研究论文

集成因子分解机及其在论文推荐中的应用研究*

杨 辰 郑若桢 王楚涵 耿 爽 王 楠
（深圳大学管理学院 深圳 518060）

摘要：【**目的**】针对现有论文推荐方法在处理论文作者映射关系稀疏和特征表达时存在成效不足的问题，开发一种基于因子分解机和集成学习的新型论文推荐框架。【**方法**】使用卷积神经网络、网络嵌入等方法处理数据获取特征表示，将特征矩阵输入因子分解机，引入随机子空间法集成训练模型，最后通过投票机制协同后输出推荐结果。【**结果**】基于CiteULike数据集的实验结果表明，本文方法的推荐精确率、准确率和F度量分别为72.6%、69.7%和76.2%，分别比基准算法提升高于20个百分点、15个百分点和9个百分点。【**局限**】负采样过程中缺乏正负样本语义相似性的考虑，在模型的输入构造、特征处理模式方面有待进一步探究。【**结论**】集成因子分解机能在数据稀疏情况下实现特征的有效表示和利用，从而提升推荐效果。

关键词：论文推荐　因子分解机　集成学习
分类号：TP311　G250
DOI：10.11925/infotech.2096-3467.2022.0775
引用本文：杨辰，郑若桢，王楚涵等．集成因子分解机及其在论文推荐中的应用研究[J]．数据分析与知识发现，2023，7(8)：128-137.(Yang Chen, Zheng Ruozhen, Wang Chuhan,et al. Ensemble Factorization Machine and Its Application in Paper Recommendation[J]. Data Analysis and Knowledge Discovery, 2023, 7(8): 128-137.)

1 引 言

学术论文是研究者展示原始研究结果、全新发明创造或总结评论的重要载体，是知识交流传播的重要纽带，也是科研人员进行研究的必备工具。论文阅读是科研人员最为重要的日常工作之一，他们通过检索、阅读和分析文献，获取有用的信息以推进自己的研究。学术出版物被大量共享在互联网中，研究者可以便捷地进行论文检索，但是学术研究的发展使网络中的论文数量不断增加，大量的论文资源导致了信息过载[1]。面对海量数据，研究者一般会根据已获得论文列出的参考文献列表来检索他们感兴趣的潜在论文资源，但这种方法具有十分明显的局限性——引用列表中罗列的参考文献数量有限，而且论文质量也难以保证[2]。研究者也会借助学术检索工具（如谷歌学术、百度学术等）进行论文检索，输入关键字，系统就会返回含有关键字的论文列表。但这种方法对研究者的概括表达能力有很高的要求，需要对词语的语义有清晰的认识，用凝练的关键字将自己的检索需求精准地表达出来，否则无法得到实际所需要的文献资源；而且输出的文献数量庞大，还需研究者进行二次筛选[3]。由此可见，用户驱动的传统检索方法耗时费力，而且用户的主观检索需求难以保证获取论文的质量。

为解决以上问题，学术论文推荐应运而生，成为推荐系统领域的一个研究热点。论文推荐的目的是利用已知的信息（如作者、参考文献、文本等）进行特定的计算向用户推荐他们可能感兴趣的论文[4-5]。时效性是学术研究中极为关键的一个因素，学术论

通讯作者（Corresponding author）：耿爽（Geng Shuang），ORCID：0000-0001-8146-0786，E-mail：gs@szu.edu.cn。

*本文系国家自然科学基金项目（项目编号：71701134，71901150）和广东省基础与应用基础研究基金资助项目（项目编号：2019A1515011392）的研究成果之一。

The work is supported by the National Natural Science Foundation of China (Grant No. 71701134, 71901150), the Guangdong Basic and Applied Basic Research Foundation (Grant No. 2019A1515011392).

文推荐能节约研究者的检索时间，使其快速获取感兴趣的或者潜在有用的论文，进而加快知识传播速度、提高科研效率[6]。

本研究在推荐系统和学术论文推荐研究的基础上，基于因子分解机（Factorization Machine，FM）和集成学习提出一种新型论文推荐算法——集成因子分解机（Ensemble Factorization Machine，EFM）。因子分解机模型能在数据稀疏的情况下，充分训练特征分量的参数，有效地进行特征组合。集成学习能够在提升模型训练效率的同时提高算法的推荐准确性和稳健性。此外，集成因子分解机融合了基于内容和协作的特征输入，并根据数据的特性使用卷积神经网络（Convolutional Neural Network，CNN）、网络嵌入（Network Embedding）技术获取特征深层次的辅助信息，实现特征交互。本文的研究成果也能够为推荐算法在其他领域的应用提供支持和参考。

2　相关研究

2.1　论文推荐

主流的学术论文推荐算法分为三种，分别是基于协同过滤的论文推荐、基于内容的论文推荐和混合论文推荐。

（1）基于协同过滤的论文推荐方法

基于协同过滤的推荐方法是指采用与目标用户相似的用户的喜好推断目标用户感兴趣的研究内容，然后进行相应的推荐[7]。McNee 等最先使用协同过滤推荐论文，利用论文之间的引文网络创建评分矩阵，计算论文间的相似度，实现了文献的推荐[8]。毕强等通过谱聚类的方法将文献资源聚集为文本簇，采用协同过滤的方法在文本簇中寻找相似度较高的邻居，向用户推荐满足其偏好的文献[9]。李亚梅等在协同过滤方法的基础上引入科研情境修正用户相似度，从而更好地满足研究者的情境化需求[10]。

在论文推荐领域，尽管协同过滤推荐备受关注，但应用较少，因为用户-论文交互数据不足（远少于文本信息），矩阵稀疏性高，同时还面临“冷启动”问题，难以对新论文进行推荐[3,11]。

（2）基于内容的论文推荐方法

基于内容的推荐方法的原理是根据与用户交互的论文数据对用户进行建模，通过对比论文特征和用户特征推荐用户可能会感兴趣的论文[11]。“交互”通常表现为浏览、收藏、下载等行为。论文特征通常从文本内容中提取，例如论文标题、论文摘要、论文主体等。Caragea 等提出在论文引用模型中嵌入提取关键词的监督模型来更好地表示论文特征，在标准数据集上进行实验，获得了不错的效果[7]。汤志康等使用 TF-IDF 和 Word2Vec 方法得到论文的向量表示，基于余弦相似度计算其与用户记录中论文的相似度，根据计算值进行排序，生成该用户的推荐列表[12]。刘健等提出一种基于本体规则推理的论文内容推荐方法，通过计算文献的语义相似度形成有针对性的推荐[13]。

基于内容的推荐在学术论文推荐应用中被广泛使用，可以有效解决协同过滤中的数据稀疏和冷启动问题，但它忽略了论文的质量、引用量、与用户的内在联系等客观属性，数据利用程度不高；推荐质量高度依赖于对内容的处理程度；而且存在高度专业化问题，推荐的论文通常与用户已经知道的论文相似[11]。

（3）混合论文推荐方法

上述方法都有各自的优劣，混合推荐方法是将多个推荐方法融合在一起为用户推荐个性化的论文[14]。刘扬提出一种基于质量的文献混合推荐模型，当协同推荐系统面临冷启动问题时，便会转换至基于内容的推荐策略[15]。Haruna 等利用公开可用的上下文元数据推断研究论文之间存在的隐藏关联，再通过协同过滤提供个性化建议[16]。

混合推荐不仅结合了协同过滤和基于内容的方法，还引入了多种辅助信息提升推荐准确性，其中将图作为辅助信息引入推荐系统的研究备受关注，不仅能实现更准确的推荐，还具有更高的可解释性[17]。基于图的方法利用学术界固有的联系（例如引用关系、社会关系）构建图形网络，研究人员和论文是图中不同的节点，研究人员、研究人员与论文、论文与论文之间的关系可以视为节点之间的边，节点和边常在低维向量空间得到嵌入向量，补充用户或论文的特征表示。Kanakia 等使用微软学术图谱，利用标题和论文摘要为所有文档建立推荐列表，从而将图和基于内容的方法结合起来[18]。王

勤洁等基于科技文献网络，通过融合学者偏好的元路径计算学者-文献相关度，将相关度排序得到推荐列表[19]。基于混合推荐的学术论文推荐通过引入辅助数据（论文内容信息、社会化标注信息等），可以在一定程度上缓解数据稀疏问题，但也面临着数据的有效表示问题。

基于内容的论文推荐忽略了论文与用户内在深层的联系，而且对内容的处理程度有限，特征提取不足。在基于协同过滤的论文推荐中，用户-论文矩阵中的数据往往非常稀疏，使得计算难度提高、结果准确性降低，数据中的有效信息也无法得到充分利用。混合论文推荐结合了以上两种方法，并可以引入多种辅助信息（如用户社交网络、知识图谱等），旨在扬长避短[20]。综合前人研究，本文提出一种混合推荐方法——集成因子分解机。因子分解机对稀疏数据有强大的参数学习能力，能够解决一般的协同过滤方法在稀疏矩阵下的固有局限，因此本文以因子分解机为基础，通过深度学习方法训练获得论文推荐场景中的文本、交互及其他内容或协作信息的特征表示，在因子分解机提供的特征范式上进行特征组合，提升数据利用程度，克服基于内容的推荐方法固有的不足。特征空间的扩大会降低训练效率，因此本文设计了一种基于随机子空间的改进集成学习方法，将其应用至因子分解机模型训练过程中，旨在提高训练效率并进一步提升预测效果。

2.2 集成学习

集成学习是指生成多个学习器，通过特定策略组合到一起解决特定机器学习任务[21]，具有避免过拟合、降低陷入局部最优风险、扩展搜索空间等优点[22]。常见的集成学习方法有袋装法（Bagging）[23]、提升法（Boosting）[24]和随机子空间法（Random Subspace Method）[25]。

袋装法是用通过有放回抽样从原始数据集中获取的实例样本分别训练独立的模型。为确保模型训练的充分性，每个模型训练集的实例数量与原始数据集相同。在预测阶段，通过投票表决整合多个模型的结果，以确定最终的预测输出。

提升法中模型之间是非独立的，每个基学习器的输出都会影响下一个学习器的权重。提升法基于所有实例进行训练，第一次迭代前所有实例的抽样权重相同，在训练过程中被错误分类的实例权重增加，使得新的学习器更加关注这部分实例。最经典的提升法算法是AdaBoost[24]。

随机子空间法基于特征子空间构建学习器，子空间维数小于原始特征空间，而训练样本的数量保持一致，可以有效解决样本量小、特征冗余等问题。

集成学习方法已被验证与单独使用某种学习模型相比具有更好的预测性能[26]，而且当模型之间存在差异时，集成往往会产生更好的结果[27]。

2.3 因子分解机

因子分解机是基于支持向量机和矩阵分解的一种通用模型[28]。基本的线性回归模型仅考虑单个特征，而因子分解机能够对特征进行两两组合，学习特征之间抽象的交互关系。特征交互导致模型参数估计量大大增加，参数学习在稀疏数据场景下难以进行，为解决这个问题，因子分解机使用矩阵分解估计组合特征的参数，通过引入隐向量解决数据稀疏时无法直接估计参数的问题。

二阶因子分解机交互模型如公式（1）和公式（2）所示。

$$y(\boldsymbol{X}) = w_0 + \sum_{i=1}^{n} w_i \boldsymbol{x}_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \boldsymbol{W}_{ij} \boldsymbol{x}_i \boldsymbol{x}_j \quad (1)$$

$$\boldsymbol{W}_{ij} = < \boldsymbol{V}_i, \boldsymbol{V}_j > = \sum_{f=1}^{k} v_{i,f} \cdot v_{j,f} \quad (2)$$

其中，$w_0 \in R, w \in R^n, \boldsymbol{V} \in R^{n \times k}$；$\boldsymbol{x}_i$代表特征分量$i$；$w_0$代表全局偏置；$w_i$代表对应特征的偏置；$\boldsymbol{W}_{ij}$即$< \boldsymbol{V}_i, \boldsymbol{V}_j >$是特征$\boldsymbol{x}_i$和$\boldsymbol{x}_j$交互的参数；$\boldsymbol{V}_i$是矩阵$\boldsymbol{V}$中具有$k$维的第$i$个行向量，引入该隐向量来解决数据稀疏时无法直接估计$\boldsymbol{W}_{ij}$的问题。

3 基于集成因子分解机的论文推荐算法

3.1 问题描述

在论文推荐研究中，用户与论文的交互一般体现为用户是否将论文添加到收藏夹中，所以本研究将论文推荐定义为一个二分类问题，针对特定用户，与该用户发生交互（即被收藏）的论文定义为正例，未与用户发生交互的论文定义为负例。根据数据集构建符合模型的输入和标签（0/1），通过最小化目标函数训练各个参数，最终输出用户与候选论文交互的概率，通过阶跃函数映射为预测类别，预测为1表示推荐，预测为0表示不推荐。

3.2　算法概述

本研究提出一种基于因子分解机和集成学习的改进论文推荐算法——集成因子分解机(EFM)。基于论文信息和用户-论文交互信息提取特征,生成多个特征子集;基分类器是因子分解机模型,采用基于随机子空间的改进集成方法在特征子集上进行训练,通过投票策略整合各个模型输出,得到推荐结果。

(1)构造特征向量

构造特征向量是EFM的关键步骤,EFM使用到的处理方法有:独热编码(One-Hot Encoding)[28]、卷积神经网络[29]、网络嵌入[30]等。

独热编码可以扩展特征,但会使数据变得稀疏。由于因子分解机模型在处理稀疏数据方面具有独特优势,独热编码已在因子分解机构造输入中广泛使用[31]。因此EFM亦遵循惯例,使用独热编码来编码每个用户和每篇论文。

使用卷积神经网络处理论文标题,获取每个文本标题的向量表示。卷积神经网络由输入层、卷积层、池化层、扁平化处理层、全连接网络层和输出层等组成[32]。待处理向量首先输入卷积层,经过每个卷积核生成特征映射,不同的卷积核能从不同特征空间对该输入的某一属性进行识别。得到卷积层的输出后再进行池化,二次提取特征,减少训练参数。反复进行多次卷积与池化。多个特征映射通过扁平化处理层展开拼接后作为全连接层的输入,全连接层的输出通过输出层激活后得到最终的预测结果。为应用卷积神经网络处理论文标题这样的文本信息,根据文本标题内容生成数字字典,将每一个文本标题转化为数字向量,再经过向量嵌入,每个向量都有一定维数特征;将所有标题向量表示转化为二维矩阵输入卷积层中;将多次卷积、池化(实验中使用4次卷积-池化来提取标题特征)的多维输出进行降维和矩阵重塑,输入全连接层,在输出层进行神经元随机舍弃处理,得到最终论文标题的特征表达。EFM使用的文本卷积网络的结构如图1所示。

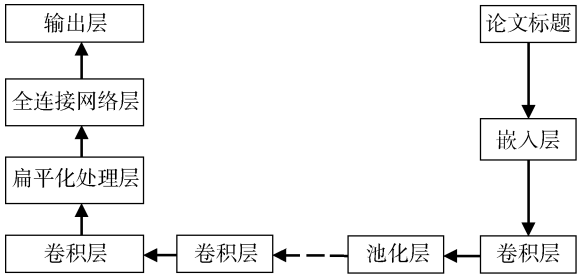

图1　卷积神经网络

Fig.1　Convolutional Neural Network

论文引用关系可以抽象为网络结构,每篇论文抽象为一个节点,论文的引用联系抽象为节点间的一条边。EFM采用网络嵌入方法结构化论文引用网络,在保留网络原有结构的同时将节点和关系嵌入低维的向量空间中,以便将之应用到因子分解机模型中。实验采用经典的网络嵌入算法DeepWalk[33]学习得到论文引用关系网络中各个节点的表示向量,并将之作为EFM模型输入的一部分。DeepWalk模型结构如图2所示,通过随机游走对网络中的节点进行采样,随机游走生成器根据引用关系图G采样一个随机顶点D_i作为随机游走序列L_{D_i}的根,从其邻域内开始均匀采样,直到达到最大长度。生成一定长度的节点序列,再用Skip-Gram模型[34]训练序列中的节点,通过最大化序列中节点之间的共现概率训练得到每个节点的向量表示。

用户收藏论文集合的信息也作为特征输入到模

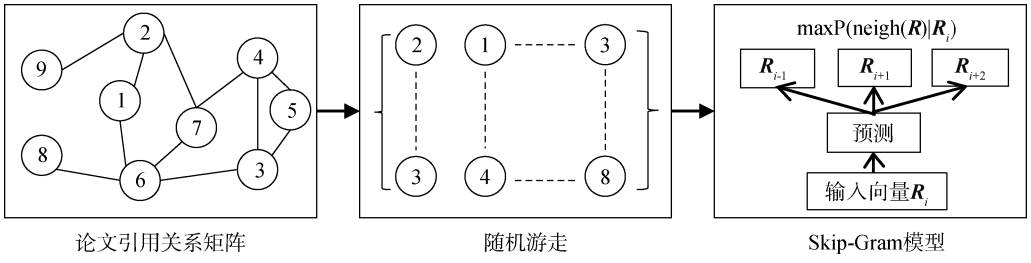

图2　DeepWalk模型

Fig.2　DeepWalk Model

型中。特征维度与论文集合数相同，每篇论文对应一个维度。非收藏论文所在维度的取值为0，收藏论文所在维度的取值为1除以该用户收藏论文集合数，即每一个特征向量的所有维取值相加总和为1。

得到特定特征输出后，再将各个特征相连接，构造出特征向量矩阵。总特征向量定义为$\boldsymbol{X}$，$\boldsymbol{X}$被分为多个非独立的特征子集$\boldsymbol{X}_1, \boldsymbol{X}_2, \cdots, \boldsymbol{X}_i$。

向量矩阵含有多个不同数值量级的特征变量，量级不一样，对目标变量的控制程度也会不一样，所以在正式输入模型前还会进行数据标准化处理，减少特征之间的差异性，提高模型的预测效果，处理方法如公式(3)所示。

$$\widehat{\boldsymbol{x}_i} = \gamma \frac{\boldsymbol{x}_i - \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{x}_i - \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i)^2 + \varepsilon}} + \beta \tag{3}$$

其中，$\boldsymbol{x}_i$代表特征数据i；$\widehat{\boldsymbol{x}_i}$代表标准化处理后的特征数据$i$；$n$代表数据总量；$\varepsilon$代表偏置项；$\gamma$和$\beta$是重构参数。

(2)模型建立与训练

因子分解机能够实现特征组合，在稀疏数据场景下也能获得良好的训练效果，所以改进算法基于因子分解机进行模型构建。原始FM模型如公式(1)所示。算法输出需在0~1之间，所以EFM模型在输出结果前通过sigmoid函数将实数映射到0~1之间。模型如公式(4)和公式(5)所示。

$$y_0 = w_0 + \sum_{i=1}^{n} w_i \boldsymbol{x}_i + \sum_{i=1}^{n}\sum_{j=i+1}^{n} < \boldsymbol{V}_i, \boldsymbol{V}_j > \boldsymbol{x}_i \boldsymbol{x}_j \tag{4}$$

$$\hat{y} = \mathrm{sigmoid}(y_0) \tag{5}$$

其中，$\boldsymbol{x}_i$代表特征分量i；w_0代表全局偏置；w_i代表对应特征的偏置；$< \boldsymbol{V}_i, \boldsymbol{V}_j >$是特征$\boldsymbol{x}_i$和$\boldsymbol{x}_j$交互的参数；$y_0$和$\hat{y}$分别表示两阶段输出。

本文研究的是二元分类问题，所以采用二元交叉熵(Binary Cross-Entropy)损失函数作为目标函数，要求最小化目标函数，如公式(6)和公式(7)所示。

$$\bar{y} = \frac{1}{1 + \mathrm{e}^{-\hat{y}}}, y \in \{0, 1\} \tag{6}$$

$$\min(L) = -y\log\bar{y} - (1 - y)\log(1 - \bar{y}) \tag{7}$$

其中，y表示用户与论文的实际交互，0代表用户与论文无交互(不收藏)，1代表用户与论文存在交互(收藏)，$\bar{y}$代表算法的预测结果，L表示目标函数。

实验使用自适应矩估计(Adaptive Moment Estimation，Adam)优化算法训练各个学习器的参数[35]。训练过程中，为尽量避免模型过拟合，将L2正则化方法用于参数更新。

(3)模型集成

在特征构造阶段，EFM在基本的用户、论文特征基础上融合文本、引用关系及论文收藏信息的特征表示，特征空间进一步扩大，同时处理所有特征向量会导致计算机资源消耗过大，降低运行效率。因此，本研究设计一种基于随机子空间的集成学习方法，在训练过程抽取部分特征训练学习器。EFM模型主要由5个同质因子分解机学习器构成，根据非独立的特征向量子集输入分别训练这5个基学习器，然后对测试集进行检测，再使用组合学习器的常用方法之一——投票集成策略，整合学习器的结果。由于基学习器之间存在差异，集成方法能够增加组合模型的多样性，带来更好的预测结果，提升推荐效果[27]。EFM算法伪代码如下。

构造输入：分别对用户、论文进行独热编码，得到特征向量$\boldsymbol{x}_1, \boldsymbol{x}_2$；
　　卷积神经网络处理论文标题文本信息，得到特征向量$\boldsymbol{x}_3$；
　　DeepWalk算法处理论文引用关系网络，得到特征向量$\boldsymbol{x}_4$；
　　处理用户收藏论文集合的信息，得到特征向量$\boldsymbol{x}_5$；
　　构造出特征向量矩阵，总特征向量定义为$\boldsymbol{x}$，$\boldsymbol{x}$被分为多个非独立的特征子集$\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_i$
输入：训练特征向量矩阵$\boldsymbol{x}_i$，学习率*learning_rate*、迭代次数*epochs*、隐向量维度k，正则化参数*w_reg*，*v_reg*等
初始化：因子分解机模型的参数$\omega_0, \omega_i, \boldsymbol{V}$，使$\omega_0$=0，$\omega_i, \boldsymbol{V}$服从标准差为0.1，均值为0的正态分布
迭代：For(*epochs*)
　　基于参数分布、目标函数、优化算法训练弱学习器$f_i, i = (1, 2, \cdots, I)$
　　For$(x, y \in \boldsymbol{X}_i)$

$\omega_0 = \mathrm{Adam}(w_0)$
End
For $i \in \{1, 2, \cdots, n\}$
$\omega_i = \mathrm{Adam}(w_i) + \mathrm{L2}(\omega_i)$
End
For $j \in \{1, 2, \cdots, n\}$
$V_i = \mathrm{Adam}(\boldsymbol{V}_i) + \mathrm{L2}(\boldsymbol{V}_i)$
End
End
输出：$\widehat{y}_i = f_i(\boldsymbol{X})$，$\widehat{y}_i \in 0/1$
If $\mathrm{Count}(\widehat{y}_i \in 0) > \mathrm{Count}(\widehat{y}_i \in 1)$
$\hat{y} = 0$
Else
$\hat{y} = 1$

4　实验和结果讨论

4.1　数据集

使用CiteULike-a数据集验证论文推荐算法的性能，原始数据来源于学术社交网站CiteULike和谷歌学术，由Wang等进行整理[36]。数据集内容包含5 551个用户对16 980篇文章产生的204 987个用户收藏论文数据，每个用户收藏论文数量都大于10，所生成的用户-项目矩阵密度为0.22%。除用户-项目信息外，本研究还利用了数据集中的论文标题和44 709条论文之间的引用信息。在负采样方面，基于全论文集合，针对每一个学术用户，根据其收藏论文集合求与全论文集合的差集，得到负样本集，随机赋予负样本一定的概率权重作为其采样可能性，最后抽取获得与该用户正样本等量的负样本。

4.2　评价指标

本文研究的问题属于二分类问题，因此可以使用分类指标衡量推荐算法的效果。使用的对比指标包括：准确率（Accuracy）、精确率（Precision）、召回率（Recall）和F度量（F-Measure）。混淆矩阵的定义如表1所示，指标计算如公式（8）-公式（11）所示。

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

$$F - Measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{11}$$

表1　混淆矩阵

Table 1　Confusion Matrix

推荐类别	用户类别	
	感兴趣	不感兴趣
推荐	True Positive（TP）	False Positive（FP）
不推荐	False Negative（FN）	True Negative（TN）

4.3　实验配置

实验操作系统环境为Windows10，算法用Python语言编写，整体基于TensorFlow 2.0实现。使用的Python版本为3.6。过程中使用到的第三方库有Numpy、Keras、Pandas、Sklearn等。

本研究的文本卷积网络实验基于Keras搭建。根据标题数字字典将每个标题转换为数字向量，再进行向量嵌入，向量每维嵌入维度*embed_dim*=8，卷积过程中滑动窗口*window_size*分别设为{2，3，4，5}，卷积核*filter_num*=8，*dropout_rate*=0.5。在网络嵌入实验中，每次随机采样的序列长度*walk_length*=10，随机游走过程中每个顶点被选取作为初始点的次数*num_walks*=80；使用Python主题模型算法库Gensim中的Skip-Gram模型训练序列，迭代次数*w2v_epoch*=3，输出的向量维度*embed_size*=64，训练窗口大小*windows*=5。

模型EFM中有三个重要的超参数——优化算法Adam的学习率*learning_rate*、训练迭代次数*epochs*和特征交互参数隐向量的维度*k*。为避免超参数取值不当对模型预测的负面影响，以单个学习器为例，选择准确率Accuracy为评价指标，使用控制变量法，在其他条件相同的情况下改变各个超参数的取值，通过对比模型准确率的高低确定最终超参数的取值。根据实验结果，最终的参数设置为*learning_rate*=0.01，*epochs*=8，*k*=16。

控制*epochs*=8、*k*=16，*batch_size*=8000，探究*learning_rate*的取值对模型预测效果的影响，如表2所示。较低的*learning_rate*会导致模型收敛速度降低，而较高的*learning_rate*会使参数在最优值邻近震荡，*learning_rate*设置为0.01最为合适。

表2　不同学习率下的模型准确率

Table 2　Model Accuracy in Different Learning Rates

learning_rate	准确率
0.001	0.560
0.005	0.616
0.01	**0.635**
0.05	0.596
0.1	0.592

控制 *learning_rate*=0.01、*k*=16，数据批大小 *batch_size*=8000，探究 *epochs* 的取值对模型预测效果的影响，如表3所示。*epochs* 过小，模型无法拟合，而 *epochs* 设置得过大，模型会过拟合，*epochs* 设置为8最为合适。

表3　不同迭代次数下的模型准确率

Table 3　Model Accuracy in Different Epochs

epochs	准确率
4	0.609
6	0.612
8	**0.635**
10	0.632
12	0.630
15	0.585

控制 *learning_rate*=0.01、*epochs*=8，数据批大小 *batch_size*=8 000，探究 *k* 的取值对模型预测效果的影响，如表4所示。当 *k*=32 时，模型的准确率最高，为0.641。当 *k*=16 时，模型准确率为0.635，与0.641相差不大，为节省计算资源，在后续实验中将 *k* 设置为16。

表4　不同隐向量维度下的模型准确率

Table 4　Model Accuracy in Different Dimensions

k	准确率
8	0.581
12	0.615
16	0.635
20	0.607
24	0.594
28	0.628
32	**0.641**

4.4　对比算法

实验选用的对比算法如表5所示。在FM和MF中，*learning_rate*、*epochs*、*k* 也保持与EFM同样的设定。在User_based CF中，实验采用Jaccard相似度公式计算用户间的相似度，根据相似用户所收藏的论文为用户进行推荐[38]。

表5　实验对比算法

Table 5　Algorithms for Comparison

算法简称	算法描述
FM	因子分解机（Factorization Machine），输入数据包括用户编码信息、论文编码信息、用户-论文交互信息，构造特征向量[28]。
MF	矩阵分解（Matrix Factorization），将用户-论文矩阵分解为低秩的用户矩阵和论文矩阵，两矩阵相乘得到预测结果[37]。
User_based CF	基于用户的协同过滤（User-based Collaborative Filtering），基于用户-论文矩阵计算用户相似度进行推荐[38]。

4.5　实验结果分析

基于上述数据集、对比算法、评价指标、设置与步骤进行对比实验，取5次实验结果的平均值为最终结果，实验结果如表6所示。从算法的准确度出发，EFM的准确率要高于其他三种算法，提升高于15个百分点。测试集中正负样本分布均衡，EFM较高的准确率能够验证其有效性，说明正确预测的样本占总待预测样本的比例较高。

表6　实验结果

Table 6　Experiment Result

算法	准确率	精确率	召回率	F度量
EFM	**0.697**	**0.726**	0.801	**0.762**
FM	0.534	0.516	0.934	0.664
MF	0.501	0.501	**0.992**	0.668
User_based CF	0.544	0.089	0.982	0.163

从精确率来看，EFM的精确率远高于其他三种算法，提升高于20个百分点。但值得关注的是EFM的召回率低于其他三种算法，FM、MF、Used_based CF的召回率都接近或者等于1。精确率衡量了用户实际感兴趣的论文占算法总推荐文章的比例，而召回率描述的是在用户实际感兴趣论文中被正确推荐的论文所占的比例。FM、MF和Used_based CF过高的召回率、较低的精确率说明它们将大部分样本都预测为正例，对应到现实中则表明算法为确保用户感兴趣的论文不被遗漏，向用户附带推荐了许多其

实际不感兴趣的论文。这样的推荐是不明智的，用户需要耗费大量的时间来进行筛选。EFM 的精确率最高，召回率也达到 80% 以上，证明了其推荐结果的优越性，而且大部分样本正例被正确预测。

F 度量反映的是精确率和召回率之间的折衷，能够综合反映算法的性能。EFM 的 F 度量值最高，达 76.2%。结合各项指标的结果，从整体上看，本研究所提出的算法 EFM 比其他基础论文推荐算法表现更好，推荐更加有效。

5　结　语

论文推荐是现阶段缓解论文资源过载和提高检索效率的有效方法，构建高效的学术论文资源推荐系统对满足研究者个性化的科研需求具有重要意义。然而，目前论文推荐算法存在无法充分利用有效信息、难以处理稀疏数据等问题。在已有研究的基础上，本研究提出集成因子分解机模型。其主要贡献有：

（1）考虑数据的属性，利用卷积神经网络、网络嵌入等深度学习技术充分挖掘论文内容中更深层抽象的特征，融合到输入特征中，提高推荐性能。

（2）将集成学习的思想引入推荐系统的经典模型——因子分解机中。因子分解机模型是一种混合推荐方法，能够将基于内容的特征和协作特征融合在一起，对特征进行组合，并且能有效处理稀疏数据；但特征分量的增加和特征组合会导致运算量激增，基于随机子空间的集成学习方法能够有效提升推荐计算效率，同时多模型运算能确保进一步提升预测精度。

但本文也存在一定的局限性。例如，在负采样过程中采用了随机抽样的方法，没有考虑抽取样本与正样本的语义相似性。未来计划引入基于语义计算的负采样策略，提高推荐模型的学习效力。还可以探究算法在模型输入和处理等方面的改进。例如在输入方面，可以引入社会、信任信息，增强属性。在处理方面，可以深化原本二阶特征交叉的处理方式，或者设计其他的特征交互方式。待 EFM 得到进一步优化后，继续探究模型在其他推荐任务上的表现，对其泛化效果进行分析。

参考文献：

[1] Kong X J, Shi Y J, Yu S, et al. Academic Social Networks: Modeling, Analysis, Mining and Applications[J]. Journal of Network and Computer Applications, 2019, 132: 86-103.

[2] Nascimento C, Laender A H F, da Silva A S, et al. A Source Independent Framework for Research Paper Recommendation [C]//Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries. New York: ACM, 2011: 297-306.

[3] 杨辰, 刘婷婷, 刘雷, 等. 融合语义和社交特征的电子文献资源推荐方法研究[J]. 情报学报, 2019, 38(6): 632-640.(Yang Chen, Liu Tingting, Liu Lei, et al. A Novel Recommendation Approach of Electronic Literature Resources Combining Semantic and Social Features[J]. Journal of the China Society for Scientific and Technical Information, 2019, 38(6): 632-640.)

[4] Champiri Z D, Asemi A, Binti S S S. Meta-Analysis of Evaluation Methods and Metrics Used in Context-Aware Scholarly Recommender Systems[J]. Knowledge and Information Systems, 2019, 61(2): 1147-1178.

[5] Bhagavatula C, Feldman S, Power R, et al. Content-Based Citation Recommendation[OL]. arXiv Preprint, arXiv: 1802.08301.

[6] Basu C, Hirsh H, Cohen W W, et al. Technical Paper Recommendation: A Study in Combining Multiple Information Sources[J]. Journal of Artificial Intelligence Research, 2001, 14: 231-252.

[7] Caragea C, Bulgarov F A, Godea A, et al. Citation-Enhanced Keyphrase Extraction from Research Papers: A Supervised Approach[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: Association for Computational Linguistics, 2014: 1435-1446.

[8] McNee S M, Albert I, Cosley D, et al. On the Recommending of Citations for Research Papers[C]//Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work. New York: ACM, 2002: 116-125.

[9] 毕强, 刘健. 基于领域本体的数字文献资源聚合及服务推荐方法研究[J]. 情报学报, 2017, 36(5): 452-460.(Bi Qiang, Liu Jian. Study on the Method of Aggregation and Service Recommendation of Digital Resource Based on Domain Ontology[J]. Journal of the China Society for Scientific and Technical Information, 2017, 36(5): 452-460.)

[10] 李亚梅, 秦春秀, 马续补. 基于科研人员情境化主题偏好的科技文献协同推荐研究[J]. 情报理论与实践, 2021, 44(12): 180-189. (Li Yamei, Qin Chunxiu, Ma Xubu. Research on Collaborative Recommendation of Scientific and Technological Literature Based on Researchers' Contextual Topic Preference [J]. Information Studies: Theory & Application, 2021, 44(12):

180-189.)

[11] Beel J, Gipp B, Langer S, et al. Research-Paper Recommender Systems: A Literature Survey[J]. International Journal on Digital Libraries, 2016, 17(4): 305-338.

[12] 汤志康, 李春英, 汤庸, 等. 学术社交平台论文推荐方法[J]. 计算机与数字工程, 2017, 45(2): 221-225. (Tang Zhikang, Li Chunying, Tang Yong, et al. Paper Recommendation Method Based on Scholar Social Platform[J]. Computer & Digital Engineering, 2017, 45(2): 221-225.)

[13] 刘健, 毕强, 刘庆旭, 等. 数字文献资源内容服务推荐研究——基于本体规则推理和语义相似度计算[J]. 现代图书情报技术, 2016(9): 70-77. (Liu Jian, Bi Qiang, Liu Qingxu, et al. New Content Recommendation Service of Digital Literature[J]. New Technology of Library and Information Service, 2016(9): 70-77.)

[14] 陈海华, 孟睿, 陆伟. 学术文献引文推荐研究进展[J]. 图书情报工作, 2015, 59(15): 133-143.(Chen Haihua, Meng Rui, Lu Wei. Research Review on Citation Recommendation of Academic Literatures[J]. Library and Information Service, 2015, 59(15): 133-143.)

[15] 刘扬. 基于质量的学术文献混合推荐模型研究[J]. 情报理论与实践, 2015, 38(2): 17-22. (Liu Yang. Research on the Hybrid Recommendation Model of Academic Reference Based on Quality[J]. Information Studies: Theory & Application, 2015, 38 (2): 17-22.)

[16] Haruna K, Ismail M A, Damiasih D, et al. A Collaborative Approach for Research Paper Recommender System[J]. PLoS One, 2017, 12(10): e0184516.

[17] Guo Q Y, Zhuang F Z, Qin C, et al. A Survey on Knowledge Graph-Based Recommender Systems[J]. IEEE Transactions on Knowledge and Data Engineering, 2022, 34(8): 3549-3568.

[18] Kanakia A, Shen Z H, Eide D, et al. A Scalable Hybrid Research Paper Recommender System for Microsoft Academic[C]// Proceedings of the 2019 World Wide Web Conference. New York: ACM, 2019: 2893-2899.

[19] 王勤洁, 秦春秀, 马续补, 等. 基于作者偏好和异构信息网络的科技文献推荐方法研究[J]. 数据分析与知识发现, 2021, 5(8): 54-64.(Wang Qinjie, Qin Chunxiu, Ma Xubu, et al. Research on Recommendation Method of Scientific and Technological Literature Based on Author Preference and Heterogeneous Information Network[J]. Data Analysis and Knowledge Discovery, 2021, 5(8): 54-64.)

[20] Ricci F, Rokach L, Shapira B. Introduction to Recommender Systems Handbook[A]//Ricci F, Rokach L, Shapira B, et al. Recommender Systems Handbook[M]. Boston, MA: Springer, 2011: 1-35.

[21] 蔡毅, 朱秀芳, 孙章丽, 等. 半监督集成学习综述[J]. 计算机科学, 2017, 44(S1): 7-13.(Cai Yi, Zhu Xiufang, Sun Zhangli, et al. Semi-Supervised and Ensemble Learning: A Review[J]. Computer Science, 2017, 44(S1): 7-13.)

[22] Sagi O, Rokach L. Ensemble Learning: A Survey[J]. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2018, 8(4): e1249.

[23] Breiman L. Bagging Predictors[J]. Machine Learning, 1996, 24 (2): 123-140.

[24] Freund Y, Schapire R E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting[J]. Journal of Computer and System Sciences, 1997, 55(1): 119-139.

[25] Ho T K. The Random Subspace Method for Constructing Decision Forests[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20(8): 832-844.

[26] Rokach L. Ensemble-Based Classifiers[J]. Artificial Intelligence Review, 2010, 33(1): 1-39.

[27] Kuncheva L I, Whitaker C J. Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy [J]. Machine Learning, 2003, 51(2): 181-207.

[28] Rendle S. Factorization Machines[C]//Proceedings of the 2010 IEEE International Conference on Data Mining. IEEE, 2011: 995-1000.

[29] Gu J X, Wang Z H, Kuen J, et al. Recent Advances in Convolutional Neural Networks[J]. Pattern Recognition, 2018, 77: 354-377.

[30] Cui P, Wang X, Pei J, et al. A Survey on Network Embedding[J]. IEEE Transactions on Knowledge and Data Engineering, 2019, 31 (5): 833-852.

[31] Choong A C H, Lee N K. Evaluation of Convolutionary Neural Networks Modeling of DNA Sequences Using Ordinal Versus One-Hot Encoding Method[C]//Proceedings of the 2017 International Conference on Computer and Drone Applications. IEEE, 2018: 60-65.

[32] 周飞燕, 金林鹏, 董军. 卷积神经网络研究综述[J]. 计算机学报, 2017, 40(6): 1229-1251. (Zhou Feiyan, Jin Linpeng, Dong Jun. Review of Convolutional Neural Network[J]. Chinese Journal of Computers, 2017, 40(6): 1229-1251.)

[33] Perozzi B, Al-Rfou R, Skiena S. DeepWalk: Online Learning of Social Representations[C]//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2014: 701-710.

[34] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[OL]. arXiv Preprint, arXiv: 1301.3781.

[35] Kingma D P, Ba J. A Method for Stochastic Optimization[C]// Proceedings of the 3rd International Conference on Learning Representations. 2015.

[36] Wang H, Chen B Y, Li W J. Collaborative Topic Regression with Social Regularization for Tag Recommendation[C]//Proceedings of the 23rd International Joint Conference on Artificial

Intelligence. 2013.
[37] Koren Y, Bell R, Volinsky C. Matrix Factorization Techniques for Recommender Systems[J]. Computer, 2009, 42(8): 30-37.
[38] Goldberg D, Nichols D, Oki B M, et al. Using Collaborative Filtering to Weave an Information Tapestry[J]. Communications of the ACM, 1992, 35(12): 61-70.

作者贡献声明：

杨辰：提出研究思路，设计研究方案，论文最终版本修订；
郑若桢：设计算法，进行实验，论文撰写和修改；
王楚涵：清洗和分析数据，论文撰写；
耿爽：完善研究思路，设计研究方案，论文修改；
王楠：论文修改。

利益冲突声明：

所有作者声明不存在利益冲突关系。

支撑数据

[1] 郑若桢 . 基于论文引用关系构造的点边数据 . DOI: 10.57760/sciencedb.j00133.00156.
[2] Hao Wang. 论文内容属性数据 . https://github. com/js05212/citeulike-a/blob/master/raw-data.csv.
[3] Hao Wang. 论文引用数据 .https://github.com/js05212/citeulike-a/blob/master/citations.dat.
[4] Hao Wang. 用户 - 论文交互数据 . https://github. com/js05212/citeulike-a/blob/master/users.dat.

收稿日期：2022-07-25
收修改稿日期：2022-10-16

Ensemble Factorization Machine and Its Application in Paper Recommendation

Yang Chen　Zheng Ruozhen　Wang Chuhan　Geng Shuang　Wang Nan
(College of Management, Shenzhen University, Shenzhen 518060, China)

Abstract: [**Objective**] This study proposes an improved paper recommendation framework based on Ensemble Learning and Factorization Machine. It addresses the issues of the existing methods, such as difficulties in processing sparse data and representing features. [**Methods**] First, we used Convolutional Neural Network, Network Embedding, and other algorithms to obtain feature representations, which were processed by Factorization Machine learners. Homogeneous weak Factorization Machine learners are then trained based on Ensemble Learning. We integrated these weak learners into a stronger learner through the voting mechanism and generated the final recommendations. [**Results**] We examined the new model with the CiteULike dataset, and the Precision, Accuracy, and F-Measure reached 72.6%, 69.7%, and 76.2%, respectively, 20%, 15% , and 9% higher than the benchmark algorithms. [**Limitations**] The input, sampling strategy, and processing mode need to be further explored. [**Conclusions**] The proposed Ensemble Factorization Machine enables effective representation and utilization of sparse data features, enhancing the recommendation performance.

Keywords: Research Paper Recommendation　Factorization Machine　Ensemble Learning