

DOI:10.13232/j.cnki.jnju.2019.06.008

## 基于张量分解和深度学习的混合推荐算法

张家精<sup>1</sup>, 夏巽鹏<sup>2\*</sup>, 陈金兰<sup>3</sup>, 倪友聪<sup>4</sup>

(1. 安徽建筑大学数理学院, 合肥, 230601; 2. 安徽建筑大学电子与信息学院, 合肥, 230601;

3. 安徽建筑大学机械与电气工程学院, 合肥, 230601; 4. 福建师范大学数学与信息学院, 福州, 350007)

**摘要:** 张量分解和深度学习已被应用于推荐系统, 并取得了较好的效果. 张量分解较好地为用户对推荐对象评分中提取用户、推荐对象以及其他影响因素的隐性的特征, 将这些特征进行匹配, 给出推荐策略, 但这种方法忽略了用户、推荐对象以及其他影响因素现有辅助数据信息中的显性特征. 深度学习是从辅助信息中提取用户、推荐对象以及其他影响因素的特征, 并进行匹配给出推荐策略, 却忽略了用户评分数据中用户、推荐对象以及其他影响因素的隐性特征. 将张量分解和深度学习两种推荐方法相融合, 提出一种基于张量分解和深度学习的混合推荐算法. 使用张量分解算法和深度学习分别从三阶用户评分数据和多源异构辅助信息中提取用户特征和推荐对象特征, 并将它们匹配得出用户对推荐对象的需求或喜爱的预测评分, 再将两种算法的预测评分进行融合给出最终综合评分, 从而提高个性化推荐的精准度. 对比实验证明混合推荐算法与传统的协同过滤算法相比误差降低了 34.0%.

**关键词:** 混合推荐算法, 张量分解, 深度学习, 辅助数据, 评分数据

**中图分类号:** TP393

**文献标识码:** A

## Blending recommendation algorithm based on tensor decompositions and deep learning

Zhang Jiajing<sup>1</sup>, Xia Xunpeng<sup>2\*</sup>, Chen Jinlan<sup>3</sup>, Ni Youcong<sup>4</sup>

(1. School of Mathematics & Physics, Anhui Jianzhu University, Hefei, 230601, China;

2. School of Electronics and Information, Anhui Jianzhu University, Hefei, 230601, China;

3. School of Mechanical and Electrical Engineering, Anhui Jianzhu University, Hefei, 230601, China;

4. School of Math and Information Science, Fujian Normal University, Fuzhou, 350007, China)

**Abstract:** The tensor decomposition and deep learning have been applied to the recommendation systems and better results have become true. The tensor decomposition algorithm better extracts the hidden features of the users, recommended objects and other influencing factors from the user rating data, and the features are matched each other to give recommendation strategies. But the algorithm ignores the features in the auxiliary data information of the user, recommended objects and other influencing factors. Deep learning extracts the features of users, recommended objects and other influencing factors from auxiliary information, and matches them to give recommendation strategies, but ignores the implicit characteristics of users, recommended objects and other influencing factors in user rating data. A

基金项目: 安徽省自然科学基金(1708085MA19), 安徽省级教学研究重点项目(2016jyxm0207), 安徽省高校优秀青年支持计划(gxyq2017024)

收稿日期: 2019-07-17

\* 通讯联系人, E-mail: xxp1074436434@outlook.sg

blending recommendation algorithm based on tensor decomposition and deep learning is proposed, which blends the two recommendation methods of tensor decomposition and deep learning. Tensor decomposition algorithm and deep learning are used to extract user features and recommendation object features from third-order user rating data and multi-source heterogeneous auxiliary information respectively, and then match them to obtain prediction ratings of user's demand or preference for recommendation objects, and the prediction ratings from the two algorithms are blended to give the final comprehensive ratings. The blending recommendation algorithm will improve the accuracy of personalized recommendation. Compared with traditional collaborative filtering algorithm, the error of blending recommendation algorithm is reduced by 34.0%.

**Key words:** blending recommendation algorithm, tensor decomposition, deep learning, auxiliary data, rating data

互联网的普及和应用给广大用户带来大量又繁杂的信息,在满足用户信息需求的同时,也使用户难以获得对自己真正有用的信息,对信息的利用率反而降低,这就是现今日益严重的“信息超载”问题<sup>[1]</sup>. 解决信息超载问题的一个非常有效的办法是使用个性化推荐系统.

传统的推荐算法大多研究用户和推荐对象两者之间的关系,例如矩阵分解通过现有部分用户对推荐对象的评分提取用户和推荐对象隐含在评分数据中的隐性特征,再根据对应特征的匹配程度进行推荐<sup>[2-3]</sup>. 然而给用户推荐的对象往往不仅仅取决于用户和推荐对象两个方面的特征,还受其他因素的影响,如销售商的服务态度、标签及物流情况等. 因此现有的推荐系统扩展了推荐的影响因素,考虑三个甚至三个以上因素特征匹配的推荐,其中张量分解就利用张量的高阶特征,从评分中提取三个或者三个以上影响因素的特征,再将三个或三个以上影响因素对应特征进行匹配,得出推荐结果,从而提高推荐的精准度<sup>[4-5]</sup>.

然而张量分解是从评分数据中提取用户、推荐对象以及其他影响因素的特征,需要对评分数据有较深入的了解,依靠人工确定要提取哪些因素的特征<sup>[6]</sup>,这样可能忽略其他没有发现的因素,同时没有考虑现有辅助信息中的用户、推荐对象和标签等直接给出的显性特征. 而协同过滤、深度学习等方法可以从用户信息、推荐对象信息、社会化关系以及其他影响因素的多源异构化辅助信息中提取用户、推荐对象以及其他影响因素等的显性特征,匹配相应特

征并给出推荐策略. 如 Guo et al<sup>[7]</sup>采用推荐对象协同过滤方法,根据所有用户的历史偏好数据计算物品之间的相似性,再把与用户喜欢的类似的物品推荐给用户. Pan et al<sup>[8]</sup>通过深度学习网络收集网络上的社会标签数据,并添加一种基于标签权重作为大众分类法的依据,利用该分类准则来为推荐物品做排序分类. Liao et al<sup>[9]</sup>从亿级的用户的使用记录中提取特征信息,从而提供相对个性化的推荐服务. 但这些基于协同过滤或深度学习的推荐系统只是对用户、推荐对象及使用记录等辅助数据(非评分数据)进行分析,在数据稀疏的情景下表现不佳,且未考虑评分数据信息. 邓腾飞<sup>[10]</sup>引入深度神经网络来预测歌曲的切歌率并利用海量用户的听歌记录数据训练出一种切歌频率预测模型. 这种方法虽然用到切歌的“评分”数据,但是过于简单,无法提取用户评分数据内部隐含的用户、推荐对象和其他影响因素的隐性特征以及这些特征之间的匹配关系.

任何单一推荐算法都存在相应的优缺点<sup>[11]</sup>,通过融合不同的推荐算法,兼顾提取研究对象不同方面的特征,融合推荐结果,却往往能表现出更优异的性能. 本文为解决张量分解只考虑评分数据、未考虑辅助信息,而深度学习只学习辅助信息、未深入考虑评分数据的问题,提出基于张量分解和深度学习的混合推荐算法(Blending Recommendation Algorithm base of Tensor Decompositions and Deep Learning, BRATDDL). 该算法先用张量分解从用户的评分数据中提取用户、推荐对象以及其他影响

因素的隐性特征及它们之间的相互关系,经相应特征匹配得出用户对推荐对象的需求喜爱的预测评分;再使用深度学习方法从多源异构的辅助信息中提取用户、推荐对象以及其他影响因素的特征,使相应特征匹配得出预测评分;最后,把张量分解方法得到的预测评分和深度学习方法得到的预测评分融合形成最终综合评分。同时在融合过程中,为处理文本深度学习优化改进深度学习算法。该算法融合张量分解和深度学习方法,兼顾用户、推荐对象以及其他影响因素在用户评分中的隐性特征和辅助信息中的特征,提高预测推荐的精准度。为验证文中提出的新的混合推荐算法,使用 Movielens-1M 和 Nexflix-3M 两个电影数据集训练推荐模型,并传统推荐算法进行性能对比,验证该混合推荐算法在社会标签推荐系统中的性能。

本文的主要贡献:(1)使用张量分解方法从评分数据中提取用户、待推荐对象以及其他影响因素的相互作用关系,再利用深度学习方法从辅助信息中提取用户、待推荐对象以及其他影响因素的特征,并将两者预测评分结果进行融合,得到精度更高的个性推荐。(2)对原有的深度学习网络进行改进,用文本卷积和词向量模型对复杂的辅助信息做预处理,使之能更有效地整合数据集中的辅助信息。(3)使用 Movie-lens-1M 和 Netflix-3M 两种数据集进行训练,对本文提出的推荐算法做验证性实验,并通过多组评价指标对推荐算法性能做量化的对比。

## 1 基于张量分解和深度学习的混合推荐框架

通常的融合形式大致分为三种:前融合、中融合、后融合。其中,后融合是指将两种或两种以上的推荐算法产生的推荐结果,以投票机制、线性组合或者可信度选择组合等方式产生最终的推荐结果。本文即采用后融合方法,分别用张量分解和深度学习获得预测评分,再将这两种预测评分用不同的混合机制进行融合,得出最终预测评分。具体流程如图 1 所示。

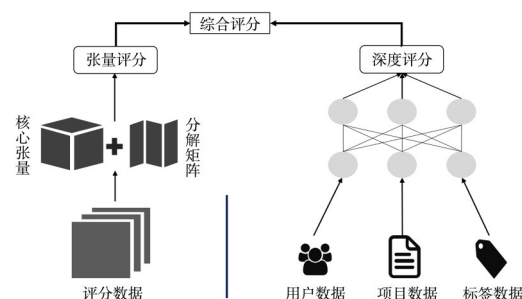


图 1 张量分解和深度学习混合推荐算法示意图

Fig. 1 Schematic diagram of blending recommendation algorithm based on tensor decomposition and deep learning

BRATDDL 算法由张量分解、深度学习及融合三部分组合而成。张量分解和深度学习分别对应两个不同输入,张量分解部分的输入为用户对推荐对象(简称项目)和标签的评分数据,深度学习部分的输入为用户、项目和标签本身对应的辅助数据信息,如用户信息、项目信息等。张量分解部分从评分数据张量中分解提取用户、项目和标签的隐性特征,并进行相应的特征匹配,得出用户对不同标签下不同项目的预测评分。深度学习部分直接从用户、项目和标签等辅助信息中提取相应的特征,再进行相应特征匹配,计算得出用户对不同标签下不同项目的预测评分。融合部分将张量分解和深度学习部分的预测评分通过加权和回归机制融合计算,得出最终综合预测评分。

## 2 张量分解

张量分解部分属于因式分解机,目标是从高阶的用户评价张量中提取用户、项目及标签的隐性特征。用户的评分数据既包含用户对某项目的喜好特征,也包含项目自身特征以及其所属的标签特征。张量分解可直接从用户评分的高阶张量数据中提取用户、项目和标签的特征,再将用户、项目和标签特征匹配获得预测评分。理论上,张量分解可以学习更高阶的特征之间相互作用关系,但由于受到数据集的限制,本文只处理三阶的特征关系,即用户、项目和标签三者的内在联系。

**2.1 张量分解算法** 使用  $Y_{ijk}$  表示一个三阶评分张量,  $X_{ijk}$  作为三阶评分张量的近似张量.

$$Y_{ijk} \approx X_{ijk}$$

本文所使用的张量分解模型为 Tucker 分解模型<sup>[12]</sup>, 此模型是一种高阶主成分分析方法, 将一个三阶评分张量分解成一个三维核心张量与三个二维矩阵乘积的形式. 分解模型如式(1)所示:

$$Y_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R U_{ip} T_{jq} I_{kr} S_{ijk} \quad (1)$$

式(1)中,  $U_{ip}$ ,  $T_{jq}$ ,  $I_{kr}$  是三个分解矩阵(通常为正交矩阵), 一般被定义为各个维度上的主成分,  $P, Q, R$  为各个矩阵对应的阶数.  $S_{ijk}$  是核心张量, 表达了各个维度上的潜在相互关系.

**2.2 计算近似张量** 近似张量的计算为张量分解的逆向运算, 通过将目标张量  $Y_{ijk}$  分解出的三个矩阵和一个核心张量进行重新组合, 还原出近似张量  $X_{ijk}$ , 如式(2)所示:

$$X_{ijk} \approx \sum_{pqr} S_{ijk} \times U_{ip} \times T_{jq} \times I_{kr} \quad (2)$$

经过重组后, 新产生的近似张量将给各个用户对应的项目和标签重新给出近似的评分, 同时预测出原来评分张量中缺省的评分, 这不仅可以还原用户的真实评价, 还可以预测出某用户对某种没有评价过的项目的评分, 并根据比较, 发现用户可能评价较高的项目, 从而把这些项目优先推荐给用户; 同时, 与评分张量  $Y_{ijk}$  进行误差比对, 通过梯度下降法更新迭代, 最终获得相似度最高的近似张量, 给出张量分解部分的预测评分  $y_{TD}$ .

**2.3 误差分析** 张量分解的误差是通过评分张量和近似张量之间的差异表示, 且为防止过拟合现象增加正则项, 其损失函数用式(3)表示:

$$Loss = \min_{e^u, e^i, e^t} \sum_{ijk} (Y_{ijk} - X_{ijk})^2 + \lambda^u \|e^u\|_F^2 + \lambda^i \|e^i\|_F^2 + \lambda^t \|e^t\|_F^2 \quad (3)$$

其中,  $e^u$ ,  $e^i$  和  $e^t$  分别对应三个矩阵.  $\lambda^u$ ,  $\lambda^i$  和  $\lambda^t$  为正则系数.

### 3 深度学习

深度学习部分是基于BP神经网络构建的, 如图2所示. 首先, 将这些稀疏、高维的原始数据通过嵌入层转换成低维、密集的实值向量; 然后将低维密集的嵌入层向量中的值传入隐含层, 每一个隐含层根据式(4)来计算更新:

$$a^{(l+1)} = f(W^{(l)}a^{(l)} + b^{(l)}) \quad (4)$$

其中,  $f$  为激活函数,  $l$  为所在隐含层层数.  $W^{(l)}$ ,  $a^{(l)}$  和  $b^{(l)}$  分别为模型对应层级的权重值、输入值和偏移量.

如图2所示, 在实验训练中使用电影数据集作为训练样本, 用户、项目和标签分别对应数据集中的用户、电影名和电影标签等信息. 深度学习部分处理用户、电影名和电影标签等信息. 其中, 用户信息和电影标签为结构化文本数据, 采用循环神经网络提取特征; 电影名信息为文本类信息, 所以采用文本卷积网络提取特征. 再将循环提取的电影标签特征和文本卷积网络提取的电影名特征统一输入次级全连接层得出电影特征, 最后把用户特征和电影特征进行联合训练, 给出深度学习部分的预测评分.

**3.1 数据的预处理部分** 深度学习的数据预处理部分如图2预处理部分所示, 本文直接使用嵌入层对用户信息和电影类型这类结构化信息进行预处理. 而电影名特征为文本类信息, 较为复杂, 需要使用文本卷积网络<sup>[13]</sup>进行处理, 再将提取的电影名特征与电影类型特征共同输入全连接层, 产生电影特征.

卷积神经网络模型构建如图3所示. 网络的第一层为嵌入层, 分别由每一个处理过的单词向量组成的嵌入矩阵. 第二层为卷积层, 使用特征提取器在嵌入矩阵上做卷积操作, 将词嵌入层中待输入的信息进行特征提取. 第三层为池化层, 提取前一层中各特征提取器产生的特征值实现代表性的特征提取. 最后一层为全连接层, 输出电影名称特征.

**3.2 循环神经网络** 如图2神经网络部分所示, 在嵌入层后使用循环神经网络对预处理得



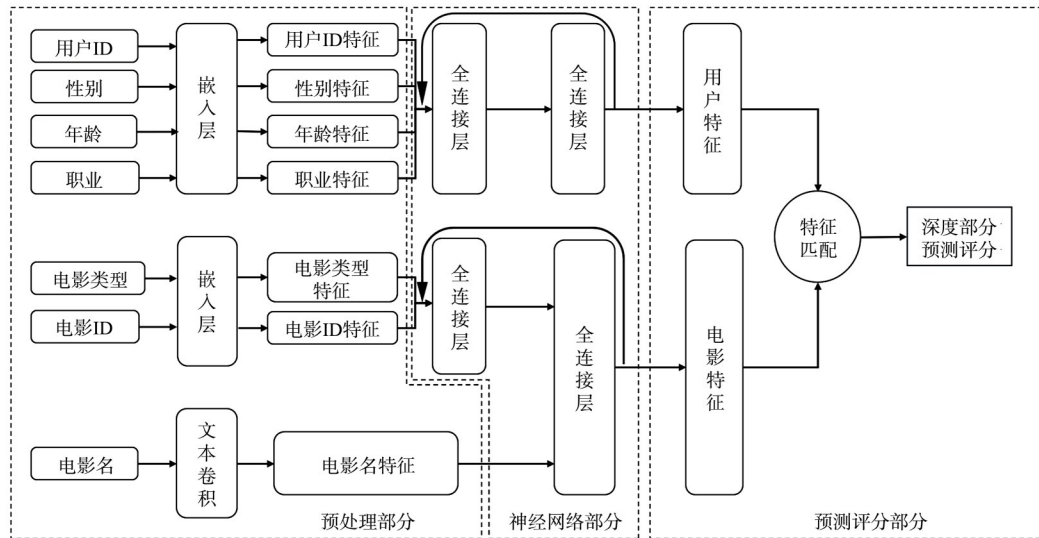


图 2 深度学习部分流程图

Fig. 2 Flow chart of deep learning part

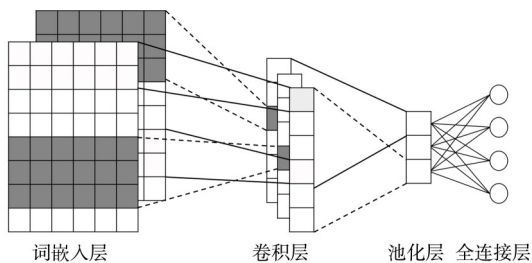


图 3 文本卷积部分具体流程示意图

Fig. 3 Specific flow chart of text convolution part

出的信息进行特征处理,神经网络的神经元连接方式为全连接,嵌入层中的预处理的数据输入全连接层中开始网络训练,得出的训练特征回传至第一级全连接层进行循环训练,得到用户特征  $h_i^n$ . 在电影 ID、电影类型和电影名中,电影 ID、电影类型和电影名特征通过两层全连接层进行训练,再将该全连接层的输出值回传至第一级全连接层进行循环训练,提取得到电影特征  $h_j^m$ .

**3.3 用户特征和电影特征的匹配** 用户特征和电影特征匹配如图 2 预测评分部分所示,通过联合训练法将神经网络给出的用户特征和电影特征进行匹配,把上节得出的用户部分特征  $h_i^n$  和电影部分特征  $h_j^m$  作为输入,传入全连接层,得到输出值,再将输出值进行逻辑斯蒂回归

得出预测评分概率值,采用 Adam 算法<sup>[14]</sup>作为深度学习部分的优化策略. 深度学习部分的模型预测评分计算公式为式(5)和式(6):

$$P(y|x) = \text{Softmax}(w_{\text{User}}^T h_i^n + w_{\text{Movie}}^T h_j^m + b) \quad (5)$$

$$y_{\text{Deep}} = \text{argmax} P(y|x) \quad (6)$$

式(5)中,  $y$  为评分等级,分别对应 1~5 个评分等级,  $P(y|x)$  为各个预测评分等级的概率值,  $x$  为待推荐项目,  $w_{\text{User}}^T h_i^n$  和  $w_{\text{Movie}}^T h_j^m$  分别代表用户模型和电影模型所对应的权重值和输入值的乘积,  $w_{\text{User}}^T$  和  $w_{\text{Movie}}^T$  的权重值依据经验给出,  $b$  为偏移量. 最后,式(6)中取  $P(y|x)$  概率最高值所对应的评分等级,作为深度学习部分的预测评分,记为  $y_{\text{Deep}}$ .

## 4 张量分解和深度学习结果融合

本文分别使用两种混合方法将张量分解和深度学习结果进行融合.

**线性回归法:**将两部分的评分相加,使用线性回归模型得出最终预测评分,预测模型如式(7):

$$y_{\text{score}} = w_1 y_{\text{TD}} + w_2 y_{\text{Deep}} + b \quad (7)$$

其中,  $y_{\text{score}} \in [0, 5]$  为线性回归法给出的最终综

合评分,  $y_{TD}$  为张量分解得到的预测评分,  $y_{Deep}$  为深度学习得到的预测评分;  $w_1$  和  $w_2$  分别为两部分权重, 用来调节张量分解和深度学习预测评分的比重, 由人工或者机器学习方式给出;  $b$  为偏移量, 也可人工或机器学习方法给出。

逻辑斯蒂回归法: 如图4所示, 本方法不同于线性回归, 先将两部分评分相加, 然后通过逻辑斯蒂回归函数转化成概率, 再取概率最大时对应的分值, 给出最终预测评分。模型如式(8)、式(9)和式(10)所示:

$$P(y|x) = g(y_{TD}, y_{Deep}) = \frac{1}{1 + \exp(-s_2)} \quad (8)$$

$$s_2 = y_{TD} + y_{Deep} \quad (9)$$

$$y_{score} = \operatorname{argmax} P(y|x) \quad (10)$$

综合评分分为五个评分等级, 选取发生概率最大的评分为最终综合评分, 记为  $y_{score}$ 。

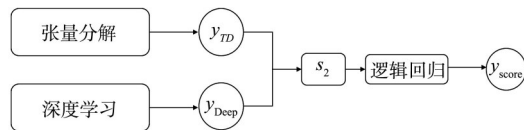


图4 逻辑斯蒂回归示意图

Fig. 4 Logistic regression diagram

## 5 实验结果

实验环境为 Anaconda 的 Python3, 以 TensorFlow 为底层网络框架创建整个算法数据流图, 并搭建深度学习网络, 分别在 Movielens-1M 和 Netflix-3M 数据集上进行试验。先从用户、电影名和电影标签数据集中提取特征, 再使用 Tensorly 工具包实现数据集的 Tucker 分解。

将 Movielens-1M 按一定比例随机划分, 再按照交叉验证原则完成评估验证。共通过 16000 次迭代训练、测试, 分五次 Epoch 进行, 每个 Epoch 为 3109, 777 次 batch, 每 20 次迭代输出一次结果, Loss 损失函数计算结果如图 5。

**5.1 张量分解和深度学习混合机制效果对比** 为了定量比较线性回归和逻辑斯蒂回归两种混合机制, 将 BRATDDL 中两种混合机制分

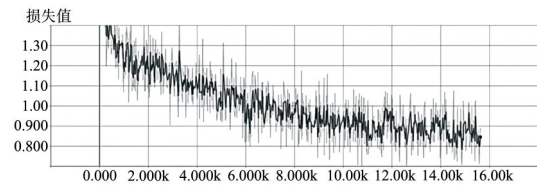


图5 模型训练损失值情况

Fig. 5 The loss value of model training

别在 Movielens-1M (推荐结果见表1) 和 Netflix-3M (推荐结果见表2) 数据集上进行试验, 并对推荐结果的 Precision, NDCG (Normalized Discounted Cumulative Gain), MRR (Mean Reciprocal Rank) 以及 MAP (Mean Average Precision) 进行对比。其中 Precision 表示精准度<sup>[14]</sup>; NDCG<sup>[15]</sup> 表示排序后各个相关等级的精准度; MRR<sup>[16]</sup> 表示对推荐结果排序取倒数, 再对所有等级的推荐结果求平均; MAP<sup>[17]</sup> 表示平均精准度。

表1 Movielens-1M 数据集上的电影推荐结果

Table 1 Recommendation of movies on Movielens-1M dataset

	Precision	NDCG	MRR	MAP
线性回归	0.2624	0.2735	0.4463	0.3902
逻辑斯蒂回归	0.3086	0.2978	0.4645	0.4132

表2 Netflix-3M 数据集上的电影推荐结果

Table 2 Recommendation of movies on Netflix-3M dataset

	Precision	NDCG	MRR	MAP
线性回归	0.2832	0.2735	0.4257	0.3904
逻辑斯蒂回归	0.2905	0.2879	0.4403	0.4103

从表1可见, 在 Movielens-1M 数据集上, 逻辑斯蒂回归机制比线性回归算法的 Precision 提升 17.60%, NDCG 提升 8.08%。从表2可见, 在 Netflix-3M 数据集上, 逻辑斯蒂回归机制比线性回归算法的 MRR 提升 1.03%, MAP 提升 1.09%。本文使用的评分数据集为用户真实评分, 共有 1~5 五个评分等级, 且评分均为整数分值。线性回归混合机制对两部分评分做直接融合, 得出的评分大多数为介于各个评分等级

之间的分数区域,最终必须取整,给出有截断误差的评分等级,因此给出的评分不可避免会出现与原始评分数据不相符的误差。

## 5.2 BRATDDL 与独立模型误差性能的对比

研究模型在电影推荐的表现,将深度学习、张量分解和 BRATDDL 算法分别在极度稀疏的 Movielens-1M 数据集上进行实验,提取训练过程中产生的误差(图 6)。和深度学习和张量分解模型相比,BRATDDL 模型误差最小。深度学习模型在大量数据训练后趋于稳定,误差比张量分解算法有较大降低。还可以看出,深度学习算法和 BRATDDL 模型的训练速度比张量分解算法更快,第二次迭代训练中已将误差收敛至 1.4 左右,第三次迭代后误差即稳定于 0.9 左右。而张量分解算法在第三次迭代时才将误差收敛至 1.4 左右,此后误差仍有持续波动。BRATDDL 模型比深度学习的平均误差降低 41.15%,比张量分解算法的平均误差降低 59.52%。可见混合模型比其他两个独立算法在收敛速度和误差上都有一定程度的降低。

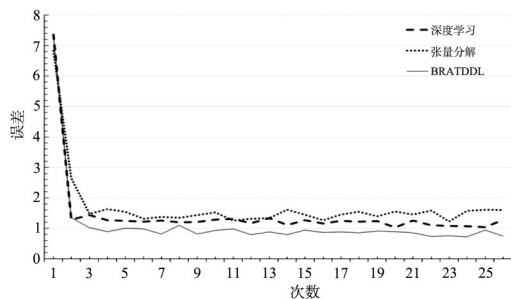


图 6 单一推荐模型和混合推荐模型损失值对比图

Fig. 6 Loss value of single recommendation model and mixed recommendation model

## 5.3 BRATDDL 推荐算法与协同过滤推荐算法对比

图 7 展示了现今几大主流推荐算法和 BRATDDL 算法在极度稀疏的 Movielens-1M 数据集上的表现,除 BRATDDL 外其他主流推荐算法的误差均引自文献[18],评价指标为平均绝对误差(MAE)。由图可见,BRATDDL 模型的损失值最低,误差比奇异值分解(Support Value Decomposition,SVD)缩小 2.8%,比基于用户的协同过滤(User Collaboration Filtering,

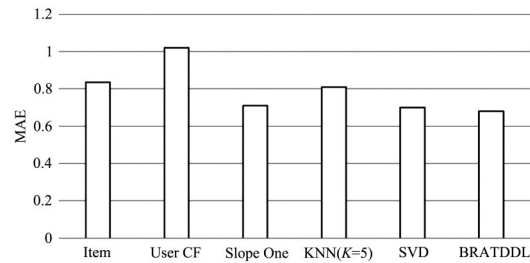


图 7 主流推荐算法和张量深度混合模型误差比较

Fig. 7 MAE of mainstream recommendation algorithm and tensor depth hybrid model

User CF)缩小 34.0%。因此在使用相同的数据集训练时,BRATDDL 混合模型比主流推荐算法模型的误差更低,推荐结果更精准。

## 6 总 结

本文分析从用户对推荐对象的评分数据中分解提取用户和推荐对象特征的张量分解和直接从用户、推荐对象及使用记录等辅助数据中提取特征的深度学习这两种方法的优点及存在的问题,发现这两种方法存在数据信息上的互补性,提出将两种方法融合的混合推荐算法 BRATDDL 算法,相互弥补单一算法提取数据来源不全面的缺陷,提高算法的推荐精度。

通过在两个数据集上的实验表明,BRATDDL 混合推荐算法的推荐精度比张量分解和深度学习单一算法有较大的提升;另外通过与奇异值分解、协同过滤等算法在相同数据集上的实验表明,BRATDDL 算法的推荐精确度也有大幅提升。综上,BRATDDL 算法能更好地实现用户的个性化推荐。

## 参 考 文 献

- [1] Ramondt S, Ramirez A S. Assessing the impact of the public nutrition information environment: adapting the cancer information overload scale to measure diet information overload. Patient Education and Counseling, 2018, 102(1): 37-42.
- [2] 黄丹. 基于张量分解的推荐算法研究. 硕士学位论文. 北京: 北京交通大学, 2016. (Hang D. Research on tensor factorization-based recommenda-

- tion algorithm. Master Dissertation. Beijing: Beijing Jiaotong University, 2016.)
- [3] 刘凯鹏, 方滨兴. 一种基于社会性标注的网页排序算法. 计算机学报, 2010, 33(6): 1014–1023. (Liu K P, Fang B X. A novel page ranking algorithm based on social annotations. Chinese Journal of Computers, 2010, 33(6): 1014–1023.)
- [4] Zhang Z K, Liu C. Hypergraph model of social tagging network. Journal of Statistical Mechanics Theory and Experiment, 2010, doi: 10.1088/1742-5468/2010/10/P10005.
- [5] Hofmann T. Latent semantic models for collaborative filtering. ACM Transactions on Information Systems, 2004, 22(1): 89–115.
- [6] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms//Proceedings of the 10<sup>th</sup> International Conference on World Wide Web. Hong Kong, China: ACM, 2001, doi: 10.1145/371920.372071.
- [7] Guo H F, Tang R M, Ye Y M, et al. DeepFM: a factorization-machine based neural network for CTR prediction//Proceedings of the 26<sup>th</sup> International Joint Conference on Artificial Intelligence. Melbourne, Australia: IJCAI, 2017, doi: 10.24963/ijcai.2017/239.
- [8] Pan R, Dolog P, Xu G. KNN-based clustering for improving social recommender systems//Cao L, Zeng Y, Symeonidis A L, et al. Agents and Data Mining Interaction. Springer Berlin Heidelberg, 2012, doi: 10.1007/978-3-642-36288-0\_11.
- [9] Liao H, Mcdermott E, Senior A. Large scale deep neural network acoustic modeling with semi-supervised training data for Youtube video transcription//2013 IEEE Workshop on Automatic Speech Recognition and Understanding. Olomouc, Czech Republic: IEEE, 2013, doi: 10.1109/ASRU.2013.6707758.
- [10] 邓腾飞. 个性化音乐推荐系统的研究. 硕士学位论文. 广州: 华南理工大学, 2018. (Deng T F. Research of personalized music recommender system. Master Dissertation. Guangzhou: South China University of Technology, 2018.)
- [11] Symeonidis P, Nanopoulos A, Manolopoulos Y. Tag recommendations based on tensor dimensionality reduction//Proceedings of the 2008 ACM Conference on Recommender Systems. Lausanne, Switzerland: ACM, 2008: 43–50.
- [12] Rendle S, Schmidt-Thieme L. Pairwise interaction tensor factorization for personalized tag recommendation//Proceedings of the 3<sup>rd</sup> ACM International Conference on Web Search and Data Mining. New York, NY, USA: ACM, 2010: 81–90.
- [13] Hsieh C K, Yang L, Cui Y, et al. Collaborative metric learning//Proceedings of the 26<sup>th</sup> International Conference on World Wide Web. Perth, Australia: International World Wide Web Conferences Steering Committee, 2017: 193–201.
- [14] 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. 计算机学报, 2018, 41(7): 1619–1647. (Huang L W, Jiang B T, Lv S Y, et al. Survey on deep learning based recommender systems. Chinese Journal of Computers, 2018, 41(7): 1619–1647.)
- [15] 廖志芳, 李玲, 刘丽敏等. 三部图张量分解标签推荐算法. 计算机学报, 2012, 35(12): 2625–2632. (Liao Z F, Li L, Liu L M, et al. A tripartite decomposition of tensor for social tagging. Chinese Journal of Computers, 2012, 35(12): 2625–2632.)
- [16] Ross I M. An optimal control theory for nonlinear optimization. Journal of Computational and Applied Mathematics, 2019, 354: 39–51.
- [17] Hernando A, Bobadilla J, Ortega F. A non negative matrix factorization for collaborative filtering recommender systems based on a Bayesian probabilistic model. Knowledge-Based Systems, 2016, 97: 188–202.
- [18] Spiegelberg J, Ruz J, Pelckmans K. Tensor decompositions for the analysis of atomic resolution electron energy loss spectra. Ultramicroscopy, 2017, 175: 36–45.

(责任编辑 杨可盛)