

DOI:10.13232/j.cnki.jnju.2021.05.012

基于特定用户约束的概率矩阵分解算法

郝昱猛, 马文明*, 王 冰

(烟台大学计算机与控制工程学院, 烟台, 264005)

摘 要:近年来,推荐系统的实用价值越来越高,良好的推荐算法可以给用户提供好的用户体验效果,然而随着信息化的不断增长,信息过载问题变得越来越突出,用户懒于对物品评分已经成为习惯.怎样向这些特定用户群体提供好的推荐算法、提高推荐质量已经成为现在的热门问题.为了更好地推动推荐系统的发展,解决这些特定用户群体的评分稀疏问题,提出一种受约束的贝叶斯概率矩阵分解算法.该算法针对特定的评分稀疏用户引入一种潜在的相似度约束矩阵来影响用户的特征向量,并结合最大后验概率(Maximum A Posteriori, MAP)估计和蒙特卡罗采样(Markov Chain Monte Carlo, MCMC)推断进行概率矩阵分解(Probabilistic Matrix Factorization, PMF),自动调整模型正则化参数,最后在数据集上进行测试评估和对比实验.实验结果表明,该算法在预测性能上得到很大提升,并且在解决特定用户的评分稀疏问题上效果更佳.

关键词:推荐系统,评分稀疏,约束矩阵,概率矩阵分解,协同过滤

中图分类号:TP301

文献标志码:A

Probabilistic matrix factorization algorithm based on specific user constraints

Hao Yumeng, Ma Wenming*, Wang Bing

(School of Computer and Control Engineering, Yantai University, Yantai, 264005, China)

Abstract: In recent years, the practical value of recommender system is getting higher and higher. A good recommendation algorithm can provide a good experience effect for users. However, with the continuous growth of information technology, the problem of information overload has become more and more prominent. The formation of living habits for users is to be lazy about rating items. How to provide good recommendation algorithm to these specific user groups and to improve the recommendation quality has become a hot issue now. In order to improve recommender system and solve rating sparse problem of specific user groups, a constrained Bayesian probability matrix factorization algorithm is proposed. This algorithm introduces a potential similarity constraint matrix to affect user eigenvectors for specific sparse rating users. We combine Maximum A Posteriori (MAP) and Markov Chain Monte Carlo (MCMC) for Probabilistic Matrix Factorization (PMF), automatically adjusting the regularization parameters. Finally, test evaluation and comparison experiments are conducted on the MovieLens dataset. The experimental results show that the proposed algorithm can improve prediction performance and perform well for specific users with sparse ratings.

Key words: recommendation system, data sparseness, constraint matrix, probabilistic matrix factorization, collaborative filtering

基金项目:国家自然科学基金(61602399),烟台大学研究生科技创新基金(YDZD2119)

收稿日期:2021-06-02

* 通讯联系人, E-mail: mwmtyu@126.com

推荐系统(Recommendation System, RS)^[1-3]提前给用户列出他们感兴趣的物品或项目,所以能为用户节省很多挑选商品的时间。RS最早的应用领域局限于电影、音乐和电视节目的推荐,随着使用范围的扩大,RS逐渐应用于购物推荐^[4]、书籍推荐^[5]和事件推荐^[6-7]等。现在国内外应用RS最广泛的平台有天猫、京东、亚马逊、豆瓣、MEETUP等,RS为他们的用户提供高效快捷的推荐列表,能提高用户的满意度,而平台也获得了更好的经济效益。大多数RS中,最经常使用的是协同过滤算法(Collaborative Filtering, CF)^[8-9],它通过分析用户的偏好信息,预测其可能喜爱的物品。

如今用户评分矩阵十分稀疏,所以传统推荐算法很难为用户配置准确的推荐列表。在这种情况下,CF利用用户与项目之间的交互并根据用户与数据集中其他可用用户的相似信息来生成项目预测,产生推荐列表。矩阵分解(Matrix Factorization, MF)^[10-11]技术在CF中被成功地使用,它通过考虑用户与项目之间的互动因素(潜在特征),有效地解决了评分矩阵数据量大、数据稀疏等问题。

随着用户对推荐准确性的要求越来越高,有很多方法通过改进MF技术来保障预测精度。Ortega et al^[12]提出一种基于伯努利分布的矩阵分解算法,利用模型分布的二进制性质提高推荐的准确性和可靠性,但该算法在稀疏矩阵问题上仍需改进。陈珏伊等^[13]提出一种基于迁移学习的联合矩阵分解算法,通过捕捉用户的潜在特征提高相似性度量效果,该算法在解决书籍稀疏问题上有很好的效果,但算法复杂度较高,计算时间较长。Salakhutdinov and Mnih^[14]通过对概率矩阵分解(Probabilistic Matrix Factorization, PMF)引入约束矩阵,可以有效解决数据稀疏问题,提高预测准确性,但其模型参数需要手动调节,很容易产生过拟合问题。Salakhutdinov and Mnih^[15]还通过引入贝叶斯模型并使用蒙特卡罗采样(Markov Chain Monte Carlo, MCMC)对模型参数进行自动控制,能有效地解决模型的过拟合和优化问题,但针对特有的评分非常稀疏的用户,模型没有给出很好的预测。

本研究针对推荐系统中存在的数据稀疏、预测准确性等问题,提出一种带有用户约束的基于贝叶斯的概率矩阵分解优化算法,通过对用户引入特征向量以外的约束向量来提高用户之间的相似性,并使用最大后验概率估计^[16]自动确定MCMC采样器的起点来逼近模型参数的后验分布,有效地提高了推荐算法的可解释性。

本文的主要贡献:

(1)介绍有约束的概率矩阵分解模型和基于贝叶斯的概率矩阵模型,并总结了两个模型的特点。

(2)提出新的有约束矩阵的贝叶斯概率矩阵分解模型,通过增加约束矩阵对模型进行优化,采用MCMC采样方法对超参数进行采样。

(3)实验结果证明,本文提出的优化算法与其他算法比较,模型的预测效果更好。

1 相关理论

本节主要介绍有约束的概率矩阵分解(Constrained Probabilistic Matrix Factorization, CPMF)模型和基于贝叶斯的概率矩阵分解(Bayesian Probabilistic Matrix Factorization, BPMF)模型的基本方法,介绍了两种模型对传统的矩阵分解模型进行改进的相关研究。

1.1 CPMF 模型 2007年Salakhutdinov and Mnih^[14]提出PMF模型。假设评分矩阵为 $R = [R_{ij}]_{N \times M}$,其中 $i \in N$ (N 表示有 N 个用户), $j \in M$ (M 表示有 M 个电影), R_{ij} 表示用户 i 对电影 j 的评分矩阵。用户和电影的特征向量分别为 $U \in R^{N \times D}$, $V \in R^{M \times D}$, D 为 D 维的潜在特征向量。CPMF模型是在PMF模型的基础上引入约束矩阵 $W \in R^{M \times D}$ 来约束特定于用户的特征向量,这对于评分不频繁的用户有很强的影响。CPMF模型的概率图模型如图1所示,模型定义的新的用户特征向量如式(1)所示:

$$U_i = Y_i + \frac{\sum_{k=1}^M I_{ik} W_k}{\sum_{k=1}^M I_{ik}} \quad (1)$$

I_{ik} 为观察到的指示矩阵 I_{ij} ,如果用户 i 对电影 j 进行了评分,那么 $I_{ij} = 1$,否则 $I_{ij} = 0$ 。这里可以看出 W 矩阵的第 i 列表示的是捕捉用户对特定电

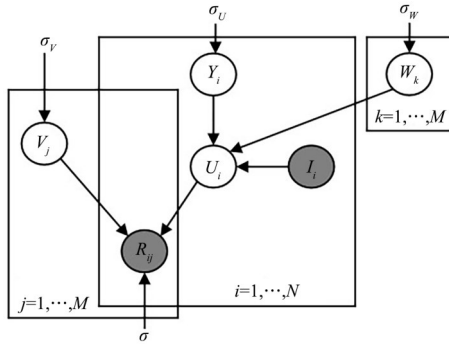


图 1 CPMF 模型

Fig. 1 CPMF model

影的评分对用户特征向量的先验平均值的影响, 所以看过共同电影的用户对其特征向量有相似的先验分布. 这里的 Y_i 是为了获取用户 i 的特征向量 U_i 而添加到先验分布平均值的偏移值. 式 (2) 为重新定义的模型条件分布函数:

$$P(R|U, V, W, \sigma^2) = \prod_{i=1}^N \prod_{j=1}^M \left[\mathcal{N}(R_{ij} | g(U_i^T V_j), \sigma^2) \right]^{l_{ij}} \quad (2)$$

使用 Logistic 函数^[17] $g(x) = \frac{1}{1 + e^{-x}}$ 来传递用户 U 与电影 V 之间的点积, 其中用户特征向量 U 、电影特征向量 V 和约束特征向量 W 的元素都是服从均值为 0, 方差分别为 σ_U^2 , σ_V^2 和 σ_W^2 的高斯先验分布, 它们的定义函数分别为:

$$P(U|\sigma_U^2) = \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \quad (3)$$

$$P(V|\sigma_V^2) = \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \quad (4)$$

$$P(W|\sigma_W^2) = \prod_{k=1}^M \mathcal{N}(W_k | 0, \sigma_W^2 I) \quad (5)$$

为计算方便, 对 U , V 和 W 的后验分布取自然对数, 得到的对数后验函数的最大值等价于最小化含有二次正则项的平方误差和的目标函数, 这里使用梯度下降法进行目标函数的最小化. 对比传统的 PMF 模型可用发现, 引入约束矩阵和 Logistic 函数对评分稀疏的特定用户有很好的推荐效果, 但没有很好地解决过拟合问题.

1.2 基于贝叶斯的概率矩阵分解(BPMF)模型

Salakhutdinov and Mnih^[15] 又提出将矩阵分解模型应用于贝叶斯框架中, 生成具有多元高斯先验分布的评分概率模型, 即 BPMF 模型, 如图 2 所示. 模型中用户 U 和电影 V 的概率分布单独存在:

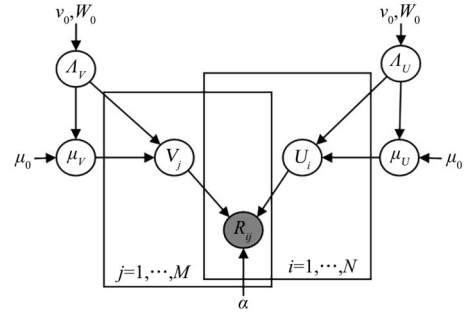


图 2 基于贝叶斯的 PMF 模型

Fig. 2 Bayesian PMF model

$$P(U|\mu_U, \Lambda_U) = \prod_{i=1}^N \mathcal{N}(U_i | \mu_U, \Lambda_U^{-1}) \quad (6)$$

$$P(V|\mu_V, \Lambda_V) = \prod_{j=1}^M \mathcal{N}(V_j | \mu_V, \Lambda_V^{-1}) \quad (7)$$

用户对电影的评分目标函数可以表示为:

$$P(R|U, V, \alpha) = \prod_{i=1}^N \prod_{j=1}^M \left[\mathcal{N}(R_{ij} | U_i^T V_j, \alpha^{-1}) \right]^{l_{ij}} \quad (8)$$

$\mathcal{N}(R_{ij} | \mu, \alpha^{-1})$ 表示期望值为 μ 、方差为 α^{-1} 的高斯分布. 为了提高模型的预测性, 将先验分布设置为高斯-威沙特分布, 其中 $\Theta_U = \{\mu_U, \Lambda_U\}$, $\Theta_V = \{\mu_V, \Lambda_V\}$. $\{\mu_U, \mu_V\}$ 通常设置为 0, 这里为了寻找更合适的参数, 将 $\{\Lambda_U, \Lambda_V\}$ 封装在模型内部, 可以减少优化参数过程.

$$P(\Theta_U | \Theta_0) = P(\mu_U | \Lambda_U) P(\Lambda_U) = \mathcal{N}(\mu_U | \mu_0, (\beta_0 \Lambda_U)^{-1}) \mathcal{W}(\Lambda_U | W_0, v_0) \quad (9)$$

$$P(\Theta_V | \Theta_0) = P(\mu_V | \Lambda_V) P(\Lambda_V) = \mathcal{N}(\mu_V | \mu_0, (\beta_0 \Lambda_V)^{-1}) \mathcal{W}(\Lambda_V | W_0, v_0) \quad (10)$$

$\mathcal{W}$ 是自由度为 v_0 、协方差矩阵为 W_0 的威沙特分布, 同时设置 $\Theta_0 = \{\mu_0, v_0, W_0\}$, $v_0 = D$, $\mu_0 = 0$.

BPMF 模型使用 MCMC 方法中的吉布斯采样器, 从先验分布和超先验分布中采样来进行近似推理. 实验证明, 该算法可以自动调整参数, 提高模型的预测精度, 但由于没有很好地确定采样的起点, 所以采样时间比较缓慢, 使 BPMF 模型不能高效地发挥性能.

2 模型构建

本节主要介绍重新创建的 BPMF 模型并引入新的约束特征向量, 采用贝叶斯诊断和 MCMC

方法产生新的有约束的BPMF模型,针对一些特定用户的极其稀疏的评分矩阵进行预测,实验结果证明了模型的可行性.

2.1 有约束的BPMF (CBPMF)模型 现有的矩阵分解模型在处理评分很少的特定用户时,其用户特征与先验分布平均值相似的用户有接近的特征向量,导致推荐系统面向特定用户的推荐质量严重下降.为了更好地处理这个问题,创建新的CBPMF模型,使针对特定用户的预测评价接近电影的平均评价.引入一种约束特定用户的特征向量的额外方式,便于模型更准确地捕获用户的兴趣特征,提高预测准确率.同时,模型中使用Logistic函数传递特定用户 U 与电影 V 的点积,控制潜在因子间的非线性关系^[18],并将贝叶斯框架应用到模型中,自动调整模型正则化参数.这对于数据过度拟合有很好的优化效果,该模型的创建增加了用户的预测精度和可解释性^[19].

首先介绍模型参数的符号,如表1所示.

CBPMF的模型如图3所示,其核心优化方法是在用户特征向量中引入潜在的相似度约束矩阵来约束用户特征,避免用户特征接近先验分布的平均值.这里设置潜在的相似度约束矩阵 $W \in R^{M \times D}$,用户新的特征向量为:

$$X_i = U_i + \frac{\sum_{k=1}^M I_{ik} W_k}{\sum_{k=1}^M I_{ik}} \quad (11)$$

式(11)表示的用户新的特征向量与式(1)相似,通过对用户 U_i 进行约束产生新的用户特征向量.若用户 i 对物品 k 进行评分则 $I_{ik}=1$,否则为0;然后累加用户 i 对所有物品的评分 W_k ,取平均值与用户 i 的先验分布平均值的偏移值相加得到新的用户特征向量.

根据式(2)的思想,使用Logistic函数传递用户 X 与电影 V 的点积,从而构造新的CBPMF模型的目标函数为:

$$P(R|U, V, W, \alpha) = \prod_{i=1}^N \prod_{j=1}^M \left[\mathcal{N}(R_{ij} | g(X_i^T V_j), \alpha^{-1}) \right]^{I_{ij}} \quad (12)$$

参照式(6)和式(7),为用户和物品的概率分布函数得到约束矩阵向量的概率分布,这里假设 W 服从均值为 μ_w 、方差为 Λ_w^{-1} 的高斯分布:

表1 模型参数的符号

Table 1 Notations of model parameters

变量	描述
R_{ij}	用户 i 对电影 j 的评分
$\Theta_U, \Theta_V, \Theta_W$	用户、电影、约束矩阵的高斯-威沙特分布的超参数
T	采样次数
D	潜在特征向量维度
$\mathcal{N}$	高斯分布函数
$\mathcal{W}$	威沙特分布函数
N, M	分别是用户和电影的个数
I_{ij}	指示变量
W_0, W_1	单位矩阵
μ_U, μ_V, μ_W	用户、电影、约束矩阵的高斯分布的均值参数
$\Lambda_U, \Lambda_V, \Lambda_W$	用户、电影、约束矩阵的高斯分布的方差矩阵
$g(x)$	Logistic函数

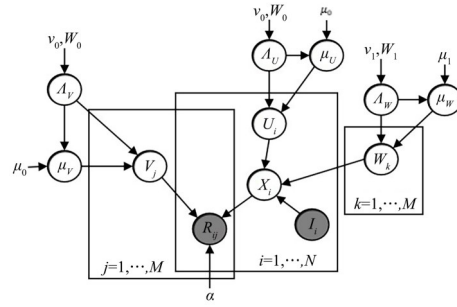


图3 CBPMF模型

Fig. 3 CBPMF model

$$P(W | \mu_w, \Lambda_w) = \prod_{k=1}^M \mathcal{N}(W_k | \mu_w, \Lambda_w^{-1}) \quad (13)$$

同时,设定超参数 $\Theta_w = \{\mu_w, \Lambda_w\}$ 的先验分布为高斯-威沙特分布,可以方便模型训练中后验概率的计算过程.

$$P(\Theta_w | \Theta_0) = P(\mu_w | \Lambda_w) P(\Lambda_w) = \mathcal{N}(\mu_w | \mu_1, (\beta_0 \Lambda_w)^{-1}) \mathcal{W}(\Lambda_w | W_1, v_1) \quad (14)$$

根据式(9)和式(10)的模型参数的初始化进一步初始化模型参数 $\Theta_0 = \{\mu_0, v_0, W_0, \mu_1, v_1, W_1\}$,同时,为了减少优化参数的过程将 $\{\Lambda_U, \Lambda_V, \Lambda_W\}$ 封装在模型内部并设定 $\mu_0 = \mu_1 = 0, v_0 = v_1 = D, W_0$ 和 W_1 为单位矩阵.

模型中用户对电影的预测评分矩阵 R_{ij}^* 的概率目标函数为:

$$P(R_{ij}^*|R, \Theta_0) = \iint P(R_{ij}^*|U_i, V_j, W_k) P(U, V, W|R, \Theta_U, \Theta_V, \Theta_W) P(\Theta_U, \Theta_V, \Theta_W|\Theta_0) d\{U, V, W\} d\{\Theta_U, \Theta_V, \Theta_W\} \quad (15)$$

由于后验分布的求解比较复杂,难以解决评分矩阵的预测精度,这里使用MCMC采样方法对复杂的目标函数进行近似推理:

$$P(R_{ij}^*|R, \Theta_0) \approx \frac{1}{T} \sum_{t=1}^T P(R_{ij}^*|U_i^t, V_j^t, W_k^t) \quad (16)$$

这里 T 表示采样的次数,模型通过马尔可夫链采样获得 $\{U_i^t, V_j^t, W_k^t\}$ 并产生适合模型的参数和超参数 $\{U, V, W, \Theta_U, \Theta_V, \Theta_W\}$ 的后验分布.

2.2 CBPMF 模型推断 在CBPMF模型中,为了易于对模型参数和超参数进行采样,使用贝叶斯推断的方法.由于模型的参数和超参数使用共轭先验的方法从后验分布导出的条件分布进行采样,采样样本数据会直接影响推断精度,所以使用MCMC方法中的吉布斯采样(Gibbs Sampling)进行贝叶斯推断.由于该模型在Pymc3框架上进行搭建,可以方便地使用鲍威尔优化(Powell Optimization)中的 scipy.optimize.fmin_powell 方法快速找到模型的最大后验概率(Maximum A Posteriori, MAP)估计,这样可以快速确定MCMC采样器的起点,节省采样时间.

在对CBPMF模型中的用户进行采样时,在其他条件已经确定的情况下,用户 U_i 的后验概率目标函数:

$$P(U_i|R, V, W, \Theta_U, \alpha) \propto \prod_{j=1}^M \left[\mathcal{N}(R_{ij}|g(X_i^T V_j), \alpha^{-1}) \right]^{I_{ij}} P(U_i|\mu_U, \Lambda_U) \quad (17)$$

为了进一步简化用户 U_i 的后验概率目标函数,对 $g(X_i^T V_j)$ 进行麦克劳林展开:

$$\begin{aligned} \mathcal{N}(R_{ij}|g(X_i^T V_j), \alpha^{-1}) &\approx \\ \mathcal{N}\left(R_{ij} \middle| \left(\frac{1}{2} + \frac{X_i^T V_j}{4} \right), \alpha^{-1}\right) &= \\ \mathcal{N}\left((4R_{ij} - 2) \middle| X_i^T V_j, \left(\frac{\alpha}{4} \right)^{-1} \right) \end{aligned} \quad (18)$$

由于后验分布转换成的条件分布更有利于模型的采样,这里采用共轭先验对参数和超参数进行处理,可以看出用户特征向量 U_i 的条件分布服从高斯分布:

$$\begin{aligned} P(U_i|R, V, W, \Theta_U, \alpha) &\propto \\ P(R|U_i, V, W, \alpha) P(U_i|\Theta_U) &\approx \\ \prod_{j=1}^M \left[\mathcal{N}\left((4R_{ij} - 2) \middle| X_i^T V_j, \left(\frac{\alpha}{4} \right)^{-1} \right) \right]^{I_{ij}} &\cdot \\ \mathcal{N}(U_i|\mu_U, \Lambda_U) &\propto \mathcal{N}\left(U_i \middle| \mu_U^*, [\Lambda_U^*]^{-1} \right) \end{aligned} \quad (19)$$

其中,

$$\Lambda_U^* = \Lambda_U + \frac{\alpha}{4} \sum_{j=1}^M [V_j V_j^T]^{I_{ij}} \cdot \mathcal{N}(U_i|\mu_U, \Lambda_U) \quad (20)$$

$$\mu_U^* = [\Lambda_U^*]^{-1} \left(\frac{\alpha}{4} \sum_{j=1}^M [V_j (4R_{ij} - 2)]^{I_{ij}} + \Lambda_U \mu_U \right) \quad (21)$$

同理,可以对 $P(V_j|R, U, W, \Theta_V, \alpha)$ 进行求解,这里省略了计算过程,现在只需要重新计算 W 的条件后验概率:

$$\begin{aligned} P(W_k|R, U, V, \Theta_W, \alpha) &\propto \\ P(R|W_k, U, V, \alpha) P(W_k|\Theta_W) &\approx \\ \prod_{j=1}^M \left[\mathcal{N}(R_{ij}|g(X_i^T V_j), \alpha^{-1}) \right]^{I_{ij}} &\cdot \\ \mathcal{N}(W_k|\mu_W, \Lambda_W) &\propto \mathcal{N}\left(W_k \middle| \mu_W^*, [\Lambda_W^*]^{-1} \right) \end{aligned} \quad (22)$$

其中,

$$\Lambda_W^* = \Lambda_W + \alpha \sum_{j=1}^M [g(X_i^T V_j)^2]^{I_{ij}} \cdot \mathcal{N}(U_i|\mu_U, \Lambda_U) \quad (23)$$

$$\mu_W^* = [\Lambda_W^*]^{-1} \cdot \left(\alpha \sum_{j=1}^M [g(X_i^T V_j) R_{ij}]^{I_{ij}} + \Lambda_W \mu_W \right) \quad (24)$$

潜在的相似度约束矩阵 W 的超参数 Θ_W 的条件分布可以由高斯-威沙特分布得出:

因为:

$$\begin{aligned} P(\mu_W, \Lambda_W|\Theta_0) &= \\ \mathcal{N}(\mu_W|\mu_1, (\beta_0 \Lambda_W)^{-1}) \mathcal{W}(\Lambda_W|W_1, v_1) \end{aligned}$$

所以:

$$\begin{aligned}
P(\Lambda_w | W, \Theta_0) &= \\
&\mathcal{W}(\Lambda_w W_w^*, v_1 + N) P(\mu_w | \Lambda_w, W, \Theta_0) = \\
&\mathcal{N}\left(\mu_w \left| \frac{\beta \mu_1 + N \bar{W}}{\beta_0 + N}, \frac{(\Lambda_w)^{-1}}{\beta_0 + N} \right.\right) \\
P(\Theta_w | W, \Theta_0) &= \\
&P(\mu_w | \Lambda_w, W, \Theta_0) P(\Lambda_w | W, \Theta_0) = \\
&\mathcal{N}\left(\mu_w \left| \frac{\beta \mu_1 + N \bar{W}}{\beta_0 + N}, \frac{(\Lambda_w)^{-1}}{\beta_0 + N} \right.\right) \mathcal{W}(\Lambda_w W_w^*, v_1 + N)
\end{aligned} \tag{25}$$

其中, $[W_w^*]^{-1} = W_1^{-1} + N \bar{S} + \frac{\beta_0 N}{\beta_0 + N} (\mu_1 - \bar{W})^2$,

$\bar{W} = \frac{1}{N} \sum_{i=1}^N W_i, \bar{S} = \frac{1}{N} \sum_{i=1}^N W_i^2$. 可以容易地获取超参数 Θ_w 的后验概率, 同理可以得到 Θ_U 和 Θ_V 的后验概率, 这就是本模型所有参数的条件概率求证过程.

2.3 算法设计 通过使用 Gibbs 采样对超参数 $\Theta_U, \Theta_V, \Theta_w$ 进行采样, 然后遍历更新得到 U, V, W 向量的特征值, 通过 Logistic 函数传递三者间的点积, 补全稀疏矩阵的空缺评分. 在测试集上进行误差验证, 得出算法的预测精度, 如下所示.

Algorithm CBPMF

Input: 用户向量 U , 电影向量 V , 约束向量 W , 潜在向量维度 D , 采样次数 T .

Output: 预测用户对电影没有评分的空缺评分值.

1. for sampling iterations t of T do
2. 采样超参数 Θ_V $\Theta_V^t \sim P(\Theta_V | V^t, \Theta_0)$
3. 采样超参数 Θ_U $\Theta_U^t \sim P(\Theta_U | U^t, \Theta_0)$
4. 采样超参数 Θ_w $\Theta_w^t \sim P(\Theta_w | W^t, \Theta_0)$
5. for j of M 更新 V 向量 do

$$V_j^{t+1} \sim P(V_j | R, V^t, \Theta_V^t)$$
6. for i of N 更新 U 向量 do

$$U_i^{t+1} \sim P(U_i | R, U^t, \Theta_U^t)$$
7. for k of M 更新 W 向量 do

$$W_k^{t+1} \sim P(W_k | R, W^t, \Theta_w^t)$$
8. end
9. end
10. end

11. for 所有的测试集数据 do
 12. 预测评分并计算预测误差
 13. end
 14. end
-

在计算算法的复杂度时可以看出, $D_{d \times d}$ 维的特征向量的复杂度为 $O(d^3)$, 可以得出 CBPMF 算法的复杂度为 $O(d^3 \sum_{j=1}^M \sum_{i=1}^N I_{ij})$, 与 BPMF 模型算法的复杂度一样. 但是本算法模型对贝叶斯模型进行了优化, 使模型提取潜在特征的能力得到了提升, 所以模型在特征向量维度很低的时候就可以得到很好的预测, 和其他模型算法相比, 本算法的时间复杂度反而更低.

3 实验设计与分析

在不同的数据集上进行实验, 将提出的优化 CBPMF 算法在预测准确性上与 PMF, CPMF 和 BPMF 推荐算法进行比较, 通过标准化的评价指标来证明该 CBPMF 的优越性.

3.1 数据集及评价指标 为了更好地展现 CBPMF 模型, 在 MovieLens 数据集上进行两次实验, 分别使用 MovieLens-100k 和 MovieLens-1M 进行实验. MovieLens-100k 是 943 名用户对 1682 个电影的 100 k 个评分信息, 评分密度为 0.063; MovieLens-1M 是 6039 名用户对 8662 个电影的 1 M 个评分信息, 评分密度为 0.043. 这里对数据进行了预处理以控制数据的稀疏性, 表 2 给出了数据集的相关信息统计.

表 2 数据集的信息统计

Table 2 Statistic information of MovieLens datasets

数据集信息	MovieLens-100k	MovieLens-1M
用户数 U	943	6039
电影数 V	1682	8662
评分范围	1~5	0.5~5.0
评分数	100 k	1 M
评分密度	6.3%	4.3%

为了更好地评价本模型算法的优越性, 使用推荐算法中常用的评价标准 RMSE 和 MAE 来衡量预测误差. RMSE 和 MAE 越小, 算法的准确性越好.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - R_{ij}^*)^2}{\sum_{i=1}^N \sum_{j=1}^M I_{ij}}} \quad (26)$$

$$MAE = \frac{\left| \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - R_{ij}^*) \right|}{\sum_{i=1}^N \sum_{j=1}^M I_{ij}} \quad (27)$$

3.2 模型基线 在对模型基线方法进行选择时,为了给模型预测提供一个好的参照点,本实验分别对随机均匀法、全局均值法和平均均值法进行比较,如图4所示.可以看出,随机均匀法基线效果最差,平均均值法基线效果最好,所以选取平均均值基线作为模型预测实验的最佳基线.

3.3 对比实验 为了进一步证实案例的可解释性,在 MovieLens 数据集上进行两组实验,分别从不同的角度对本模型算法进行性能和可行性测试,最后以 $RMSE$ 为评价指标进行验证.

A 组实验. 为了进一步证明本文的优化算法在更加稀疏的数据集上性能良好,在评分密度为 4.3% 的 MovieLens-1M 数据集上随机选取数据集的 20% 为测试集、80% 为训练集进行测试. 通过选取不同的特征向量维度,检验算法对稀疏矩阵提取潜在特征向量的能力. 设置的特征向量维

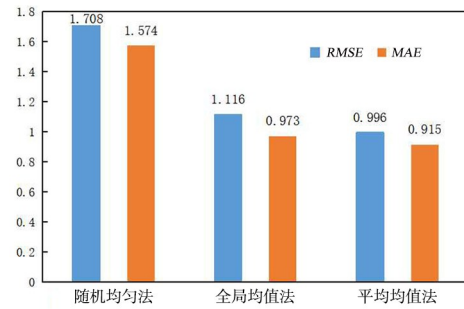


图4 不同的模型基线方法的 $RMSE$ 和 MAE 的比较

Fig. 4 $RMSE$ and MAE of different model baseline methods

度 D 分别为 10, 30, 50, 70, 100, 实验迭代 $epochs=100$ 次, 取算法的 $RMSE$ 和 MAE 的平均值进行比较. 实验的数据对比如表3所示, 表中黑体字为数据的最优值. 可以看出, 在数据稀疏更加明显的情况下, CBPMF 算法的 $RMSE$ 和 MAE 比 BPMF 算法有明显的降低, 分别减少约 0.10 和 0.04. 维度 $D=30$ 时, CBPMF 模型算法的 $RMSE$ 和 MAE 最小, 虽然优势不很明显, 但从整体实验结果可以看出优化算法可以很好地解决数据稀疏的性能, 预测精度也有很大提高.

表3 MovieLens-1M 数据集上 BPMF 和 CBPMF 算法在不同维度下的 $RMSE$ 和 MAE

Table 3 The $RMSE$ and MAE of BPMF and CBPMF in different dimensions on MovieLens-1M dataset

Dataset	Metrics	Method	10D	30D	50D	70D	100D
MovieLens-1M	MAE	BPMF	0.9113	0.9029	0.9186	0.9374	0.9786
		CBPMF	0.8219	0.8087	0.8174	0.8329	0.8610
	RMSE	BPMF	0.9536	0.9495	0.9579	0.9678	0.9889
		CBPMF	0.8561	0.8533	0.8587	0.8613	0.8743

图5和图6为对比数据表3的图形展示. 可以看出, 在传统的 BPMF 模型中引入约束特定用户的特征的额外向量, 可以有效地解决对评分不频繁的用户预测不准确的问题. 在维度很低的情况下 CBPMF 算法的误差值很低, 预测精度更高, 算法的复杂度也更低, 算法的可用性更强. 通过观察可知, 维度 $D=30$ 是模型最优的维度.

B 组实验. 在评分密度为 6.3% 的 MovieLens-100k 数据集上随机选取数据集的 20% 为测试集、80% 为训练集进行实验. 将 CBPMF 算法与 PMF, CPMF 和 BPMF 算法依据 $RMSE$ 进行预测效果对比. 实验中设置参数为 $\alpha=2, \beta_0=2$,

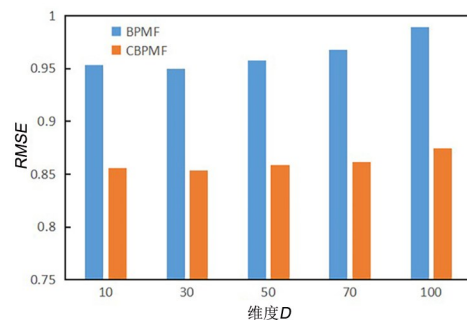


图5 MovieLens-1M 数据集上 BPMF 和 CBPMF 的 $RMSE$ 比较

Fig. 5 The $RMSE$ of BPMF and CBPMF on MovieLens-1M dataset

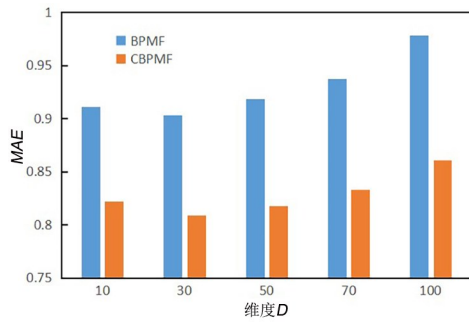


图6 MovieLens-1M数据集上BPMF和CBPMF的MAE比较

Fig. 6 The MAE of BPMF and CBPMF on MovieLens-1M dataset

$\mu_0 = \mu_1 = 0, v_0 = v_1 = D = 30, W_0$ 和 W_1 为维度为 D 的单位矩阵, 采样次数 $T=100$, 迭代次数 $epochs=50$.

实验结果如图7和图8所示. 在MovieLens-100k数据集上, CBPMF的最小RMSE为0.853,

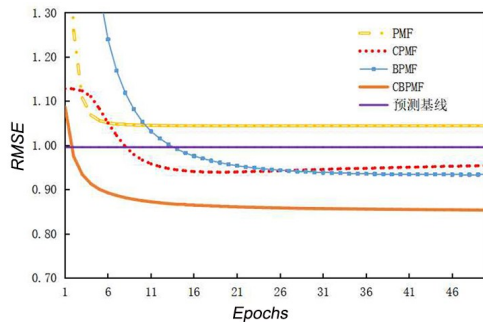


图7 几种算法在MovieLens-100k数据集上的RMSE比较

Fig. 7 RMSE of different algorithms on MovieLens-100k dataset

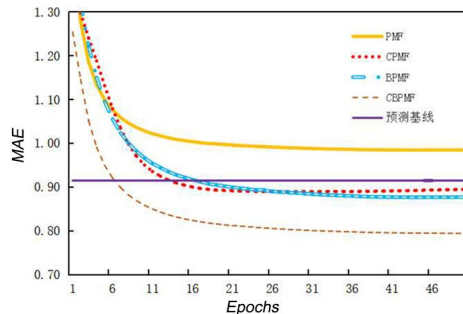


图8 几种算法在MovieLens-100k数据集上的MAE比较

Fig. 8 MAE of different algorithms on MovieLens-100k dataset

而PMF, BPMF和CPMF算法的最小RMSE分别为1.044, 0.933和0.939, 分别提升19.1%, 8%和8.6%. 随着迭代次数的不断增加, CBPMF的RMSE和MAE都比其他算法更低, 并且能很快地达到平稳状态. CBPMF的误差值比三种对比算法更小, 模型的收敛速度也有很大的提升. 综上, CBPMF的预测精度有很大提升.

从以上两组实验结果可看出, 不论是与传统的PMF, CPMF和BPMF矩阵分解算法对比, 还是在模型本身性能上进行验证, 在维度大小不同以及数据集评分密度更小的情况下, 本文提出的CBPMF模型算法的预测准确性都更高. 所以, CBPMF算法对特定用户可以给出更好的用户体验, 而商家针对评分稀疏的特定用户, 也可以更快捷地推荐用户感兴趣的物品, 提高推荐质量. 在模型的可解释性上, 和传统模型相比, CBPMF算法的超参数是采用贝叶斯中的MCMC方法对模型不断更新迭代进行采样获取的最优的模型参数, 所以模型的可解释性更强, 超参数的选取更合理. 完整的贝叶斯定理处理算法模型可以显著提升预言准确性.

3.4 案例分析 为了进一步验证本文优化算法的精确程度是否与用户约束矩阵有关以及是否会影响模型的预测性能, 随机挑选四名用户进行实验对比, 实验结果如表4所示.

表4 案例分析的比较结果

Table 4 Comparative results of case analysis

用户	评分数	无约束矩阵预测误差	有约束矩阵预测误差
		RMSE	RMSE
A	50	94.4%	90.7%
B	14	93.6%	86.6%
C	5	98.2%	82.5%
D	9	96.4%	86.2%

针对模型有无约束矩阵对用户进行约束, 比较预测用户的评分误差RMSE来证实实验的可解释性. 通过对用户的预测误差比较可以看出, 对于特定用户(指用户评分相对稀疏), 约束矩阵发挥了很好的预测效果, 用户评分越稀疏, 有约束矩阵的模型的预测误差越小, 预测准确性越高. 如表4所示, 用户A评论了50个电影, 用户C评论

了 5 个电影,所以用户 C 为特定用户. 有无约束矩阵对模型的评分预测影响很小,预测误差都在 90% 左右,然而用户 C 的预测对于模型有无约束矩阵影响很大. 在模型计算中,对于特定评分稀疏的用户而言,由于传统模型对其预测时预测结果近似物品的平均评分,无法保证评分准确性,而对模型引入约束矩阵可以有效地缓解这一问题. 用户 C 的预测误差 RMSE 提高了 16%,证实了本文提出的优化算法的可解释性.

4 结 论

针对现有的协同过滤算法中的矩阵分解算法不能很好地解决那些不喜欢去评分的特定用户的评分稀疏和推荐质量不高的问题,本文提出了一种受约束的贝叶斯概率矩阵分解优化算法,通过对 PMF 模型引入潜在的相似度约束矩阵来影响特定用户的特征向量,这样可以更快捷地捕获用户之间的相似性以提高预测准确性,同时使用 MCMC 方法对模型进行采样训练,自动调整模型的正则化参数,可以更好地获得更加具有可解释性的超参数. 使用 Pymc3 框架进行模型搭建使模型架构更加简洁,同时使用 MAP 估计来明确 MCMC 采样的起点,节约了采样时间. 在稀疏评分矩阵下,CBPMF 算法喝传统的推荐算法相比,预测评分的准确性得到了提升,优化算法具有很强的预测性能和可解释性. 实验结果表明,在对用户特征向量之外引入潜在的相似度约束矩阵可以有效地解决特定用户评分预测不准确的问题,使推荐系统可以更准确地向特定用户推荐更感兴趣的物品.

参考文献

- [1] Resnick P, Varian H R. Recommender systems. Communications of the ACM, 1997, 40(3): 56—58.
- [2] Huang L W, Fu M S, Li F, et al. A deep reinforcement learning based long-term recommender system. Knowledge - Based Systems, 2021, 213: 106706.
- [3] 王立才, 孟祥武, 张玉洁. 上下文感知推荐系统. 软件学报, 2012, 23(1): 1—20. (Wang L C, Meng X W, Zhang Y J. Context - aware recommender systems. Journal of Software, 2012, 23(1): 1—20.)
- [4] 倪维健, 郭浩宇, 刘彤等. 基于多头自注意力神经网络的购物篮推荐方法. 数据分析与知识发现, 2020, 4(2—3): 68—77. (Ni W J, Guo H Y, Liu T, et al. Online product recommendation based on multi-head self - attention neural networks. Data Analysis and Knowledge Discovery, 2020, 4(2—3): 68—77.)
- [5] Hikmatyar M, Ruuhwan. Book recommendation system development using user-based collaborative filtering. Journal of Physics: Conference Series, 2020, 1477(3): 032024.
- [6] Liu X J, He Q, Tian Y Y, et al. Event-based social networks: Linking the online and offline social worlds//Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, USA: ACM, 2012: 1032—1040.
- [7] 魏晓辉, 孙冰怡, 崔佳旭. 基于图神经网络的兴趣活动推荐算法. 吉林大学学报(工学版), 2021, 51(1): 278—284. (Wei X H, Sun B Y, Cui J X. Interest in activities recommended algorithm based on neural network diagram. Journal of Jilin University (Engineering Science), 2021, 51(1): 278—284.)
- [8] 夏景明, 刘聪慧. 一种基于用户和商品属性挖掘的协同过滤算法. 现代电子技术, 2020, 43(23): 120—123. (Xia J M, Liu C H. A collaborative filtering algorithm based on user and commodity attribute mining. Modern Electronic Technology, 2020, 43(23): 120—123.)
- [9] 孔麟, 黄俊, 马浩等. 融合多层相似度与信任机制的协同过滤算法. 计算机工程与设计, 2020, 41(12): 3405—3411. (Kong L, Huang J, Ma H, et al. Collaborative filtering algorithm fusing multi - level similarity and trust mechanism. Computer Engineering and Design, 2020, 41(12): 3405—3411.)
- [10] 王英博, 孙永荻. 基于 GNN 的矩阵分解推荐算法. 计算机工程与应用, 2020: 1—11. (Wang Y B, Sun Y D. GNN-based matrix factorization recommendation algorithm. Computer Engineering and Application, 2020: 1—11.)
- [11] Koren Y. Factorization meets the neighborhood: A multifaceted collaborative filtering model//Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, USA: ACM, 2008: 426—434.

- [12] Ortega F, Lara-Cabrera R, González-Prieto Á, et al. Providing reliability in recommender systems through Bernoulli Matrix Factorization. *Information Sciences*, 2021(553):110—128.
- [13] 陈珏伊,朱颖琪,周刚等. 基于迁移的联合矩阵分解的协同过滤算法. *四川大学学报(自然科学版)*, 2020, 57(6):1096—1102. (Chen J Y, Zhu Y Q, Zhou G, et al. Collaborative filtering recommendation based on transfer learning and joint matrix decomposition. *Journal of Sichuan University (Natural Science Edition)*, 2020, 57(6):1096—1102.)
- [14] Salakhutdinov R, Mnih A. Probabilistic matrix factorization//*Proceedings of the 20th International Processing Conference on Neural Information Processing Systems*. New York, NY, USA: Curran Associates Inc., 2007:1257—1264.
- [15] Salakhutdinov R, Mnih A. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo//*Proceedings of the 25th International Conference on Machine Learning*. New York, NY, USA: ACM, 2008:880—887.
- [16] Yang Y, Gao X G, Chen D Q, et al. Learning Bayesian networks using the constrained maximum a posteriori probability method. *Pattern Recognition*, 2019(91):123—134.
- [17] 毛宜钰,刘建勋,胡蓉等. 基于Logistic函数和用户聚类的协同过滤算法. *浙江大学学报(工学版)*, 2017, 51(6):1252—1258. (Mao Y Y, Liu J X, Hu R, et al. Collaborative filtering algorithm based on logistic function and user clustering. *Journal of Zhejiang University (Engineering Science)*, 2017, 51(6):1252—1258.)
- [18] Ning X, Karypis G. SLIM: sparse linear methods for top - N recommender systems//*2011 IEEE 11th International Conference on Data Mining*. Vancouver, Canada: IEEE, 2011:497—506.
- [19] 吴宾,娄铮铮,叶阳东. 联合正则化的矩阵分解推荐算法. *软件学报*, 2018, 29(9):2681—2696. (Wu B, Lou Z Z, Ye Y D. Co-regularized matrix factorization recommendation algorithm. *Journal of Software*, 2018, 29(9):2681—2696.)

(责任编辑 杨可盛)