

矩阵分解技术在电影推荐系统中的应用

蔡崇超^{1,2}, 许华虎¹

(1. 上海大学 计算机工程与科学学院, 上海 200444; 2. 湖州职业技术学院 物流与信息工程学院, 浙江 湖州 313000)

摘要:随着大数据、移动互联网的快速发展,推荐系统成为解决网络信息过载的有力工具。为解决传统推荐系统由于没有将社交网络中用户关系考虑进去而导致的稀疏矩阵、冷启动等问题,提出一种基于矩阵分解技术的电影推荐系统算法MFMRs。该算法充分考虑到社交网络中用户之间的关系对推荐结果的影响,通过设置特征参数、损失函数、随机梯度下降等方法对推荐系统的精度进行改进。结果表明,通过应用该算法,Douban数据集的精度提升62%,Netflix数据集的精度提升51%。

关键词:推荐系统;社交网络;矩阵分解;梯度下降

DOI:10.11907/rjdk.201996

开放科学(资源服务)标识码(OSID):



中图分类号:TP319

文献标识码:A

文章编号:1672-7800(2021)001-0174-04

Application of Matrix Factorization in Movie Recommendation System

CAI Chong-chao^{1,2}, XU Hua-hu¹

(1. School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China;

2. College of Logistic and Information Engineering, Huzhou Vocational & Technical College, Huzhou 313000, China)

Abstract: With the rapid development of big data, recommendation system has become a powerful tool to solve network information overload. The main purpose of this paper is to solve the problem of sparse matrix and cold start in traditional recommendation system, which does not take the user relationship into account. In this paper, an algorithm of movie recommendation system based on matrix factorization is proposed. The algorithm fully considers the influence of the relationship between users in social network on the recommendation results, and solves the above problems by setting feature parameters, gradient descent and offset variables. The results show that the accuracy of Douban dataset and Netflix dataset is improved by 62% and 51% respectively by using this algorithm.

Key Words: recommendation system; social network; matrix factorization; gradient descent

0 引言

为解决社交网络信息过载问题,需引入用户或内容推荐系统。传统的推荐系统以协同过滤算法为主,该算法没有考虑到社交网络中不同用户之间的关联关系对推荐结果的影响。并且,在社交网络研究过程中发现,大多数用户对于内容的评论较少,绝大多数内容由少数人产生,即存在着稀疏矩阵问题。

本文主要研究基于社交网络关系的推荐系统在电影领域的应用情况。随着移动互联网在人们日常生活中的全面渗透,互联网公司往往应用千人千面技术增加用户粘度,在影视领域这种情况更加普遍。

相比之下,电影点评推荐相关研究开展得较早。目

前,在推荐系统领域较为成熟的电影数据集有很多,例如 movielens^[1]、Netflix^[2]、Douban^[3],以上都是推荐系统较为常用的数据集,本实验主要在 Douban 和 Netflix 上进行。

协同过滤推荐算法^[4-5]在电影推荐领域较为常见,该算法通过记录用户行为预测用户喜好,进而完成电影推荐任务。王骏等^[4]通过改进神经协同过滤模型,利用多层感知机的非线性特征处理提取隐含高阶特征信息以及贝叶斯个性化排序算法提取排序信息,使推荐更加精准,但是该方法忽略了社交网络中用户之间关系的影响力。随着社交网络的兴起,用户之间有了更加复杂的关联关系,这使得单纯的协同过滤算法推荐精度有所下降。

在社交网络中用户之间的社交关系和用户发布的内容往往有很强的关联性^[6-7],人们总是倾向于关注与自己兴趣点近似的人群。Koren 等^[2]首次将矩阵分解技术引入

收稿日期:2020-08-18

作者简介:蔡崇超(1983-),男,上海大学计算机工程与科学学院博士研究生,湖州职业技术学院物流与信息工程学院副教授,研究方向为自然语言处理、社交网络、人工智能;许华虎(1968-),男,博士,上海大学计算机工程与科学学院教授、博士生导师,研究方向为人工智能、大数据、图像处理

电影推荐系统中,不过其当时的数据集较小,同时仅仅将其应用到了一个数据集,本文实验扩展了 Koren 等的数据集规模。肖晓丽等^[3]通过引入信任机制,通过定义直接信任、间接信任、传递路径和计算方法度量社交网络用户之间隐含的信任值,将社交网络转换为信任网络,为推荐系统中基于用户关系提升推荐精度提供了理论依据。

本文的创新之处在于,传统的电影推荐系统主要以协同过滤算法推荐为主,很少将社交网络信息作为重要参考,本文通过矩阵分解技术^[2,8-9]设置特征参数、损失函数、随机梯度下降等方法对推荐系统的精度进行改进,同时融合社交网络用户信息关系,进而提升推荐系统精度。由实验结果可知,在两个实验数据集上精度分别提高 62%、51%。

1 相关问题

社交网络是一个巨大的实时信息传播平台,根据用户发布的内容寻找相似用户,可以找到自己感兴趣的电影或内容,Massa 等^[10]于 2004 年提出将社交网络中的相互关系融入推荐算法中构建推荐模型。Lin & Gemmis 等^[11-12]通过构建用户—项目评分矩阵进行用户相似度计算,以提升推荐系统精确度。

传统的推荐系统假设用户是独立且分布相同,这种假设忽略了使用者的社交关系。在类似于电影社交网络领域,网络中的不确定关系往往成为影响推荐精度的重要因素,如果两个用户兴趣相同,观点相近,那么认为他们之间的关联关系会大于那些具有不同兴趣和关注内容的人。如果两个人喜欢电影的个数超过一定比例,则认为他们有相同的欣赏品味,这种社交网络关系可以在很大程度上影响推荐结果。

1.1 矩阵分解技术

对于一个电影推荐系统而言,其用户和电影之间的关系转换为一个 user-item 矩阵。矩阵中的行代表用户,列代表电影。若用户对电影有过评分,则矩阵中处于用户对应行与电影对应列的交叉位置表示用户对电影的评分,该矩阵被称为评分矩阵。通过矩阵分解技术^[13-14],可以将 user-item 评分矩阵分解为 2 个低秩的用户电影矩阵,同时降低计算复杂度。利用线性回归思想,通过最小化观察数据的平方寻求最优用户和项目的隐含向量表示,进而用于预测缺失评分。

1.2 梯度下降算法

求解矩阵分解最优化问题时,本文采用梯度下降法(Gradient Descent),其核心思想是沿梯度下降方向逐步迭代。梯度是一个向量,表示一个函数在该点处沿梯度方向变化最快,变化率最大,而梯度下降方向指负梯度方向。在实际实验过程中采用随机梯度下降算法,随机梯度下降算法^[15-17]指在迭代过程中随机选择一个或几个样本的梯度替代总体梯度,从而极大降低了计算复杂度。本文求解目标函数最小化方法使用的就是随机梯度下降算法。

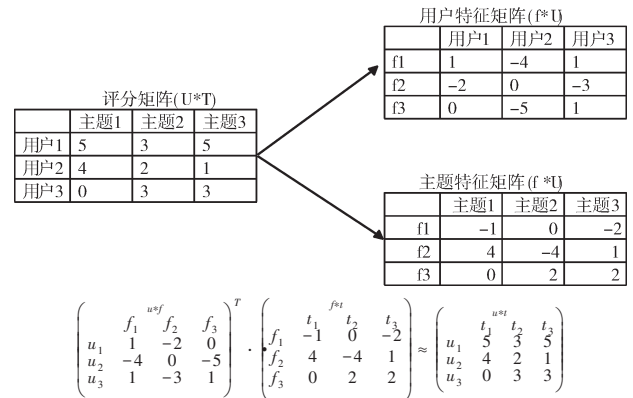


Fig. 1 Relationship among user, topic and latent factor matrix

图 1 用户、主题、潜在因子矩阵之间的关系

2 基于矩阵分解技术的电影推荐系统

矩阵分解技术用途十分广泛,基于矩阵分解技术的电影推荐系统可以解决传统推荐系统技术中存在的矩阵稀疏、冷启动等问题。本文首先分析基础矩阵分解技术,然后考虑损失函数构建,最后通过梯度下降算法进行最小值求解。

2.1 基础矩阵分解技术

通过矩阵分解技术构建了一个用户—电影矩阵。在该推荐系统中用户集合 $U=\{U_1, U_2, \dots, U_m\}$ 。电影集合 $M=\{M_1, M_2, \dots, M_m\}$, 用户对电影的评分矩阵 $R=\{R_{ut}\}$, 其中, R_{um} 表示用户 u 对电影 m 的评分,每一部电影都得到一个向量 q , 每一个用户也得到一个向量 p , 如式(1)所示。

$$\hat{R} = P^T Q \quad (1)$$

为了防止过拟合问题,加入正则化 $\lambda(\|p_u\|^2 + \|q_i\|^2)$, λ 为正则化参数。最终得到求解公式如式(2)所示。

$$\min_{p,q} = \frac{1}{2} \sum_{(u,i) \in K} (R_{u,i} - p_u^T q_i)^2 + \lambda(\|p_u\|^2 + \|q_i\|^2) \quad (2)$$

接下来利用求最小值方法计算出新评分矩阵的各项参数值。

2.2 损失函数构建

当在电影社交网络中的两个用户同时讨论一个电影,其中一个较为偏激,针对某一个话题经常表达较为悲观的情绪,而另一个用户则显示出较为正面的情绪。在电影选取过程中,由于电影分布不同也可能存在偏差。对于用户而言,不同的用户针对相同电影表现出的情感倾向性的激烈程度并不相同。此外,同一个用户针对不同的电影时,表现出的情感倾向性也不同。为此,通过 Koren 等^[2]的研究表明,在推荐系统中考虑用户和主题偏差可以有效改进系统的推荐精度。矩阵分解模型通过明确考虑以下偏差参数解决这些问题,如式(3)所示。

$$b_{u,i} = \mu + b_u + b_i \quad (3)$$

其中, b_u 代表了用户偏差, b_i 代表电影偏差。将偏差计算整合到式(1)中,得到公式如式(4)所示。

$$\min_{p,q} = \frac{1}{2} \sum_{(u,i) \in K} (R_{u,i} - \mu - b_u - b_i - p_u^T q_i)^2 + \gamma(b_u^2 + b_i^2 + \|p_u\|^2 + \|q_i\|^2) \quad (4)$$

2.3 最小值求解

建立评分矩阵和损失函数后,需通过优化算法进行最小值求解。本文使用随机梯度下降算法进行最小值求解,过程如式(5)一式(8)所示。

对于目标函数式(4),对其进行求梯度,如式(5)一式(6)所示。

$$\frac{\partial(R_{ui} - \hat{R}_{ui})}{\partial P_{uk}} = -2(R_{ui} - \sum_{k=1}^K P_{uk} Q_{ki})Q_{ki} + 2\lambda P_{uk} \quad (5)$$

$$\frac{\partial(R_{ui} - \hat{R}_{ui})}{\partial Q_{ki}} = -2(R_{ui} - \sum_{k=1}^K P_{uk} Q_{ki})P_{uk} + 2\lambda Q_{ki} \quad (6)$$

利用梯度下降进行最小值求解,如式(7)一式(8)所示。

$$P_{uk} = P_{uk} + \alpha(R_{ui} - \sum_{k=1}^K P_{uk} Q_{ki})Q_{ki} - \lambda P_{uk} \quad (7)$$

$$Q_{ki} = Q_{ki} + \alpha(R_{ui} - \sum_{k=1}^K P_{uk} Q_{ki})P_{uk} - \lambda Q_{ki} \quad (8)$$

通过上述随机梯度算法,对分解后的矩阵进行参数估计,得到相应的矩阵参数。

2.4 算法流程

本文首先对数据进行抽取式整理,其次根据测试数据比例进行实验,设置矩阵分解模型参数并进行不同角度的观察,最后利用梯度下降算法得到最优解。流程如图2所示。

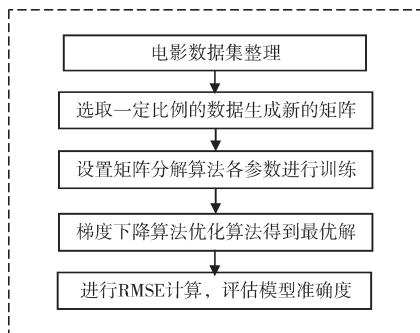


Fig. 2 Process of algorithm

图2 算法过程

4 实验设计与分析

4.1 数据集分析

本文选择 Douban 电影评论和 Netflix 数据集作为社会推荐的参考数据集。

Douban 是一个以电影评论为主的社交网站。用户注册后,能够对其已经发布的电影或即将发布的电影发布评论和打分,评分登记是 1~5 的整数。这些评价分数和评论会影响其它网站用户,是一部电影是否值得观看的依据。网站用户之间可以有关注和被关注的关系,评论内容有短评和长评。本文实验中使用的数据集来源于豆瓣网站,tsv 文件总大小为 46.8M,包括 94 890 个用户,共有 81 906 个不同项目,每个用户至少有一次评论,总共有 11 742 260 条评论。在实验过程中为了考察不同数据集大小情况,分别选取 K 篇文章的所有评论进行试验,K 的大小分别为 5、10、15、20、25。选取这些数据内容是为了与 Netflix 数据集规模保持一致,方便对比。

Netflix 数据集由 Netflix 网站提供,该数据总计有 100 万用户评论,其中评论用户 500 000,电影总计 17 000 部。每一个点评的评分在 1~5 区间。在实验过程中,分别选取 10 000、25 000、50 000、75 000、100 000 条评论数据作为试验数据。数据集规模如表 1 所示,其中豆瓣电影数据集以讨论的电影数目进行分类,Netflix 数据集以评论数据进行分类。从整体看,评论数目的规模较为相近。

Table 1 Dataset size

表 1 数据集规模

数据集名称	电影数	评论总数	数据集规模
Douban	5	45 872	888KB
Douban	10	70 220	1.32MB
Douban	15	90 819	1.71MB
Douban	20	111 793	2.11MB
Douban	25	133 606	2.51MB
Netflix	8	10 000	200KB
Netflix	17	25 000	502KB
Netflix	26	50 000	0.98MB
Netflix	28	75 000	1.47MB
Netflix	30	100 000	1.96MB

4.2 评价指标与模型参数

采用均方根误差 RMSE 和评价绝对误差 MAE 对试验结果进行评价。这两个评价指标的值越低表示预测准确度越高。

$$RMSE = \sqrt{\frac{1}{|T|} \sum_{(u,i) \in T} (\hat{r}_{ui} - r_{ui})^2} \quad (9)$$

矩阵分解模型是本次实验的主要贡献,除原本矩阵外,主要参数有 4 个,如表 2 所示。

Table 2 Parameters and values

表 2 参数含义和取值范围

参数名称	参数含义	取值范围
K	潜在用户特征数量	2~10
alpha	学习率	0.01
beta	正则化参数	0.01
iterations	迭代次数	10~100

其中,K 表示潜在用户特征向量个数,将其取值范围设定为 2~10。

Alpha 表示学习率(learning rate),代表梯度下降算法中迭代步长。学习率一般不宜过大,过大时,迭代过程会出现震荡现象。在实验过程中,将其值设置为 0.01。

Beta 表示过拟合参数,在实验过程中将其设置为 0.01。在进行梯度下降算法求解过程中可能会造成过拟合问题,所谓过拟合,就是过分地拟合了样本数据,造成引入了大量噪音,无法呈现原本规律,但又无法针对特殊样本数据进行剔除,因此干脆对所有参数加入随机因子,即正则项。正则化是降低过拟合问题的常见手段。

Iterations 表示迭代次数,迭代次数越多,准确度越高,但是相应的计算开销也会增大,在实验过程中将其值的范围设定为 10~100。

4.3 实验结果与分析

依据上述数据集、评价指标和模型参数设置等内容,

通过实验观察该模型如何改变推荐精度。

首先将本文提出的模型应用于豆瓣数据集中,数据集规模按照评论个数划分为45 872、70 220、90 819、111 793、164 122。具体结果如表3所示。

Table 3 Evaluation values

表3 评测值

数据集	评论数	RMSE	MF-RMSE	数据集	评论数	RMSE	MF-RMSE
Douban	45 872	1.125	0.412	Netflix	10 000	0.540	0.302
Douban	70 220	0.894	0.338	Netflix	25 000	0.368	0.190
Douban	90 819	0.792	0.302	Netflix	50 000	0.315	0.161
Douban	111 793	0.682	0.298	Netflix	75 000	0.318	0.153
Douban	133 606	0.643	0.296	Netflix	100 000	0.312	0.148

RMSE 的值表示对原始电影评论集设置测试用例比例后得到的 RMSE 误差,MF-RMSE 表示应用本文提到的矩阵分解技术后得到的 RMSE 值。可以看出,MF-RMSE 对于推荐精度有了较大提升。同时还可以看出,随着评论数即样本数据的增加,无论是 Douban 数据集还是 Netflix 数据集,评论精度有较大幅度提升,具体如图3和图4所示。

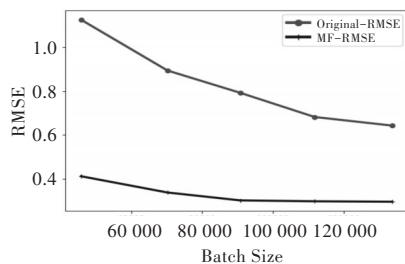


Fig. 3 Douban dataset

图3 Douban数据集

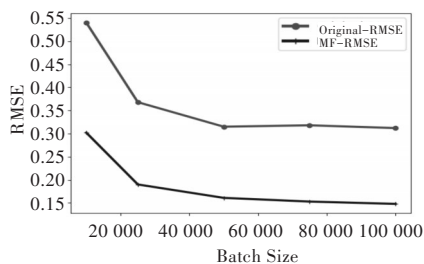


Fig. 4 Netflix dataset

图4 Netflix数据集

通过上述实验内容可以看出,使用矩阵分解技术前后,推荐系统的准确度有了相应提升。随着数据集的增加,推荐精度也会有相应增加,但是增长幅度会有所减小,数据集规模达到一定程度后,要提高精度有一定难度。上述两个实验虽然基于不同的数据集,但是数据规模较为相似,得到的数据结论也较为相似,因此本文提出的算法颇具代表性。

5 结语

本文提出了一种矩阵分解算法,将其应用到电影推荐系统中。该算法通过融合用户和电影内容之间的评分机制,利用矩阵分解技术进行内容推荐,首先将用户—电影评分矩阵分解为用户潜在因子矩阵和主题潜在因子矩阵,

然后引入损失函数防止出现过拟合现象,最后利用梯度下降算法进行最优值求解。将该方法应用于 Douban 电影数据集和 Netflix 电影数据集后取得了较好效果。在未来工作中,可将用户之间的关联关系和动态内容作为一个重要因素整合到模型中,以更好地提高推荐系统精度。

参考文献:

- [1] ADOMAVICIUS G, TUZHILIN A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions[J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(6): 734-749.
- [2] KOREN Y, BELL R, VOLINSKY C. Matrix factorization techniques for recommender systems[J]. Computer, 2009, 42(8):30-37.
- [3] XIAO X L, QIAN Y L, LI D J, et al. Clustering recommendation algorithm based on user interest and social trust[J]. Journal of Computer Applications, 2016, 36(5): 1273-1278.
- [4] WANG J, YU G. Movie recommendation system based on improved neural collaborative filtering model[J]. Computer Engineering and Design, 2020, 41(7): 2069-2075.
- [5] YEHUDA K. Factorization meets the neighborhood: a multifaceted collaborative filtering model[C]. In Proceedings of SIGKDD, 2008: 426-434.
- [6] REN Z C, LIANG S S, LI P J, et al. Social collaborative viewpoint regression with explainable recommendations[C]. In Proceedings of WSDM, 2017:485-494.
- [7] YANG B O, L Y, LIU J M, et al. Social collaborative filtering bytrust[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(8): 1633-1647.
- [8] CHEN S, PENG Y. Matrix factorization for recommendation with explicit and implicit feedback[J]. Knowledge Based Systems, 2018: 109-117.
- [9] HE X, ZHANG H, KAN M, et al. Fast matrix factorization for online recommendation with implicit feedback[DB/OL]. <https://arxiv.org/abs/1708.05024>, 2017.
- [10] MASSA P, BHATTACHARJEE B. Using trust in recommender systems: an experimental analysis[C]. International Conference on Trust Management. Springer, Berlin, Heidelberg, 2004: 221-235.
- [11] LIN T H, GAO C, LI Y. Recommender systems with characterized social regularization[C]. Proceedings of the 27th ACM International Conference on Information and Knowledge Management. ACM, 2018: 1767-1770.
- [12] GEMMIS M D, LOPS P, SEMERARO G, et al. An investigation on the serendipity problem in recommender systems[J]. Information Processing & Management, 2015, 51(5): 695-717.
- [13] LIU H F, JING L P, YU J. Survey of matrix factorization based recommendation methods by integrating social information[J]. Journal of Software, 2018, 29(2): 340-362.
- [14] 刘华峰,景丽萍,于剑.融合社交信息的矩阵分解推荐方法研究综述[J].软件学报,2018,29(2):340-362.
- [15] ZHANG Q S, WENG L J. Review of social recommender systems[J]. Computer Engineering and Applications, 2020, 56(1): 1-10.
- [16] 张岐山,翁丽娟.社会化推荐系统综述[J].计算机工程与应用, 2020,57(1):1-10.
- [17] BOTTOU L, CURTIS F E, NOCEDAL J. Optimization methods for large-scale machine learning[J]. SIAM Review, 2018, 60(2): 223-311.
- [18] ROBBINS H, MONRO S. A stochastic approximation method[J]. Annals of Mathematical Statistics, 1951, 22(3): 400-407.
- [19] KASAI H. Stochastic variance reduced multiplicative update for non-negative matrix factorization[C]. Proceedings of the 44th IEEE International Conference on Acoustics, Speech and Signal Processing. Alberta, 2018:6338-6342.

(责任编辑:孙娟)