

融合邻域结构信息的概率矩阵分解算法

张洪为

摘 要:为解决概率矩阵分解算法未利用用户/物品的邻域结构信息问题,该文提出一种融合邻域结构信息的概率矩阵分解算法,该算法通过将概率矩阵分解与包含用户和物品邻域结构信息的图正则化项,整合到一个统一的优化问题中,以进一步提炼用户和物品的潜在因子,提高评分预测的精度.在两个真实数据集上的实验结果表明,所提算法能够实现推荐性能的显著提升.

关键词:推荐系统;协同过滤;概率矩阵分解;邻域结构信息;图正则化

中图分类号:TP391

文献标志码:A

文章编号:1008-7974(2022)08-0050-04

DOI:10.13877/j.cnki.cn22-1284.2022.08.008

随着 Web2.0 技术的迅速发展,在线信息呈爆炸式增长,推荐系统已成为帮助用户有效找到所需信息的重要工具^[1].推荐系统旨在为用户提供符合他们偏好的物品,如产品或服务.协同过滤是推荐系统中最流行的技术^[2],它利用收集到的具有相似行为的其他用户的历史评分记录为目标用户生成推荐^[3].早期的协同过滤方法主要是基于邻域的方法^[4],其更关注物品之间或用户之间的关系,现在已逐渐演化为基于模型的方法,涉及机器学习技术,如矩阵分解^[5-7],概率模型^[8]和深度神经网络^[9]等.概率矩阵分解(Probabilistic Matrix Factorization, PMF)^[6]是一种最具代表性的矩阵分解方法,它假设观测到的评分服从高斯分布,然后通过最大化用户和物品

特征的对数后验,来学习用户和物品特征. PMF 方法的实质是将用户/物品评分矩阵分解为两个低秩矩阵,然后利用它们的乘积来对用户/物品评分矩阵进行重构,进而实施推荐.然而,PMF 方法仅利用了用户与物品的交互信息,没有利用用户/物品的邻域结构信息,即用户与用户以及物品与物品的相似性关系,限制了模型的推荐性能.

为解决上述问题,本文提出一种融合邻域结构信息的概率矩阵分解算法(简称 GPMF),该算法将 PMF 模型与包含用户和物品邻域结构信息的图正则化项整合到一个统一的优化问题中.然后利用一种迭代优化算法对该优化问题进行求解.在两个真实数据集上的实验结果验证了该方法的有效性.

收稿日期:2022-05-14

作者简介:张洪为,黑龙江绥化人,博士,通化师范学院数学学院讲师(吉林 通化 134001).

1 概率矩阵分解

给定一个用户-物品评分矩阵 $R \in \mathbb{R}^{M \times N}$, 其中 R_{ij} 表示第 i 个用户对第 j 个物品的评分, 推荐系统的任务是对 R 中的缺失评分进行预测, 然后将预测评分高的物品推荐给相应的用户. 概率矩阵分解算法是一种有效的协同过滤算法, 其目标是将 R 分解为两个低秩矩阵 $U \in \mathbb{R}^{M \times k}$ 和 $V \in \mathbb{R}^{N \times k}$, 分别称为用户的潜在偏好矩阵和物品的潜在特征矩阵, 其中 k 表示用户的潜在偏好个数或物品的潜在特征个数. 求解 U 和 V 等价于求解如下优化问题

$$\min \frac{1}{2} \|I \odot (R - UV^T)\|_F^2 + \frac{\lambda}{2} \|U\|_F^2 + \frac{\lambda}{2} \|V\|_F^2, \quad (1)$$

其中: I 是指示矩阵, 表示用户是否对物品进行了评分, 如果第 i 个用户对第 j 个物品进行了评分, 则 $I_{ij} = 1$, 否则 $I_{ij} = 0$. \odot 表示矩阵按元素相乘. $\|\cdot\|_F$ 为 Frobenius 范数. λ 为控制约束对模型复杂度影响的正则化参数.

通过对 U 和 V 进行梯度下降, 能够找到式 (1) 的局部最小值. 然后, 推荐可由下式给出

$$\hat{R} = UV^T, \quad (2)$$

其中: \hat{R} 表示对用户-物品评分矩阵 R 的重构. 将 \hat{R} 中预测评分高的物品推荐给用户.

2 融合邻域结构信息的概率矩阵分解

概率矩阵分解方法的本质是利用用户的潜在偏好向量 $U_{i\cdot}$ (U 的第 i 行) 与物品的潜在特征向量 $V_{\cdot j}$ 的内积来对 R 中的未知评分进行建模, 它仅利用了用户与物品的交互信息, 而没有使用用户与用户, 以及物品与物品的相似性关系, 限制了模型的推荐性能. 本文将用户图和物品图的图正则化整合到概率矩阵分解模型中, 以进一步提升模型的推荐性能.

2.1 图正则化

图正则化^[10-11]已被广泛应用于降维、聚类和半监督学习中, 融合邻域结构信息的用户图和物品图的图正则化可分别定义为:

$$\mathcal{R}(U) = \frac{1}{2} \sum_{ij} (W_U)_{ij} \|U_{i\cdot} - U_{j\cdot}\|^2 = \text{tr}(U^T L_U U), \quad (3)$$

$$\mathcal{R}(V) = \frac{1}{2} \sum_{ij} (W_V)_{ij} \|V_{\cdot i} - V_{\cdot j}\|^2 = \text{tr}(V^T L_V V), \quad (4)$$

其中: $\text{tr}(\cdot)$ 为矩阵的迹, $U_{i\cdot}$ 和 $V_{\cdot j}$ 分别表示 U 的第 i 行和 V 的第 j 行, $L_U = D_U - W_U$ 和 $L_V = D_V - W_V$ 分别为用户图和物品图的拉普拉斯矩阵, D_U 和 D_V 分别为以矩阵 W_U 和 W_V 的行和为主对角线元素的对角矩阵, 可表示为 $D_U = \text{diag}(\sum_j W_U)$, $D_V = \text{diag}(\sum_j W_V)$, W_U 和 W_V 分别为用户之间和物品之间的余弦相似度矩阵, 计算公式如下:

$$(W_U)_{ij} = \frac{R_{i\cdot} R_{j\cdot}^T}{\|R_{i\cdot}\|_2 \cdot \|R_{j\cdot}\|_2}, \quad (5)$$

$$(W_V)_{ij} = \frac{R_{\cdot i} R_{\cdot j}^T}{\|R_{\cdot i}\|_2 \cdot \|R_{\cdot j}\|_2}, \quad (6)$$

式中: $R_{i\cdot}$ 和 $R_{\cdot j}$ 分别表示用户-物品评分矩阵 R 的第 i 行和第 j 列.

2.2 图正则化概率矩阵分解

将包含用户和物品邻域结构信息的图正则化项引入到概率矩阵分解模型中, 能使模型在学习的过程中保持用户之间和物品之间固有的相似性信息, 进而提升推荐的性能. 为此, 将图正则化项整合到概率矩阵分解模型中, 得如下优化问题

$$\min \frac{1}{2} \|I \odot (R - UV^T)\|_F^2 + \frac{\alpha}{2} (\text{tr}(U^T L_U U) + \text{tr}(V^T L_V V)) + \frac{\lambda}{2} (\|U\|_F^2 + \|V\|_F^2), \quad (7)$$

其中: α 为控制图正则化对概率矩阵分解影响的图正则化参数.

利用梯度下降法可得学习模型的迭代算法如下:

$$U \leftarrow U - \eta((I \odot (UV^T - R))V + \alpha L_U U + \lambda U), \quad (8)$$

$$V \leftarrow V - \eta((I \odot (UV^T - R))^T U + \alpha L_V V + \lambda V), \quad (9)$$

其中: η 为学习率.

3 实验

3.1 数据集

本文采用两个真实的数据集 Netflix 和 MovieLens20M 来评价所提出的基于图正则化的概率矩阵分解算法. Netflix 数据集包含了 100 480 507 个评分, 由 480 189 名用户对 17 770 部电影上给出, 其值包含在集合 $\{1, 2, 3, 4, 5\}$ 中. MovieLens20M 数据集包含 20 000 263 个评

分, 由 138 493 名用户在 26 744 部电影上给出, 其值包含在集合 $\{0.5, 1, 1.5, \dots, 5\}$ 中, 本文将其规范化到 $\{1, 2, 3, 4, 5\}$. 随机提取 Netflix 或 MovieLens20M 数据集上 5 000 名用户对 5 000 部电影的密集评分矩阵 R , 然后将 R 随机划分为训练集 R_{tr} 和测试集 R_{te} , 分别包含 50% 的评分. 让 R_{te} 保持不变, 分别从 R_{tr} 中随机提取 1%, 1.5% 和 2% 的评分用于训练.

3.2 评价标准

本文采用广泛使用的平均绝对误差 (MAE) 和均方根误差 (RMSE) 作为评价标准, 分别定义为

$$MAE = \sum_{(u,i,R_{ui}) \in R_E} |R_{ui} - \hat{R}_{ui}| / |R_E|, \quad (10)$$

$$RMSE = \sqrt{\sum_{(u,i,R_{ui}) \in R_E} (R_{ui} - \hat{R}_{ui})^2 / |R_E|}, \quad (11)$$

其中: R_{ui} 和 \hat{R}_{ui} 分别表示真实评分和预测评分, $|R_E|$ 为测试评分数. MAE 和 RMSE 越小, 代表算法的推荐性能越好.

表 1 三种方法在 Netflix 子集上的性能比较

评价标准	方法	密集度			平均
		1%	1.5%	2%	
MAE	AG	0.881 7±0.000 2	0.881 7±0.000 1	0.881 6±0.000 2	0.881 7±0.000 2
	PMF	0.720 5±0.002 0	0.696 3±0.000 7	0.672 6±0.001 2	0.696 4±0.001 3
	GPMF	0.704 6±0.001 1	0.685 2±0.000 6	0.666 3±0.000 8	0.685 4±0.000 9
RMSE	AG	1.065 6±0.000 1	1.065 7±0.000 1	1.065 7±0.000 1	1.065 6±0.000 1
	PMF	0.920 7±0.001 7	0.892 6±0.001 2	0.870 4±0.002 1	0.894 6±0.001 7
	GPMF	0.898 5±0.001 5	0.871 5±0.000 9	0.860 1±0.001 7	0.876 7±0.001 4

表 2 三种方法在 MovieLens20M 子集上的性能比较

评价标准	方法	密集度			平均
		1%	1.5%	2%	
MAE	AG	0.880 3±0.000 2	0.880 4±0.000 2	0.880 3±0.000 2	0.880 3±0.000 2
	PMF	0.682 8±0.001 5	0.659 8±0.000 5	0.651 0±0.001 4	0.664 5±0.001 1
	GPMF	0.671 1±0.001 0	0.655 6±0.000 6	0.644 0±0.000 9	0.656 9±0.000 8
RMSE	AG	1.055 3±0.000 3	1.055 4±0.000 3	1.055 3±0.000 3	1.055 3±0.000 3
	PMF	0.877 5±0.001 6	0.851 0±0.000 7	0.847 6±0.001 6	0.858 7±0.001 2
	GPMF	0.863 0±0.001 5	0.842 1±0.000 8	0.836 0±0.000 7	0.847 0±0.001 0

3.3 实验结果与分析

本文使用 AG 方法(即利用观测评分的平均值来预测未知评分)及 PMF 方法与所提出的 GPMF 方法进行对比,三种方法在 Netflix 子集和 MovieLens20M 子集上的对比结果如表 1 和表 2 所示.从中可以看到,相比于对比方法,所提出的 GPMF 方法在两个真实数据集的三个不同密集度上,实现了最好的推荐性能.验证了将包含用户/物品邻域结构信息的图正则化项整合到概率矩阵分解模型中,能够进一步提炼用户和物品的潜在因子,实现推荐性能的提升.

4 结语

本文针对概率矩阵分解算法未利用用户/物品邻域结构信息影响推荐性能的问题,提出一种融合邻域结构信息的概率矩阵分解算法,该算法将概率矩阵分解模型与包含用户和物品邻域结构信息的图正则化项整合到一个统一的优化问题中,以对用户和物品潜在因子作进一步提炼,实现推荐性能的提升.在两个真实数据集上的实验结果验证了所提算法的有效性和稳定性.

参考文献:

- [1] BOBADILLA J, ORTEGA F, HERNANDO A, et al. Recommender systems survey [J]. Knowl-based syst, 2013, 46: 109-132.
- [2] SARWAR B, KARYPIS G, KONSTAN J, et al. Item-based collaborative filtering recommendation algorithms [C]//International Conference on World Wide Web, Kong Hong, China, 2001: 285-295.
- [3] KOREN Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model [C]//ACM SIGKDD International Conference on Knowledge Discovery and Data mining, Las Vegas, Nevada, USA, 2008: 426-434.
- [4] HERLOCKER J L, KONSTAN J A, BORCHERS A, et al. An algorithmic framework for performing collaborative filtering [C]//ACM SIGIR conference on Research and development in information retrieval, Berkeley California USA, 1999: 230-237.
- [5] ZHANG S, WANG W, FORD J, et al. Learning from incomplete ratings using non-negative matrix factorization [C]//SIAM International Conference on Data Mining, Bethesda, MD, USA, Society for Industrial and Applied Mathematics, 2006: 549-553.
- [6] MNIH A, SALAKHUTDINOV R R. Probabilistic matrix factorization [C]//Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 2008: 1257-1264.
- [7] CHEN G, WANG F, ZHANG C. Collaborative filtering using orthogonal nonnegative matrix tri-factorization [J]. Inform Process Manag, 2009, 45(3): 368-379.
- [8] LIU B, XIONG H, PAPADIMITRIOU S, et al. A general geographical probabilistic factor model for point of interest recommendation [J]. IEEE T Knowl Data En, 2014, 27(5): 1167-1179.
- [9] LU J, XUAN J, ZHANG G, et al. Structural property-aware multilayer network embedding for latent factor analysis [J]. Pattern Recogn, 2018, 76: 228-241.
- [10] SMOLA A J, KONDOR R. Kernels and regularization on graphs [C]//Learning theory and kernel machines. Springer, Berlin, Heidelberg, 2003: 144-158.
- [11] CAI D, HE X, HAN J, et al. Graph regularized nonnegative matrix factorization for data representation [J]. IEEE T Pattern Anal, 2011, 33(8): 1548-1560.

(责任编辑:陈衍峰)