



计算机应用研究
Application Research of Computers
ISSN 1001-3695, CN 51-1196/TP

《计算机应用研究》网络首发论文

题目: 融合评分矩阵和评论文本的深度神经网络推荐模型
作者: 周传华, 于猜, 鲁勇
DOI: 10.19734/j.issn.1001-3695.2020.04.0089
收稿日期: 2020-04-20
网络首发日期: 2020-10-10
引用格式: 周传华, 于猜, 鲁勇. 融合评分矩阵和评论文本的深度神经网络推荐模型
[J/OL]. 计算机应用研究. <https://doi.org/10.19734/j.issn.1001-3695.2020.04.0089>



网络首发: 在编辑部工作流程中, 稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定, 且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式 (包括网络呈现版式) 排版后的稿件, 可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定; 学术研究成果具有创新性、科学性和先进性, 符合编辑部对刊文的录用要求, 不存在学术不端行为及其他侵权行为; 稿件内容应基本符合国家有关书刊编辑、出版的技术标准, 正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性, 录用定稿一经发布, 不得修改论文题目、作者、机构名称和学术内容, 只可基于编辑规范进行少量文字的修改。

出版确认: 纸质期刊编辑部通过与《中国学术期刊 (光盘版)》电子杂志社有限公司签约, 在《中国学术期刊 (网络版)》出版传播平台上创办与纸质期刊内容一致的网络版, 以单篇或整期出版形式, 在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊 (网络版)》是国家新闻出版广电总局批准的网络连续型出版物 (ISSN 2096-4188, CN 11-6037/Z), 所以签约期刊的网络版上网络首发论文视为正式出版。

融合评分矩阵和评论文本的深度学习推荐模型^{*}周传华^{1,2,3}, 于猜^{1†}, 鲁勇¹

(1. 安徽工业大学 管理科学与工程学院, 安徽 马鞍山 243002; 2. 中国科学技术大学 计算机科学与技术学院, 合肥 230026; 3. 复杂系统多学科管理与控制安徽普通高校重点实验室, 安徽 马鞍山 243002)

摘要: 针对个性化推荐中用户评分矩阵数据集稀疏, 用户和项目描述信息未充分利用的问题, 提出融合评分矩阵和评论文本的深度学习推荐模型(deep neural network recommendation model, DeepRec)。首先将通过数据预处理得到的用户偏好特征和项目属性特征的文本集合分别输入到卷积神经网络进行训练, 得到用户和项目的深层次非线性特征; 同时, 将评分矩阵输入多层感知机得到用户偏好隐表示, 并对两种模型提取的用户偏好隐表示进行融合; 其次利用多层感知机建模用户和项目隐表示对用户进行个性化推荐; 最后基于3组数据集以均方根误差为评估指标进行对比实验, 结果表明DeepRec的预测误差更低, 有效提高推荐的精准度。

关键词: 评分矩阵; 评论文本; 卷积神经网络; 多层感知机; 数据稀疏

中图分类号: TP301.6 **doi:** 10.19734/j.issn.1001-3695.2020.04.0089

Recommendation model of deep neural network combining rating matrix and review text

Zhou Chuanhua^{1,2,3}, Yu Cai^{1†}, Lu Yong¹

(1. School of Management Science & Engineering, Anhui University of Technology, Ma'anshan Anhui 243002, China; 2. School of Computer Science & Technology, University of Science & Technology of China, Hefei 230026, China; 3. Key Lab. of Multidisciplinary Management & Control of Complex Systems of Anhui Higher Education Institutes, Ma'anshan Anhui 243002, China)

Abstract: In view of the problems of sparse user rating matrix data set in personalized recommendation and insufficient utilization of user and item description information, The paper proposed a deep neural network recommendation model (DeepRec) that combines the rating matrix and comment text. Firstly, through data preprocessing to obtain a text collection of user preference features and item attribute features, and then input the text collections into the convolutional neural network for training to obtain deep nonlinear features of users and items; at the same time, Input rating matrix into the multi-layer perception to get the user implicit representation, and integrate the user implicit representations; secondly, use multi-layer perceptron to model users and item implicit representations to personalize recommendations for users; finally, based on 3 sets of data sets to mean square The root error is an evaluation index for comparison experiments. The results show that the prediction error of DeepRec is lower, which effectively improves the accuracy of the recommendation.

Key words: rating matrix; review text; convolutional neural network; multi-layer perceptron; data sparse

0 引言

在信息爆炸的时代, 如何从数据中挖掘和利用有价值的信息成为当今学者研究的趋势。作为缓解信息过载的技术, 推荐系统被广泛应用在电子商务^[1]、在线新闻^[2]、智能交通^[3]和社交网站^[4]等领域。推荐系统根据用户的历史行为数据向用户推荐其感兴趣的项目。为了精确的预测用户的偏好, 其必须从海量的数据中学习用户的历史行为信息, 然后基于用户偏好之间的相似性来选择项目。

目前应用最成功的推荐系统大多都是基于协同过滤技术, 其主要思想是以往具有相同偏好的用户群体在未来会作出相似的选择。其中应用最广泛的是隐语义模型(latent factor model, LFM)^[5], 该算法将评分矩阵分解为用户偏好矩阵和项目特征矩阵, 通过点积的形式预测用户对项目的评分。然而, 在实际条件下, 协同过滤方法通常面临着数据稀疏和冷启动问题。此外, 经典的协同过滤方法采用的是浅层模型, 无法学习到用户和项目的深层次特征。随着互联网中越来越多的

数据被感知获取, 融合多源异构辅助信息的推荐方法深受研究学者的重视^[6-8]。该方法能缓解推荐系统中的冷启动和数据稀疏问题, 但是辅助信息往往具有多模态、难提取、数据异构和分布不均匀等特征。因此融合多元异构数据的推荐方法依然面临着严峻的挑战。

随着数据的指数式增长、GPU等计算资源的增加以及算法的改进, 深度学习已成为许多领域极具影响力的工具。例如, 深度学习模型已经成功的应用在计算机视觉^[9]、语音识别^[10]和自然语言处理^[11,12]等领域, 并显示出巨大的潜力。然而, 相对于传统的推荐系统算法的大量研究, 使用深度学习的工作要少得多。

Meshal等人提出DeepHCF^[13](deep hybrid collaborative filtering)是目前理论和实验效果都比较出色的融合模型。该模型使用深层神经网络挖掘用户和项目的隐表示, 将用户和项目隐表示输入因子分解机预测评分。本文对DeepHCF进行改进, 主要包括以下3个方面的工作:

a) 结合评分矩阵和评论文本提出了一种基于深度学习

收稿日期: 2020-04-20; 修回日期: 2020-06-07 基金项目: 国家自然科学基金资助项目(71772002, 61702006); 复杂系统多学科管理与控制安徽普通高校重点实验室资助项目(CS2020-04)

作者简介: 周传华(1965-), 男, 安徽马鞍山人, 教授, 硕导, 博士, 主要研究方向为机器学习、数据挖掘、智能算法; 于猜(1995-), 女(通信作者), 河南商丘人, 硕士, 主要研究方向为数据挖掘、推荐算法、机器学习(2598977911@qq.com); 鲁勇(1995-), 男, 安徽芜湖人, 硕士, 主要研究方向为数据挖掘、智能算法。

的模型 DeepRec, 它是一种由多层感知机(multilayer perceptron, MLP)和卷积神经网络(convolutional neural network, CNN)组成的推荐方法, 该模型将两种不同的数据类型作为输入。

b) 针对 MLP 和 CNN 提取的用户隐表示以相加的方式进行合并, 然后将用户和项目隐表示串联, 并采用 MLP 对目标用户评分进行预测。

c) 在 3 组数据集上进行实验, 结果表明 DeepRec 模型的预测误差更低。

1 相关工作

1.1 基于评分矩阵的推荐方法

在基于评分矩阵的推荐方法中, 研究者主要提出了基于邻域和基于模型的推荐方法, 最常见的是基于邻域的模型。其中, 基于用户的协同过滤方法通过计算用户之间的相似度, 通过排序找到与目标用户相似度最高的用户来预测未知评分, 基于项目的协同过滤则是依据项目之间的相似度进行预测。

目前最流行的基于模型的推荐方法之一是 Koren 等人^[5]提出的隐语义模型, 该方法将用户偏好和项目属性表达为 2 个隐向量, 最后将两个向量的点积作为用户对项目的评分。随后, Salakhutdinov 等人^[14]提出概率矩阵分解方法(probabilistic matrix factorization, PMF), 在奇异值分解(SVD)的基础上进一步改进 LFM。多项研究表明基于模型的推荐明显优于基于邻域的方法。

尽管基于矩阵分解的推荐方法取得了良好的推荐效果, 然而评分的稀疏性问题一直制约着该模型的发展。因此, 近年来研究者通过挖掘评论文本信息来进一步提高模型预测的准确度。

1.2 融入评论文本的推荐方法

在文本的自然语言处理领域中, 基于文本内容的挖掘方法有着广泛的研究基础。2009 年, Jakob 等人^[15]最早使用评论文本, 其认为评论文本中关于价格、服务、情感等描述能够提高模型的推荐质量。但是其仅停留在这种能够一一对应的显性特征, 未能进一步挖掘潜在的隐表示。Huang 等人^[16]使用潜在狄利克雷概率模型(latent Dirichlet allocation, LDA)从 Yelp 数据集中挖掘子话题来构建隐表示特征预测评分。Mukherjee 等人^[17]通过实验发现对评论文本进行情感分析也有助于提高预测精度。李琳等人^[18]将评论主题与矩阵分解隐因子融合, 通过添加潜在主题分布作为预测评分引导项。上述的相关工作大多是通过主题模型挖掘评论文本中用户和项目的潜在特征分布, 但是基于词袋的模型无法保留词序信息, 忽略了情感分析中的局部上下文信息; 其次, 这些方法学习到的都是浅层的线性特征, 不能对非线性的隐藏特征进行挖掘。深度学习模型能够保留词序信息, 进而提高推荐的质量, 因此, 主题模型逐渐被深度学习模型所取代。

2015 年 Li 等人^[19]提出 DCL(deep collaborative filtering)将深度学习融合协同过滤方法, 通过边缘降噪自编码器与概率矩阵分解相结合, 极大地提高了目标用户的推荐质量。2016 年 Kim 等人^[20]提出 ConvMF+(convolution matrix factorization)考虑文本的局部词序信息, 产生更准确的项目隐向量。然而, ConvMF+仅考虑使用项目的文本信息, 忽略了用户的文本信息。2017 年 Zheng 等人^[21]提出 DeepCoNN(deep cooperative neural network), 其使用两个并行的神经网络提取用户和项目的隐表示, 但是 DeepCoNN 未利用评分矩阵。He 等人^[21]利用深度神经网络建模用户与项目特征之间的复杂交互, 提出一种神经协同过滤方法(neural collaborative filtering, NCF), 然而 NCF 仅考虑到用户、项目的显式反馈信息, 忽略了用户和项目评论等隐式信息。

与本文相似的研究是 Meshal 等人提出 DeepHCF^[13](deep hybrid collaborative filtering), 其通过深度神经网络提取用户和项目隐表示, 利用因子分解机对用户和项目的隐表示模型预测评分。然而 DeepHCF 仅仅考虑了项目的文本信息, 并且没有考虑到用户和项目隐表示的非线性交互。因此, 本文提出 DeepRec 模型, 通过使用 MLP 从评分矩阵中提取用户隐表示, 同时使用两个并行的 CNN 分别对用户评论集和项目评论集进行处理, 提取用户和项目的隐表示, 最后通过 MLP 学习用户和项目隐表示的非线性特征, 并对目标用户对项目的评分进行预测。

2 模型

本文提出的推荐模型结构如图 1 所示。DeepRec 模型由三个并行的子网络和顶端的融合和预测层构成。子网络分别是 MLP 和 CNN, 通过联合训练的方式预测最终评分, 最大程度的减小训练误差。模型结构图中各个模块的具体细节将在下面的三个小节进行介绍。

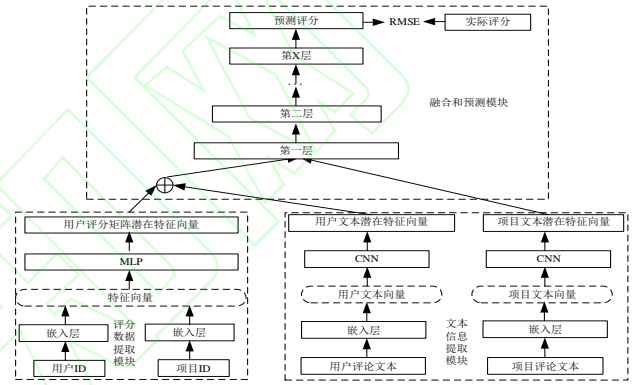


图 1 模型整体结构

Fig. 1 Model overall structure

2.1 用户评分潜在隐向量提取模块

MLP 具有多个神经元层, 能对非线性数据进行处理。DeepRec 模型中 MLP 包含三个主要组成部分: 输入层, 嵌入层, 隐藏层。用户 ID 和项目 ID 作为 MLP 模型的输入, 用户 ID 和项目 ID 经过嵌入层被嵌入到不同的隐空间, 然后将两个嵌入层串联输入到隐藏层。嵌入层是一个函数, 即 $f: X \rightarrow \mathbb{R}^n$, 将用户偏好和项目属性从较大的稀疏分类域映射到稠密矩阵。后面的隐藏层呈塔状, 意味着每个隐藏层的神经元数量都比前一层少, 神经元的激活值 $a_j^{(\ell)}$ 计算如下:

$$a_j^{(\ell)} = \sigma(\sum_{i=1}^n a_i^{(\ell-1)} w_{ij}^{(\ell)} + b_j^{(\ell)}) \quad (1)$$

其中, ℓ 表示当前神经元的层数, σ 表示非线性激活函数。在 Sigmoid, tanh 和 Relu 中, 相对于 Sigmoid, tanh 函数, Relu 函数收敛速度更快, 并且可以有效避免梯度消失问题。因此在 MLP 架构中使用 Relu 函数作为激活函数。其被定义如下:

$$f(x) = \max\{0, x\} \quad (2)$$

2.2 评论文本潜在特征向量提取模块

CNN 模型通过从用户和项目的描述文档捕获上下文信息生成文档隐藏特征向量, 将得到的文档隐表示用于推荐。本文用于提取用户和项目评论文本的模型由四层构成: 嵌入层、卷积层、池化层和全连接层。CNN 模型的框架如图 2 所示。

1) 嵌入层

嵌入层将用户和项目评论文本的稀疏表示转换为稠密矩阵。具体地说, 该过程把评论文本看做一个具有 m 个词的序列文档, 通过连接文档中各个词向量将文档表示为矩阵。词嵌入向量使用预训练的词嵌入模型 Glove^[22]进行初始化。得到的文本矩阵 $D \in R^{m \times n}$ 表示为

$$D = \begin{bmatrix} w_{11} & \cdots & w_{1i} & \cdots & w_{1n} \\ w_{21} & \cdots & w_{2i} & \cdots & w_{2n} \\ \vdots & & \vdots & & \vdots \\ w_{m1} & \cdots & w_{mi} & \cdots & w_{mn} \end{bmatrix} \quad (3)$$

其中, m 表示单词的数量, n 表示嵌入的维度, $w_{[i,j]}$ 表示第 i 个词的向量形式。经过嵌入层的转换之后, 每个词的维度变为 n 维。

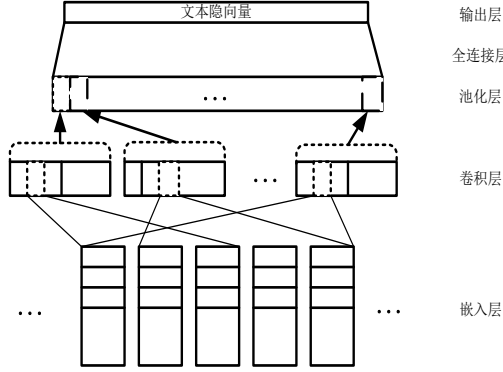


图 2 CNN 模型

Fig. 2 Cnn model

2) 卷积层

卷积层用来提取评论文本的上下文特征向量。具体地, 上下文特征 $c^j \in \mathbb{R}$ 由第 j 个共享权值 $W_c^j \in \mathbb{R}^{p \times ws}$ 提取, 共享权值的窗口大小 ws 决定了周围词的数量:

$$c^j = f(W_c^j * D_{c,j+ws-1} + b_c^j) \quad (4)$$

其中, $*$ 是卷积算子, $b_c^j \in \mathbb{R}$ 是 W_c^j 的偏置, f 为非线性激活函数。在 CNN 中使用 Relu 函数作为激活函数。然后, 通过公式(5)构造上下文特征向量 $c^j \in \mathbb{R}^{l-ws+1}$ 。一个共享权值仅捕获了一种类型的上下文特征, 因此使用多个共享权值来捕获多种类型的上下文特征。

$$c^j = (c_1^j, c_2^j, c_3^j, c_4^j, \dots, c_{l-ws+1}^j) \quad (5)$$

3) 池化层

文档经过卷积操作得到的上下文特征向量具有可变的长度。这种表示存在两个问题: (1)存在的大多数上下文特征对增强性能没有帮助; (2)上下文特征向量的可变长度使得很难构建接下来的层。因此利用最大池化操作, 从每个上下文特征向量中仅提取最大上下文特征。将文档表示缩减至 n_c 个固定长度, 如下所示。

$$df = [\max(c^1), \dots, \max(c^j), \dots, \max(c^{n_c})] \quad (6)$$

4) 全连接层

全连接层对所有特征进行加权处理获得用户和项目的隐表示, 在全连接层将用户和项目特征分别进行全连接处理, 其操作定义如下:

$$u_i = f(W \times df + b) \quad (7)$$

最后, 通过 CNN 模型得到用户评论和项目评论的隐表示。

2.3 特征融合和评分预测模块

用户的深度特征 u_i 同时包含了 MLP 模型学习到的用户隐表示 $u_{R,i}$ 和 CNN 模型从用户评论文本中学习的用户隐表示 $u_{text,i}$, 其融合方式如下:

$$u_i = u_{R,i} + u_{text,i} \quad (8)$$

用户深度特征联合 CNN 模型提取的项目隐表示 v_j 作为 MLP 模型的输入向量, 进而预测项目评分。MLP 的定义如下:

$$\begin{aligned} z_1 &= \text{concatenate}(u_i, v_j), \\ \phi_1(z_1) &= a_1(W_1^T z_1 + b_1), \\ &\dots \\ \phi_K(z_K) &= a_K(W_K^T z_K + b_K), \\ y_{ui} &= \sigma(h^T \phi_K(z_K)) \end{aligned} \quad (9)$$

其中, W_x 表示权值矩阵, b_x 表示偏置向量, a_x 是第 x 个隐藏层的激活函数, h^T 是输出层的边缘权重向量, y_{ui} 表示预测评分。MLP 网络结构的设计遵循 3.1 节 MLP 的塔式结构。

2.4 模型学习

对用户和项目建模预测评分实际上是一个回归问题, 常用的目标函数为平方损失:

$$Loss = \sum_{u,i \in \Omega} (y_{ui} - y_{ui})^2 + \lambda \|\Theta\|_2^2 \quad (10)$$

其中, Ω 表示训练集中的样本, y_{ui} 表示用户对项目的真实评分。 λ 是 L2 正则化项的常数参数, Θ 表示模型的所有参数。为了最小化损失函数, 采用 Adam(adaptive moment estimation) 作为优化器进行优化, 利用梯度的一阶、二阶矩估计动态调整每个参数的学习率, 使模型更快收敛。

3 实验结果与分析

3.1 实验数据

实验采用亚马逊评论公共数据集 Amazon 5-core 中的 3 个子类别数据集 Toys and Games, Instant Video 和 Digital Music。三个数据集均包含用户对项目 1 到 5 之间的显示评分, 以及来自 Amazon 网站 1996 年 5 月到 2014 年 7 月之间的用户真实评论。表 1 显示了各个数据集的基本统计信息。

表 1 数据集基本信息

Tab. 1 Basic information of dataset

数据集	用户	项目	评分数	稀疏度
Toys and Games	19412	11924	167597	92.75%
Instant Video	5130	1685	37126	99.57%
Digital Music	5541	3568	6476	99.67%
Average	32643	6154	284923	99.72%

3.2 对比方法

本文采用的对比方法如表 2 所示, LFM, SVD++ 在参数学习方面仅仅使用了评分矩阵, 后面的 ConvMF、DTMF+ 和 DeepHCF 同时使用了评分矩阵和评论文本。DeepCoNN 仅使用了评论文本进行参数的学习。

表 2 对比方法

Tab. 2 Comparison method

	评分矩阵	评论文本	深度学习
LFM	✓	\	\
SVD++	✓	\	\
DTMF+	✓	✓	\
DeepCoNN	\	✓	✓
ConvMF+	✓	✓	✓
DeepHCF	✓	✓	✓
DeepRec	✓	✓	✓

LFM^[5]: 这是最经典的矩阵分解算法, 其通过矩阵分解方法对未知项目评分进行预测。

SVD++^[24]: 在奇异值分解(SVD)的基础上加入隐式反馈, 使用用户的历史数据来进一步提高模型预测精度。

DTMF+^[18]: 将用户评论集和商品评论集的主题分布作为正则化项融入到矩阵分解模型, 然后将两个主题分布作为偏好引导项进一步提高推荐的质量。

DeepCoNN^[20]: 通过 CNN 将用户评论集和项目评论集结合起来对用户和项目进行建模。

ConvMF+^[19]: 将预训练的词向量嵌入模型引入卷积矩阵分解模型, 从项目评论文本中学习项目隐表示。

DeepHCF^[13]: 结合 CNN 和 MLP 分别从评分和项目评论中挖掘用户和项目的潜在特征, 并使用因子分解方法预测评分。

3.3 评价指标

为了衡量本文所提出算法的预测准确度, 实验采用均方

根误差(root mean square error, RMSE)来验证所得到的预测结果和真实评分的差异, 计算公式如下:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_{ui} - \hat{y}_{ui})^2} \quad (11)$$

其中, n 表示测试集的大小, RMSE 值越小, 表明预测值和真实值越接近, 模型预测结果的准确率越高。

3.4 实验设置

本文使用 TensorFlow 架构对 DeepRec 模型进行实现, 并使用 GPU(GTX1660)对训练进行加速。本文将每个数据集随机划分为训练集(80%), 验证集(10%)、测试集(10%)。在验证集上进行超参数的选取, 在测试集上进行模型评估。对比算法均根据对应论文进行参数初始化, 通过微调使其达到最佳性能。对于 DeepRec 模型通过网格搜索法从 $\{1, 10, 100\}$ 确定用户和项目的正则项系数。当 $\lambda_i = 1$, $\lambda_j = 100$ 时 DeepRec 模型预测误差最小, 使模型的性能发挥到最佳。

针对卷积神经网络对文本数据的处理部分, 在 Instant Video 数据集比较 50、200、300 维预训练的词嵌入模型, 如图 3 所示, 当词向量的嵌入维度为 300 时, 模型的预测误差最低, 因此本文采用词嵌入的维度为 300 维。

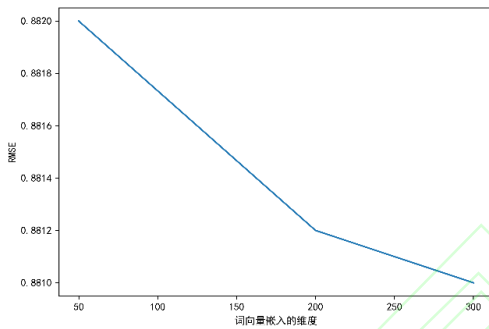


图 3 词向量嵌入维度的对模型性能的影响

Fig. 3 Impact of word vector embedding dimension

3.5 性能比较

本文提出的 DeepRec 模型和其他对比方法在 Amazon 的 3 个子数据集的预测结果如表 3 所示, 通过对实验结果进行分析, 可以得到以下结论。

表 3 实验结果

Tab. 3 Experimental results

	Toys and Games	Instant Video	Digital Music
LFM	0.898	0.973	0.922
SVD++	0.896	0.972	0.915
DTMF+	0.899	0.965	0.906
DeepCoNN	0.894	0.962	0.899
ConvMF+	0.849	0.917	0.858
DeepHCF	0.835	0.909	0.832
DeepRec	0.799	0.881	0.802
提升(%)	4.31%	3.08%	3.62%

与基于评分数据的方法(LFM, SVD++)相比, 利用评论文本信息(DTMF+, DeepCoNN, DeepHCF, DeepRec)的方法具有更好的推荐质量。因为评论文本中包含的有关用户偏好和商品属性信息可以对评分数据进行补充。

在同样考虑评论文本的推荐模型中, 基于深度学习的模型(DeepCoNN, ConvMF+, DeepHCF, DeepRec)较于传统模型DTMF+具有更优的推荐性能。这是因为传统模型大都采用基于 LDA 主题模型的方式处理文本数据, 已有大量研究工作[20, 21]表明基于深度学习的技术在文本分析方面比 LDA 主题模型更优, 而且传统模型仅采用线性特征表示的方式, 深度学习模型则能够以非线性的方式对用户和项目建模, 另外深度学习模型采用的 dropout 技术可以避免模型陷入过度拟

合。

DeepRec 模型在所有的基准方法中表现最佳, 较 DeepHCF 模型平均预测误差降低了 3.67%。相比仅使用评论文本的深度学习模型 DeepCoNN, ConvMF+, DeepHCF 和 DeepRec 通过加入评分数据的用户隐表示, 获得更好的表现效果。这表明融合评分矩阵和评论文本进行建模能够极大的提高推荐的精准度。虽然 DeepHCF 模型取得不错的推荐效果, 但是其未考虑到用户评论数据隐表示, 而且没有涉及到用户和项目隐表示的非线性特征交互。所以本文提出的方法 DeepRec 对评论文本进行处理时, 同时提取用户和项目的隐表示, 并使用 MLP 提取用户和项目的隐表示的非线性特征。实验结果表明, DeepRec、DeepCoNN、DeepHCF、ConvMF+ 具有更优的性能。

3.6 DeepRec 模型性能分析

为了探讨超参数对模型有效性的影响, 本文进行了三组参数分析实验, 分别是: 隐因子的数量, MLP 中隐藏层的数量和 Dropout 比率。

a) 隐因子的个数。针对不同隐因子的数量[16, 32, 64, 128]在 Instant Video 数据集展开实验。实验结果如图 4 所示, 当隐因子的数量为 64 时获得的推荐效果最佳。说明适当增加隐因子的数量能够提升推荐效果, 但是当隐因子的数量超过 64 时, 不但不能提高推荐质量, 还会增加时间和空间复杂度。

b) MLP 隐藏层的数量。为了研究 MLP 隐藏层的层数对模型性能的影响, 设置隐藏层的数量分别为[1, 2, 3, 4], 三个数据集的实验结果如图 5 所示。随着隐藏层数量的增加, 三个数据集上的 RMSE 先减小后增加。这是因为当隐藏层的数量太少时, 模型无法从评分矩阵中捕获有效信息; 而当隐藏层的数量太多, MLP 学习到的参数太多导致模型过拟合。

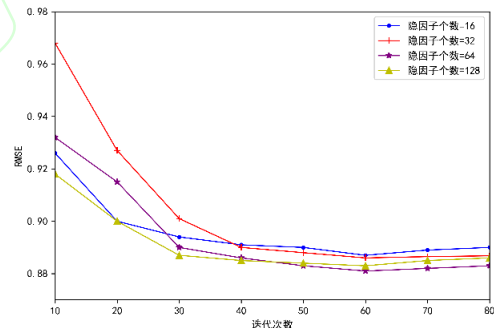


图 4 隐因子的个数对模型的影响

Fig. 4 Impact of the number of hidden factors on the model

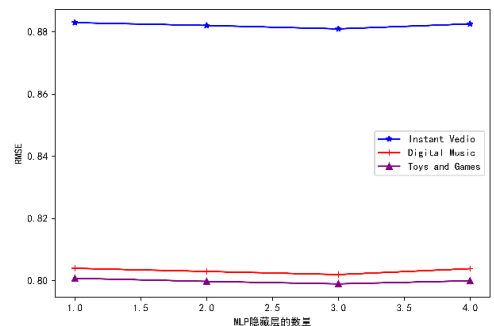


图 5 MLP 隐藏层的数量对模型的影响

Fig. 5 Impact of the number of MLP hidden layers on the model

c) Dropout 比率。图 6 所示的实验结果表明, 将 DeepRec 的 Dropout 比率设置为 0.5 时, DeepRec 在三个数据集上均取得最佳效果。从实验结果可以看出 RMSE 呈现先增加再减小, 然后增加的趋势, 当 Dropout 比率为 0.5 时, RMSE 最小。这表明 Dropout 技术可以防止模型过拟合, 从而获得更好的泛化性能。

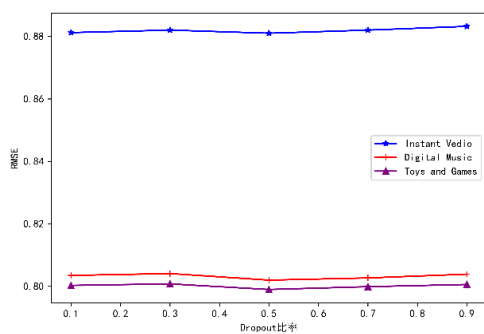


图 6 Dropout 比率对模型性能的影响

Fig. 6 Impact of dropout ratio on model performance

4 结束语

本文利用 MLP 获取评分矩阵的用户隐表示, 同时通过深度神经网络挖掘用户评论文本和项目评论文本的隐藏特征表示, 生成包含用户偏好特征和项目属性特征集, 利用 MLP 对非线性数据处理的优势对目标用户的评分进行预测。在此基础上, 基于 Amazon 数据集的比较实验结果表明 DeepRec 模型有效提高推荐的精准度。

参考文献:

- [1] Huang Zhenhua, Shan Guangxu, Cheng Jiujun, *et al.* TRec: An efficient recommendation system for hunting passengers with deep neural networks [J]. *Neural Computing and Applications*, 2019, 31 (1): 209-222.
- [2] de Koning E, Hogenboom F, Frasincar F. News recommendation with CF-IDF+ [C]// *Proc of International Conference on Advanced Information Systems Engineering*. Springer, Cham, 2018: 170-184.
- [3] Hwangbo H, Kim Y S, Cha K J. Recommendation system development for fashion retail e-commerce [J]. *Electronic Commerce Research & Applications*, 2018, 28: 94-101.
- [4] 姜建楼, 邹伟, 王玲, 等. 社交网络大数据下贪婪式实时网站推荐算法 [J]. *计算机应用研究*, 2015, 32 (05): 1361-1364. (Lou Jianlou, Zou Wei, Wang Ling, *et al.* User-based greedy real-time websites recommendation algorithm for big data in social network [J]. *Application Research of Computers*, 2015, 32 (05): 1361-1364.)
- [5] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems [J]. *Computer*, 2009, 42 (8): 30-37.
- [6] 李琳, 朱阁, 解庆, 等. 一种潜在特征同步学习和偏好引导的推荐方法 [J]. *软件学报*, 2019, 30 (11): 3382-3396. (Li Lin, Zhu Ge, Xie Qing, *et al.* Recommendation approach by simultaneous learning latent features and preferences guidance. *Journal of Software*, 2019, 30 (11): 3382-3396)
- [7] Rafailidis D, Kefalas P, Manolopoulos Y. Preference dynamics with multimodal user-item interactions in social media recommendation [J]. *Expert Systems with Applications*, 2017, 74: 11-18.
- [8] Kiran R, Kumar P, Bhasker B. DNNRec: A novel deep learning based hybrid recommender system [J]. *Expert Systems with Applications*, 2020, 144: 113054.
- [9] Nassar N, Jafar A, Rahhal Y. A novel deep multi-criteria collaborative filtering model for recommendation system [J]. *Knowledge-Based Systems*, 2020, 187: 104811.
- [10] Ravanelli M, Omologo M. Automatic context window composition for distant speech recognition [J]. *Speech Communication*, 2018, 101: 34-44.
- [11] 冀振燕, 宋晓军, 皮怀雨, 等. 基于深度学习的融合多源异构数据的推荐模型 [J]. *北京邮电大学学报*, 2019, 42 (06): 35-42. (Ji Zhenyan, Song Xiaojun, Pi Huaiyu, *et al.* Recommended Model for Fusing Multi-Source Heterogeneous Data Based on Deep Learning [J]. *Journal of Beijing University of Posts and Telecommunications*, 2019, 42 (06): 35-42.)
- [12] 杨丽, 王时绘, 朱博. 基于 ranking 的深度张量分解群组推荐算法 [J]. *计算机应用研究*, 2020, 37 (05): 1311-1316. (Yang Li, Wang Shihui, Zhu Bo. Ranking based hybrid deep tensor factorization model for group recommendation [J]. *Application Research of Computers*, 2020, 37 (05): 1311-1316.)
- [13] Alfarhood M, Cheng J. DeepHCF: A deep learning based hybrid collaborative filtering approach for recommendation systems [C]// *Proc of 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2018: 89-96.
- [14] Mnih A, Salakhutdinov R R. Probabilistic matrix factorization [C]// *Proc of Advances in neural information processing systems*. 2008: 1257-1264.
- [15] Jakob N, Weber S H, Müller M C, *et al.* Beyond the stars: exploiting free-text user reviews to improve the accuracy of movie recommendations [C]// *Proc of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*. 2009: 57-64.
- [16] Huang J, Rogers S, Joo E. Improving restaurants by extracting subtopics from yelp reviews [J]. *iConference 2014 (Social Media Expo)*, 2014.
- [17] Mukherjee S, Basu G, Joshi S. Incorporating Author Preference in Sentiment Rating Prediction of Reviews [M]// *Incorporating author preference in sentiment rating prediction of reviews*. 2013.
- [18] 李琳, 刘锦行, 孟祥福, 等. 融合评分矩阵与评论文本的商品推荐模型 [J]. *计算机学报*, 2018, 041 (007): 1559-1573. (Li Lin, Liu Jinxing, Meng Xiangfu, *et al.* Recommendation Models by Exploiting Rating Matrix and Review Text [J]. *Chinese Journal of Computer*, 2018, 041 (007): 1559-1573.)
- [19] Li Sheng, Kawale J, Fu Yun. Deep collaborative filtering via marginalized denoising auto-encoder [C]// *Proc of the 24th ACM International Conference on Information and Knowledge Management*. 2015: 811-820.
- [20] Kim D, Park C, Oh J, *et al.* Convolutional matrix factorization for document context-aware recommendation [C]// *Proc of the 10th ACM Conference on Recommender Systems*. 2016: 233-240.
- [21] Zheng Lei, Noroozi V, Yu P S. Joint deep modeling of users and items using reviews for recommendation [C]// *Proc of the Tenth ACM International Conference on Web Search and Data Mining*. 2017: 425-434.
- [22] He Xiangnan, Liao Lizi, Zhang Hanwang, *et al.* Neural collaborative filtering [C]// *Proc of the 26th international conference on world wide web*. 2017: 173-182.
- [23] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation [C]// *Proc of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014: 1532-1543.
- [24] Koren Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model [C]// *Proc of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2008: 426-434.