

# 基于改进矩阵分解算法的推荐方法研究

田海梅

(金陵科技学院计算机工程学院, 江苏 南京 211169)

**摘要:**推荐系统可以主动帮助用户找到符合偏好的个性化物品并推荐给用户,矩阵分解方法是推荐系统中较为经典的方法。为了解决推荐系统矩阵分解中数据稀疏导致矩阵分解性能不佳的问题,提出了对矩阵分解进行改进的方法。首先对矩阵分解进行改进,然后在改进的矩阵分解的基础上融合项目本身潜在的附加信息,最后通过引入项目之间的相似度来进行计算,从而提高推荐算法的准确率。实验结果表明,新方法明显优于现有的矩阵分解推荐模型,具有很好的应用价值。

**关键词:**矩阵分解;不对称因子;隐向量;隐式反馈;推荐系统

中图分类号:TP3-0

文献标识码:A

文章编号:1672-755X(2022)01-0021-05

## Research on Recommendation Method Based on Improved Matrix Factorization Algorithm

TIAN Hai-mei

(Jinling Institute of Technology, Nanjing 211169, China)

**Abstract:** The recommendation system is able to help users actively find personalized items that meet their preferences and recommend them to users. The matrix factorization method is a classic method in the recommendation system. In order to solve the problem of poor performance of matrix factorization due to sparse data in matrix factorization of recommendation systems, a method of improving matrix factorization is proposed. Firstly, the matrix factorization is improved, and then the potential additional information of the item itself is merged on the basis of the improved matrix factorization. Finally, the similarity between items is introduced for calculation, thereby improving the accuracy of the recommendation algorithm. The experimental results show that the improved method is significantly better than the existing matrix factorization recommendation model, and has significant application value.

**Key words:** matrix factorization; asymmetric factors; implicit vectors; implicit feedback; recommendation system

随着互联网飞速发展,信息数量不断增长,而人收集、鉴别信息的能力有限,于是推荐系统的重要性显现出来。推荐系统<sup>[1-2]</sup>分析用户的历史行为和爱好需求,使用合适的推荐算法挖掘用户的潜在兴趣,来为用户提供有价值的推荐。推荐一般有两个目的,一个是激发用户去做某个事情,另一个是解决信息过载问题。按照建模方式的不同,推荐系统分为协同过滤推荐、基于知识的推荐、基于内容的推荐和多种特征混

收稿日期:2021-09-03

基金项目:江苏省现代教育技术研究课题(2018-R-62636)

作者简介:田海梅(1980—),女,江苏海安人,高级实验师,硕士,主要从事推荐系统研究。

合推荐等。目前推荐系统已经运用到很多方面,比如商品平台和社交平台等<sup>[3]</sup>。许多电商通过推荐系统来给用户进行个性化推荐,帮助用户快速发现感兴趣的商品从而给企业带来可观的收入<sup>[4-6]</sup>。

在众多协同过滤推荐算法中,矩阵分解算法具有可扩展性并易于实现,成为最为普遍和最受欢迎的一种协同推荐算法。在推荐系统中,矩阵分解算法可以解决数据的稀疏性问题,比其他协同过滤推荐算法更节省内存,更精确。一般情况下矩阵分解将构建的用户与项目对应的评分矩阵分解为两个低维矩阵乘积,因没有考虑项目与项目之间的非对称相似性问题,最终推荐结果不理想。为了解决这一问题,一些学者已经提出了解决方法。文献[7]提出通过信任感知来进行矩阵分解,从而提高推荐系统的准确性。文献[8]和[9]分别提出利用项目与项目属性之间的相互关系来进行矩阵分解,以及通过一种用户间的非对称的相似性方法来进行矩阵分解,从而达到推荐的效果。以上方法都没有考虑项目本身的潜在知识信息。本文在矩阵分解技术的基础上融入项目本身潜在的知识信息,加上项目与项目之间的不对称相似度,从而提高推荐算法的准确率。

## 1 推荐系统的预备知识

### 1.1 隐向量

对项目与用户之间的评分矩阵进行传统的矩阵分解得到的向量,称为低维向量<sup>[10]</sup>。在推荐系统中,用户(User)对物品(Item)的推荐评分,用物品和用户对应的隐向量的内积来表示<sup>[11-12]</sup>。

### 1.2 传统的矩阵分解算法

矩阵分解算法的主要思想是:首先构造一个由用户ID、项目ID、评分组成的评分矩阵。然后将构建的评分矩阵通过传统的矩阵分解算法转化成两个低维特征矩阵的乘积。最后通过这两个低维特征矩阵去估算那些没有做出评价的缺失项目<sup>[13-14]</sup>。

矩阵分解中用户的隐藏特征矩阵用  $P$  表示,项目的隐藏特征矩阵用  $Q$  表示。矩阵分解首先对构建的原始评分矩阵  $R$  进行分析,然后学习矩阵  $P$  和  $Q$ ,最后缺失项用  $R$  来表示, $R$  由  $P$  和  $Q$  预测。

设  $P \in R^{K \times N}$  和  $Q \in R^{K \times M}$ ,其中向量的维数用  $K$  表示, $K \ll \min(N, M)$ ,然后用分解的低维矩阵  $P$  和  $Q$  的乘积来近似表示  $R$ ,如公式(1)所示:

$$R \approx P^T Q \quad (1)$$

用  $p_u$  表示用户  $u$  的隐藏特征向量, $q_i$  表示项目  $i$  的隐藏特征向量,用  $p_u$  与  $q_i$  内积的形式来预测缺失项  $\hat{r}_{ui}$ ,如公式(2)所示:

$$\hat{r}_{ui} = p_u^T q_i \quad (2)$$

传统的矩阵分解利用损失函数和误差平方最小化来不断学习矩阵  $P$  和  $Q$ ,如公式(3)所示:

$$\min_{p,q} \frac{1}{2} \sum_{(u,i) \in \theta} (r_{ui} - p_u^T q_i)^2 + \frac{\lambda_1}{2} \|P\|_F^2 + \frac{\lambda_2}{2} \|Q\|_F^2 \quad (3)$$

其中, $\theta$  为  $R$  中用户和项目对集合; $\|\cdot\|_F^2$  为  $F$ -范数,正则化  $\|P\|_F^2$  和  $\|Q\|_F^2$  用来避免过拟合; $\lambda_1$  和  $\lambda_2$  是为了防止过拟合而影响隐藏特征向量的正则化参数。在正则化中,要阻止  $P$  和  $Q$  中出现非常大的系数值,以保证稳定性。

通过传统的矩阵分解算法得到用户  $u$  与项目  $i$  的隐向量,虽然降低了隐向量的维度,但是并没有该用户及其评价过的项目特征的许多附件信息,用户  $u$  的隐向量跟真实情况之间有一些差别<sup>[15-18]</sup>。

### 1.3 加权非对称(asymmetric)的相似性度量方法

定义1 给定关联开放数据(linked open data, LOD)中两个资源  $r, s \in R$ ,  $F_r$  和  $F_s$  分别表示它们的特征集合,  $F_r \cap F_s$  是  $r$  和  $s$  的共同特征, $r$  和  $s$  共同的分区信息内容(partitioned information content, PIC)值占  $r$  的 PIC 值的比例由公式(4)定义,占  $s$  的 PIC 值的比例由公式(5)定义:

$$CPIC(r, s) = \frac{PIC(F_r \cap F_s)}{PIC(F_r)} \quad (4)$$

$$CPIC(s, r) = \frac{PIC(F_s \cap F_r)}{PIC(F_s)} \quad (5)$$

CPIC 的取值范围为 $[0, 1]$ , CPIC 是一种非对称的方式, 即  $CPIC(r, s) \neq CPIC(s, r)$ 。

$$EPIC(r, s) = 1 - \exp(-CPIC(r, s)) \tag{6}$$

$$EPIC(s, r) = 1 - \exp(-CPIC(s, r)) \tag{7}$$

EPIC 的取值范围为 $[0, 1]$ , EPIC 是一种非对称的方式, 即  $EPIC(r, s) \neq EPIC(s, r)$ 。

定义 2 给定 LOD 中两个资源  $r, s \in \mathbf{R}$ ,  $F_r$  和  $F_s$  分别表示它们的特征集合,  $F_r - F_s$  表示在  $r$  中而不在  $s$  中的特征,  $F_s - F_r$  表示在  $s$  中而不在  $r$  中的特征。两个资源之间差异化的 DPIC 由公式(8)和公式(9)定义:

$$DPIC(r, s) = \frac{1}{PIC(F_r - F_s) + PIC(F_s - F_r) + 1} \tag{8}$$

$$DPIC(r, s) = DPIC(s, r) \tag{9}$$

DPIC 的取值范围为 $(0, 1]$ , DPIC 是一种对称的度量方式, 分母加 1 是防止分母为 0。本文采用以上两个概念 EPIC 和 DPIC, 提出了一种基于 PIC 的非对称的语义相似度计算方法 AEPICSS, 用来计算 LOD 中资源之间的相似度, AEPICSS 由公式(10)和(11)定义:

$$AEPICSS(r, s) = EPIC(r, s) \cdot DPIC(r, s) \tag{10}$$

$$AEPICSS(s, r) = EPIC(s, r) \cdot DPIC(s, r) \tag{11}$$

AEPICSS 的取值范围为 $[0, 1]$ , 因为 CPIC 是非对称的, AEPICSS 显然是一种非对称的基于 LOD 的项目相似度计算方法, 即  $AEPICSS(r, s) \neq AEPICSS(s, r)$ 。

## 2 改进的矩阵分解算法在推荐系统中的应用

### 2.1 改进的矩阵分解算法和不对称因子模型

早期的基于矩阵分解的推荐算法是对用户与项目之间对应的评分矩阵进行分解, 没有充分考虑项目属性附加信息, 对项目与项目之间的不对称相似性问题缺乏考虑, 导致推荐的准确率一般。

本文改进的矩阵分解算法的主要思想是: 在早期矩阵分解的基础上融入项目本身潜在的知识信息, 加上项目与项目之间的不对称相似度, 从而提高推荐算法的准确率。对早期的矩阵分解方法进行改进后的流程如图 1 所示。

融合后的改进算法目标函数如公式(12)所示。

$$e_{ij}^2 = \delta \left( r_{ij} - \sum_{f=1}^k p_{if} \times q_{fj} \right)^2 + \frac{\beta}{2} \sum_{f=1}^k (p_{if}^2 + q_{fj}^2) + \epsilon \sum_{t=1}^M (\mu \times e^{1-\frac{1}{sim_{jt}}} ) \left[ \left( r_{it} - \sum_{f=1}^k p_{if} \times q_{fj} \right) - 5 \left( 1 - e^{1-\frac{1}{sim_{jt}}} \right) \right]^2 \tag{12}$$

设有  $M$  个项目, 其中  $r_{it}$  是第  $i$  个用户对第  $t$  个项目的评分。当  $r_{it} > 0$  时,  $\mu = 1, \delta = 1, \epsilon = 0$ ; 当  $r_{it} = 0$  时,  $\mu = 0, \delta = 0, \epsilon = 1$ 。  $sim_{jt}$  是项目  $t$  和  $j$  的相似度。其中  $sim$  用加权非对称(asymmetric)相似度来度量。

运用随机梯度下降的方法求解  $e_{ij}^2$  的局部最小解。对目标函数  $e_{ij}^2$  关于  $p_{ij}$  来求偏导数如公式(13)所示:

$$\frac{\partial e_{ij}^2}{\partial p_{ij}} = \beta p_{ij} - 2\delta q_{fj} \left( r_{ij} - \sum_{f=1}^k p_{if} \times q_{fj} \right) - 2q_{fj} \epsilon \sum_{t=1}^M (\mu \times e^{1-\frac{1}{sim_{jt}}} ) \left[ \left( r_{it} - \sum_{f=1}^k (p_{if} \times q_{fj}) \right) - 5 \left( 1 - e^{1-\frac{1}{sim_{jt}}} \right) \right] \tag{13}$$

### 2.2 推荐系统的评价

在推荐系统中, 一般采用均方差(mean square error, MSE)、均方根误差(root mean square error, RMSE)和平均绝对值误差(mean absolute error, MAE)来对比两个数值之间的不同。MSE 是指估计值与真实值之间的差异值进行平方的期望值。RMSE 是指预测模型对实验数据的精确度的描述, 是均方差的算术平方根, RMSE 越高表示精确度越低, RMSE 越低表示精确度越高。实际的测值误差本文用 MAE 来表示。

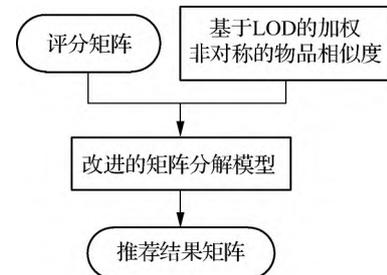


图 1 改进的矩阵分解模型流程

$$\text{MAE} = \frac{\sum_{u,i \in T} |r_{ui} - \hat{r}_{ui}|}{|T|} \quad (14)$$

其中  $r_{ui}$  为用户  $u$  对物品  $i$  的评分,  $\hat{r}_{ui}$  为用户  $u$  对物品  $i$  的预测评分,  $|T|$  是测试评分数。

$$\text{MSE} = \frac{\sum_{u,i \in T} (r_{ui} - \hat{r}_{ui})^2}{|T|} \quad (15)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{u,i \in T} (r_{ui} - \hat{r}_{ui})^2}{|T|}} \quad (16)$$

### 3 实验结果与分析

本文采用 MovieLens 100k、MovieLens 1M 和 Netflix 数据集。MovieLens 是用户对电影评分的数据集,包括用户 ID、电影 ID 和用户对电影的对应评分,MovieLens 100k 与 MovieLens 1M 由 GroupLens 公司提供。Netflix 数据集由 Netflix 公司提供,由于该数据集十分庞大,本文从中随机选取 990 个用户在 1 499 个物品上的 49 933 条评分记录,且每个用户至少对 20 部电影进行评分。这 3 个数据集都被广泛用于推荐系统研究,相关信息如表 1 所示。

表 1 数据集统计结果

数据集	用户数	物品数	评分数	每个用户平均评分数	每个物品平均评分数
MovieLens 100k	943	1 682	100 000	106.045	59.453
MovieLens 1M	6 040	3 900	1 000 209	165.598	256.464
Netflix	990	1 499	49 933	50.437	33.311

本文的训练数据为随机从构建好的用户项目评分矩阵中抽取 70%,测试数据为评分矩阵中剩余的 30%。评价指标用 MSE、RMSE 和 MAE 来表示。本文选取传统的矩阵分解(basic MF)、非负矩阵分解(non-negative MF)和 SVD++ 模型进行了对比实验,采用多次实验的平均值作为结果。参数设定过程中,为了与基准算法进行公平的比较,视迭代次数达到 100 或者训练误差小于  $10^{-6}$  时本文模型的学习过程达到收敛。

学习速率  $\alpha$  和正则项参数  $\beta$  通常设置为很小的值。用 3 个数据集上的 RMSE 来选择这两个参数,对于 MovieLens 100k,  $\alpha$  取 0.003 时 RMSE 最小,  $\beta$  取 0.007 时 RMSE 最小;对于 MovieLens 1M,  $\alpha$  取 0.001 时 RMSE 最小,  $\beta$  取 0.008 时 RMSE 最小;对于 Netflix,  $\alpha$  取 0.003 时 RMSE 最小,  $\beta$  取 0.007 时 RMSE 最小。特征数通常代表物品的分类数,在实验中,当特征数小于 70 时 RMSE 变化很小。在下面的实验中,选择 20 作为特征数。

参数设置为最优值、特征个数设置为 20 时的结果如表 2 所示。从表 2 可以看出,本文的改进方法优于所有的对比方法,即 MSE、RMSE 和 MAE 均低于所有的对比方法,其中在 Netflix 数据集上的 MSE 比其他方法 MSE 最多提升了 14.31%。本文方法的准确率,在 MovieLens 100k 数据集上较 SVD++ 提高了 0.9%,在 MovieLens 1M 数据集上较 SVD++ 提高了 6.24%,在 Netflix 数据集上较 SVD++ 提高了 7.81%。

表 2 各方法在不同数据集上的结果

方法	MovieLens 100k 数据集			MovieLens 1M 数据集			Netflix 数据集		
	MSE	RMSE	MAE	MSE	RMSE	MAE	MSE	RMSE	MAE
basic MF	1.075	1.036	0.799	0.802	0.895	0.722	1.067	1.033	0.799
non-negative MF	1.036	1.028	0.790	0.782	0.884	0.701	0.996	0.998	0.782
SVD++	0.998	0.993	0.785	0.761	0.872	0.698	0.982	0.991	0.773
本文方法	0.969	0.984	0.778	0.735	0.857	0.657	0.859	0.927	0.717
	(3.00%)	(0.91%)	(0.90%)	(3.50%)	(1.75%)	(6.24%)	(14.31%)	(6.90%)	(7.81%)

注:括号内数值表示本文方法比 SVD++ 方法提高的百分比。

## 4 结 语

本文结合改进的矩阵分解模型和不对称因子模型的优点,在矩阵分解技术的基础上融合项目本身潜在的附加信息,并考虑项目之间的不对称相似度,从而大大提高推荐算法的准确率。同时数据的稀疏性也有所降低。改进的矩阵分解算法对整个推荐系统中的一些应用起到了很好的优化作用。从 MovieLens 100k、MovieLens 1M 和 Netflix 数据集上的实验数据来看,本文方法的 MSE、RMSE、MAE 值均小于所有的对比方法。实验结果表明,本文方法有效地提高了推荐的性能,可以在推荐系统中得到更好的推荐效果,这说明本文采用的不对称因子模型已经很好地整合到改进的矩阵分解中。本文的方法提升了推荐的准确性,网络线上商城、美食团购推荐平台可根据用户的历史评分和项目的属性进行推荐,给用户带来更好的使用感受;但是在空白数据填充和冷启动方面仍存在一定的问題,这将是下一步的研究方向。

### 参考文献:

- [1] KLAŠNJA-MILIĆEVIĆ A, IVANOVIĆ M, NANOPOULOS A. Recommender systems in e-learning environments: a survey of the state-of-the-art and possible extensions[J]. Artificial Intelligence Review, 2015, 44(4): 571–604
- [2] KHAN M M, IBRAHIM R, GHANI I. Cross domain recommender systems[J]. ACM Computing Surveys, 2017, 50(3): 1–34
- [3] 黄立威, 江碧涛, 吕守业, 等. 基于深度学习的推荐系统研究综述[J]. 计算机学报, 2018, 41(7): 1619–1647
- [4] CHENG Z Y, DING Y, ZHU L, et al. Aspect-aware latent factor model: rating prediction with ratings and reviews[C]. Lyon: Proceedings of the 2018 World Wide Web Conference, 2018: 639–648
- [5] 赵桂升, 潘善亮. 基于 IRGAN 模型和 Hadoop 的电影推荐系统的设计[J]. 计算机应用与软件, 2019, 36(5): 43–50
- [6] 蒋云, 倪静, 房宏扬. 融合社交行为和标签行为的推荐算法研究[J]. 计算机应用研究, 2019, 36(7): 1965–1969
- [7] FORSATI R, MAHDAVI M, SHAMSFARD M, et al. Matrix factorization with explicit trust and distrust side information for improved social recommendation[J]. ACM Transactions on Information Systems, 2014, 32(4): 1–38
- [8] LI F F, XU G D, CAO L B. Coupled item-based matrix factorization[M]. Cham: Springer International Publishing, 2014: 1–14
- [9] PIRASTEH P, HWANG D, JUNG J J. Exploiting matrix factorization to asymmetric user similarities in recommendation systems[J]. Knowledge-based Systems, 2015, 83: 51–57
- [10] 韦峰. 推荐系统中矩阵分解算法研究[D]. 合肥: 中国科学技术大学, 2017
- [11] 高玉凯, 王新华, 郭磊, 等. 一种基于协同矩阵分解的用户冷启动推荐算法[J]. 计算机研究与发展, 2017, 54(8): 1813–1823
- [12] ZHENG D X, XIONG Y H. A unified probabilistic matrix factorization recommendation algorithm[C]. Changsha: 2018 International Conference on Robots & Intelligent System(ICRIS), 2018: 246–249
- [13] ZHANG W, WANG J, FENG W. Combining latent factor model with location features for event-based group recommendation[C]//Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Chicago: ACM, 2013: 910–918
- [14] AGARWAL D, CHEN B C. Regression-based latent factor models[C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Paris: ACM, 2009: 19–28
- [15] KOREN Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model[C]//Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Las Vegas: ACM, 2008: 426–434
- [16] 田震, 潘腊梅, 尹朴, 等. 深度矩阵分解推荐算法[J]. 软件学报, 2021, 32(12): 3917–3928
- [17] 徐俊, 张政, 杜宣萱, 等. 基于项目语义的协同过滤冷启动推荐算法研究[J]. 小型微型计算机系统, 2021, 42(11): 2246–2251
- [18] 董志恒, 牟胜东. 基于用户评分权重的矩阵分解推荐算法[J]. 吉林工程技术师范学院学报, 2021, 37(5): 92–94

(责任编辑: 湛 江)