多项式分布，即 $w_{jn} \sim \mathrm{Mult}(\beta_{z_{jn}})$。

对于文本形式的推荐项目，我们可以运用 LDA 模型学习该推荐项目的主题分布向量 θ_j。从而可以对推荐项目 j 的特征向量实施约束，使其满足均值为 θ_j 的正态分布，即 $V_j \sim N(\theta_j, \lambda_v^{-1} I_K)$，其中 I_K 表示秩为 K 的单位矩阵。在协同过滤算法中引入 LDA 模型模型可以有效提高推荐性能。LDA 模型的具体详细算法描述可见参考文献［8］。

1.3　CTR 模型简介

CTR 模型是一种基于主题模型的协同过滤算法，图 2 是 CTR 模型的概率图。

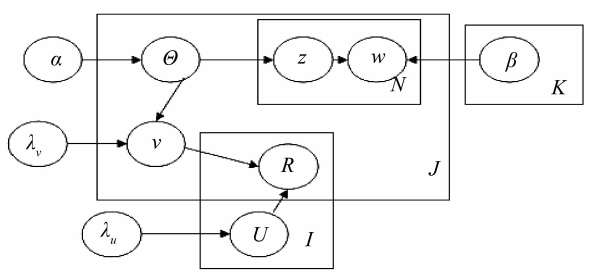

图 2　CTR 模型

Fig. 2　CTR model

CTR 模型综合了传统的协同过滤算法和概率主题模型的优点。其中用户的特征向量符合均值为 0 的正态分布，用于表示用户的兴趣；推荐项目符合均值为 θ 的正态分布，其潜在方差为 ε。CTR 模型的运行过程如以下算法 2 所示。

算法 2　CTR 模型的运行过程

输入：用户的正则化系数 λ_u，推荐项目的正则化系数 λ_v。

输出：矩阵 R 的逼近矩阵 X。

① 对于每个用户 i，我们从分布 $N(0, \lambda_u^{-1} I_K)$ 中抽取用户的潜在特征向量 U_i，即 $U_i \sim N(0, \lambda_u^{-1} I_K)$；

② 对于每个文本形式的推荐项目 j；

（i）运用算法 1 所描述的 LDA 模型得到其主题分布 θ_j。

（ii）得到推荐项目的潜在方差 ε_j，ε_j 满足分布 $N(0, \lambda_v^{-1} I_K)$。

（iii）得到推荐项目的特征向量 $V_j = \theta_j + \varepsilon_j$。

即 $V_j \sim N(\theta_j, \lambda_v^{-1} I_K)$。

③对于每个评分点 (i,j)，得到相应的预测评分

$$X_{ij} \sim N(U_i^{\mathrm{T}} V_j, C_{ij}^{-1})$$

在这里参数 C_{ij} 是对于评分点的值 X_{ij} 的信任参数，参数 C_{ij} 的值越大，表示评分值 X_{ij} 越可信；当参数 C_{ij} 的值为 0 时，可解释为用户 i 对推荐项目 j 不感兴趣或没有留意到项目 j。这就是有名的单类协同过滤问题。我们与文献［4－5，9－11］中的处理方式一样，来为参数 C_{ij} 赋予一定的权值

$$C_{ij} = \begin{cases} a, & \text{if} \quad X_{ij} = 1, \\ b, & \text{if} \quad X_{ij} = 0, \end{cases} \tag{1}$$

这里的 a，b 是控制参数，满足 $1 \geqslant a > b > 0$。

在 CTR 模型中，我们只是考虑对推荐项目 j 的特征向量实施约束，使其满足均值为推荐项目 j 的主题分布向量的正态分布，即 $V_j \sim N(\theta_j, \lambda_v^{-1} I_K)$；其实我们也可以对用户的特征向量实施约束，使其也符合以某种主题分布向量为均值的正态分布。基于这个思想，我们提出了一种新的基于双向主题模型的协同过滤算法 DCTR。

2 基于双向主题模型的协同过滤算法（DCTR）介绍

DCTR 模型是一种基于双向主题模型的协同过滤算法，图 3 是 DCTR 模型的概率图。

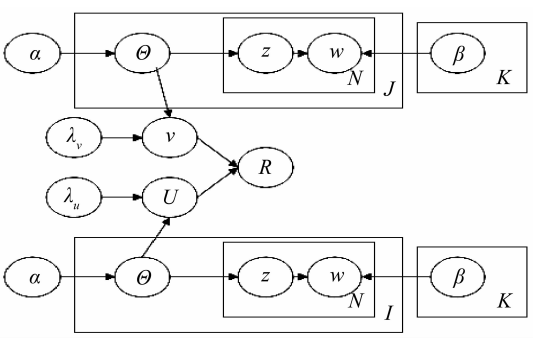

图 3 DCTR 模型

Fig. 3 DCTR model

DCTR 从用户和推荐项目这两个方面，分别对用户的特征向量和推荐项目的特征向量进行约束，使他们都符合以某种主题分布向量为均值的正态分布，即 $U_i \sim N(\theta_i, \lambda_u^{-1} I_K)$，$V_j \sim N(\theta_j, \lambda_v^{-1} I_K)$。其中 θ_i 为用户 U_i 的主题分布向量，θ_j 为推荐项目 V_j 的主题分布向量。DCTR 模型的运行过程如以下算法 3 所示。

算法 3 DCTR 模型的运行过程

输入：用户的正则化系数 λ_u，推荐项目的正则化系数 λ_v。

输出：矩阵 R 的逼近矩阵 X。

①对于每个用户 i；

（i）得到其主题分布 θ_i。θ_i 的值有两种获得方法：

a）取用户 i 所评过的项目的主题分布向量的均值作为用户 i 的主题分布向量。

b）把用户 i 所评过的项目的描述文本的集合作为用户 i 的描述文本内容，重新运用算法 1 所描述的 LDA 模型来学习用户 i 的主题分布向量。

（ii）得到用户的潜在方差 ε_i，ε_i 满足分布 $N(0, \lambda_u^{-1} I_K)$。

（iii）得到用户的特征向量 $U_i = \theta_i + \varepsilon_i$。即：$U_i \sim N(\theta_i, \lambda_u^{-1} I_K)$。

② 对于每个文本形式的推荐项目 j；

（i）运用算法 1 所描述的 LDA 模型得到其主题分布 θ_j。

（ii）得到推荐项目的潜在方差 ε_j，ε_j 满足分布 $N(0, \lambda_v^{-1} I_K)$。

（iii）得到推荐项目的特征向量 $V_j = \theta_j + \varepsilon_j$。即：$V_j \sim N(\theta_j, \lambda_v^{-1} I_K)$。

③ 对于每个评分点 (i,j)，得到相应的预测评分：

$$X_{ij} \sim N(U_i^{\mathrm{T}} V_j, C_{ij}^{-1})$$

为了学习模型的参数，我们在这里提出了一种与文献［4－5］中类似的 EM 算法。模型的参数我们可以通过最大化公式（2）得到

$$L(U,V) = -\sum_{ij} \frac{C_{ij}}{2} (R_{ij} - U_{i\cdot} V_{j\cdot}^{\mathrm{T}})^2 - \frac{\lambda_u}{2} (U_{i\cdot} - \theta_i)^{\mathrm{T}} (U_{i\cdot} - \theta_i) - \frac{\lambda_v}{2} (V_{j\cdot} - \theta_j)^{\mathrm{T}} (V_{j\cdot} - \theta_j) + \sum_j \sum_n \log\left(\sum_k \theta_{jk} \beta_{k,w_{jn}} \right) \tag{2}$$

在这里我们忽略了一个常数项，并且设置狄利克雷分布的参数 $\alpha = 1$。我们可以通过梯度下降法求解公式（2）。固定 V，对 U_i 求导 $\frac{\partial L(U,V)}{\partial U_i} = 0$，我们得到下面求解 $U_{i\cdot}$ 的公式。

$$U_{i\cdot} = (VC_i V^{\mathrm{T}} + \lambda_u I_K)^{-1} (VC_i R_{i\cdot} + \lambda_u \theta_i), \quad i \in [1 \sim I] \tag{3}$$

同理，得到求解 $V_{j\cdot}$ 的公式。

$$V_{j\cdot} = (UC_j U^{\mathrm{T}} + \lambda_v I_K)^{-1} (UC_j R_{j\cdot} + \lambda_v \theta_j), \quad j \in [1 \sim J] \tag{4}$$

在这里 C_i 表示一个以矩阵 C 的第 i 行为对角元素，其它元素值为 0 的对角矩阵。C_j 的定义与 C_i 的定义一致。从公式（3）、（4）不难看出用户 i 的主题分布向量 θ_i 影响用户 i 的特征向量 U_i，推荐项目 j 的主题分布向量 θ_j 影响推荐项目的特征向量 V_j。

反复迭代运用公式（3）、（4）更新 U、V，直到本算法计算出的 recall@ M 值收敛或迭代次数足够多而结束迭代。$X = UV^{\mathrm{T}}$，矩阵 X 即为矩阵 R 的逼近矩阵。

3　实验结果及分析

本节首先介绍本文实验所采用的数据集及评价标准。接着给出了本文所提出的基于双向主题模型的协同过滤算法的参数对实验结果的影响，并把所提算法的试验结果与其他几个经典算法的实验结果进行比较。

3.1　实验数据集

在本实验中，我们使用了与文献［4］相同的 CiteULike 数据集。

CiteULike 数据集是一个有关科学研究者参考科学文献的数据集。在这个数据集中每个用户维护有一个他敢兴趣的文献列表。这个数据集包括了 5 551个用户对 16 980 篇科学文献的 204 986 个引用记录。这是一个 0/1 数据集。矩阵的稀疏度是 99.8%。平均来说，每个用户引用了 37 篇文献，引用的范围最少 10 篇，最多 403 篇。93% 的用户引用的文献数少于 100 篇。对于每篇文献我们把它的题目和摘要合起来作为它的描述性文本，我们移除其中的标点符号，通过 TF-IDF 方法选择其中的 8 000 个单词构成词库。这些文献的发表时间是从 2004 年到 2010 年。平均来说，每篇文献出现在 12 个用户的引用列表中，最少出现在一个用户的引用列表，最多出现在 321 个用户的引用列表。97% 的文献出现在用户列表的次数少于 40 次。

我们采用 5 折交叉确认的方式来进行试验。对于出现在用户的列表中超过 5 次的文献，我们把它的评分点（0 和 1）平均的分为 5 份。我们迭代的选择其中的 4 份为训练集，剩下的一份为测试集。对于那些出现在用户的文献列表中少于 5 次的文献都放入训练集。这就保住了测试集中的每个文献都出现在训练集中。对所有的用户求 5 次 5 折交叉确认的试验结果，取平均值作为该用户的最终试验结果，所有用户的实验结果求平均作为整个系统的最终试验结果。

3.2　实验的评价标准

本文实验采用 recall@ M 作为评价标准[4-5]，recall@ M 通过对模型的预测值进行排序，计算排序后的前 M 个项目中占所有该用户的测试项的比例来作为试验结果。当 M 值取某个较小的固定值的情况下，recall@ M 越大系统性能越好，这个系统的 recall@ M 值为每个用户的 recall@ M 值的平均值。recall@ M 的定义如下：

$$\text{recall@ M} = \frac{\text{numberofarticlestheuserlikesintopM}}{\text{totalnumberofarticletheuserlikes}} \tag{5}$$

3.3　实验结果

本实验中的所有实验结果 recall@ M 的 M 值均取值为 200。参数 C_{ij} 的取值为 $a = 1$，$b = 0.01$。

3.3.1　DCTR 模型的参数对模型性能的影响分析　从图 4 和图 5 可以看出随着 λ_u 和 λ_v 的增大，DCTR 模型的性能均先升高，后下降。说明用户和推荐项目的特征向量偏离他们的主题分布向量不能太远，也不能太近，有一个临界值。从本模型的实验可以看出，λ_u 和 λ_v 的最优值均是 100。

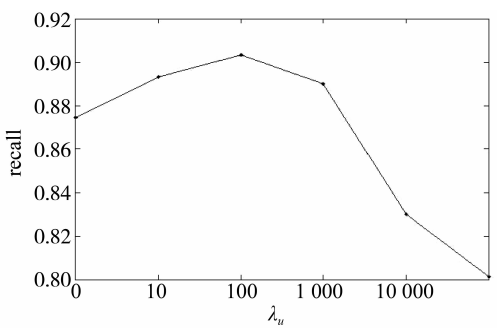

图 4　参数 λ_u 对 DCTR 模型的性能影响

Fig. 4　The influence of λ_u to DCTR model's performance

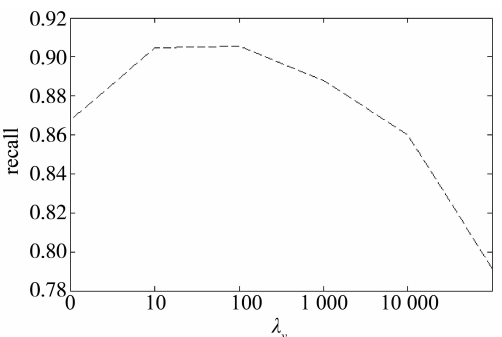

图 5　参数 λ_v 对 DCTR 模型的性能影响

Fig. 5　The influence of λ_v to DCTR model's performance

3.3.2　基于 DCTR 模型的算法和几个经典的 CF 算法的性能比较　在这里我们将把 DCTR 模型和几个

经典的 CF 算法的性能比较。本实验中要比较的几个 CF 算法分别是：

CTR 算法，是文献［4］中所提出的一种基于主题模型的协同过滤算法，它只是对推荐项目的特征向量实施约束。通过实验交叉验证，在该算法中 λ_u 取值为 0.01，λ_v 取值为 100 时，算法性能最好。

CTRUI 算法，是基于本文所提出的 DCTR 模型的协同过滤算法，在该算法中，用户的主题分布向量取值为他所评过的所有项目的主题分布向量的均值。在该算法中 λ_u 和 λ_v 均取最优值 100。

WPMF 算法，是加权的基于概率矩阵分解的协同过滤算法[9-11]。通过实验交叉验证，在该算法中 λ_u 和 λ_v 取值为 0.01 时，算法性能最好。

CTRUIReal 算法，是基于本文所提出的 DCTR 模型的协同过滤算法，在该算法中，我们把某个用户评过的所有推荐项目的文本描述的集合作为该用户的文本描述。通过对每个用户运行 LDA 算法，来得到该用户的主题分布向量。在该算法中 λ_u 和 λ_v 均取最优值 100。

图 6 中横轴表示各个算法的迭代次数，纵轴表示各个算法的 recall 值。

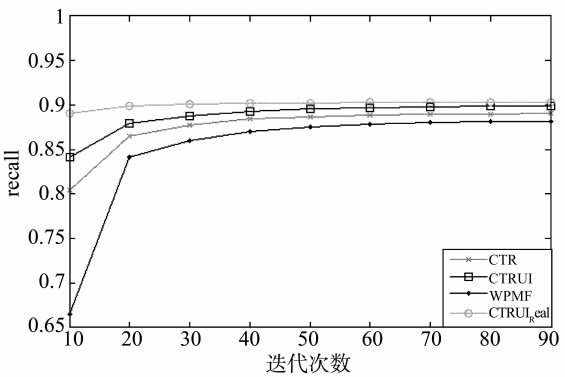

图 6 基于 DCTR 模型的算法和几个经典的 CF 算法的性能比较

Fig. 6 The performance comparation of DCTR model and several classical CF methods

从图 6 可以看出基于 DCTR 模型的两个协同过滤算法 CTRUIReal 算法和 CTRUI 算法在 recall 性能上均优于 CTR 算法和 WPMF 算法，随着迭代次数的增加性能的差异越来越明显，这说明对用户和推荐项目的特征向量分别引入主题模型进行约束能够有效提高算法性能。并且 CTRUIReal 算法的性能优于 CTRUI 算法的性能，这说明相比于取评过的推荐项目的主题分布向量的均值作为该用户的主题分布向量，通过主题模型直接学习用户的主题分布向量更为可靠。还可看出基于 DCTR 模型的两个协同过滤算法 CTRUIReal 算法和 CTRUI 算法的收敛速度也明显快过 CTR 算法和 WPMF 算法，这也进一步说明了本文所提算法的有效性。

4 总 结

本文在传统的矩阵分解模型和主题模型的基础上提出了一种新的基于双向主题模型的协同过滤算法，它运用 LDA 算法从用户和推荐项目两个方向对用户和推荐项目的特征向量进行约束，以便更有效的推荐文本形式的对象给所需要的用户。在真实的数据集上的实验结果表明：在 recall@ M 性能指标下，基于本文所提出的 DCTR 模型的协同过滤算法的性能明显优于几个传统的协同过滤算法。在以后的工作中我们还将研究本文所提算法的并行化问题。

参考文献：

[1] WU J L. Collaborative filtering on the netflix prize dataset [D]. Peking University, 2010.

[2] RICCI F, ROKACH L, SHAPIRA B, et al. Recommender system handbook [J]. Springer, 2011, 12 – 120.

[3] 罗辛,欧阳元新,熊璋,等. 通过相似度支持度优化基于 K 近邻的协同过滤算法[J]. 计算机学报,2010, 33(8): 99 – 105.

[4] WANG C, BLEI D. Collaborative topic modeling for recommending scientific articles[C]//In ACM KDD, 2011, 448 – 456.

[5] PURUSHOTHAM S, LIU Y, KUO C J. Collaborative topic regression with social matrix factorization for recommendation systems [C]//In ACM ICML, 2012, 1255 – 1265.

[6] MA H, ZHOU D, LIU C, et al. Recommender system with social regularization [C]//In ACM WSDM, 2011, 287 – 296.

[7] LI Y, HU J, ZHAI C, et al. Improving one-class collaborative filtering by incorporating rich user information [C]//In ACM CIKM, 2010, 959 – 968.

[8] BLEI D, NG A, JORDAN M. Latent Dirichlet allocation [J]. Journal of Machine Learning Research, 2002, 993 – 1022.

[9] PAN R, ZHOU Y, CAO B, et al. On e-class collaborative filtering [C]//In IEEE ICDM, 2008, 502 – 511.

[10] PAN R, MARTIN S. Mind the gaps: weighting the unknown in large-scale one-class collaborative filtering [C]//In ACM KDD, 2009, 667 – 675.

[11] YANG X, STECK H, GUO Y, et al. On top-k recommendation using social networks [C]//In ACM RecSys, 2012, 67 – 74.