文章编号：2096-1472(2023)-06-30-05
DOI:10.19644/j.cnki.issn2096-1472.2023.006.007

融合标签文本的 k-means 聚类和矩阵分解算法

居晓媛，汪明艳

(上海工程技术大学管理学院，上海 201620)
✉ ju_xiaoyuan_123@163.com; wmy61610@126.com

摘 要：针对推荐系统中依赖用户对项目的评分信息带来的稀疏性问题，提出一种融合标签文本的 k-means 聚类和矩阵分解的推荐算法。该模型首先对项目信息构建项目特征画像，利用 k-means 聚类提取项目的潜在特征数量，然后利用隐语义模型 LFM 进行矩阵分解，将用户-评分矩阵进行分解重构得到预测评级，并根据排序推荐。将算法在 MovieLens 数据集上进行实验，结果表明该推荐算法的均方根误差(*RMSE*)和绝对平均误差(*MAE*)表现较好，在 ml-latest-small 数据集中的准确率(*precision*)和召回率(*recall*)较次优算法分别提升了 14.5%和 20.7%。通过将 k-means 聚类应用到用户的潜在兴趣和项目的潜在特征提取中，提升了推荐算法的有效性。

关键词：推荐算法；矩阵分解；k-means；LFM
中图分类号：TP391 **文献标识码：A**

An Algorithm of Combining K-means Clustering and Matrix Decomposition of Label Text

JU Xiaoyuan, WANG Mingyan

(*School of Management, Shanghai University of Engineering Science, Shanghai* 201620, *China*)
✉ ju_xiaoyuan_123@163.com; wmy61610@126.com

Abstract: Aiming at the sparsity problem caused by the reliance on user rating information for projects in recommendation systems, this paper proposes a recommendation algorithm combining k-means clustering and matrix decomposition of label text. In this model, a project feature portrait is firstly constructed based on project information, and the number of potential features of the project are extracted by using k-means clustering. Then, Latent Factor Model (LFM) is used for matrix decomposition. The user-rating matrix is decomposed and reconstructed to obtain a predictive rating, and recommendations are made based on sorting. The algorithm has been tested on the MovieLens dataset, and the results shows that the proposed recommended algorithm performs well in root mean square error (RMSE) and absolute mean error (MAE). The accuracy and recall rates in the ml-latest-small dataset are improved by 14.5% and 20.7%, respectively. K-means clustering is applied to the extraction of users′ potential interests and items′ potential features, which proves the effectiveness of the recommendation algorithm.

Keywords: recommendation algorithm; matrix decomposition; k-means; LFM

1 引言(Introduction)

推荐算法是以大数据为依托，以用户行为对象，分析用户浏览的历史记录、习惯、偏好，有针对性地为用户推荐符合其喜好的内容。常见的推荐算法包括协同过滤[1]、基于内容的推荐[2]及混合推荐算法等。协同过滤主要包括两种：基于内存的协同过滤(Memory-based CF)和基于模型的协同过滤(Model-based CF)。基于模型的协同过滤通过建模的方式模拟用户对项目的评分行为，其使用机器学习与数据挖掘技术，从训练数据中确定模型并将模型用于预测未知商品评分。作为常见的基于模型的协同过滤方式，矩阵分解已成为降低数据维数、提取数据潜在特征和减少稀疏性的有效方法。根据矩阵分解可以解决数据稀疏的特点，所以矩阵分解被广泛

应用于推荐系统中。常用的矩阵分解技术包括奇异值分解(SVD)[3]、主成分分析(PCA)[4]、概率矩阵分解(PMF)[5]和非负矩阵分解(NMF)[6]。YAO 等提出了一种基于联合概率矩阵分解的群体推荐方法，能更好地模拟群体推荐问题，在面向群体的推荐问题中取得较好的效果[7]。杨辰等提出一种基于细粒度属性偏好聚类的新型推荐模型，对项目-属性关系和用户-属性偏好进行建模，基于用户簇或项目簇采用协同过滤算法生成推荐列表[8]。YUAN 等认为隐语义模型(Latent Factor Model)属于 SVD 的一种变体，它降低了用户项目评级矩阵的维数，表示评级矩阵中用户和项目的潜在特征，其提出了一种混合方面级的隐语义模型(HALFM)优化了全局方面级潜在因子和本地方面级潜在因子，提高了模型预测效果[9]。李淑芝提出一种评论文本和评分矩阵交互(RTRM)的深度模型，取得了较好的推荐效果[10]。邢长征等在推荐模型中加入用户和项目标签信息，通过标签使用次数反映用户喜好和项目特征，结果表明模型能有效提高跨域推荐的准确性[11]。

尽管学者们从不同的角度改进了矩阵分解算法，在一定程度上提高了算法的准确率和召回率，也有学者关注到项目本身的属性信息，但是忽略了将用户的评论标签与矩阵分解结合使用。本文将矩阵分解算法和 k-means 聚类算法相结合，首先基于用户评论的标签文本构建项目特征画像和用户兴趣画像，利用 k-means 算法将其聚类，找出最优聚类数，然后利用隐语义模型进行矩阵分解，将用户-评分矩阵进行分解重构并做出推荐，最后在 MovieLens 数据集上进行实验，验证该算法的综合表现。

2　算法设计(Algorithm design)

针对推荐系统中依赖用户对项目评分信息的数据稀疏性问题，本文提出一种融合标签文本的 k-means 聚类和矩阵分解的推荐算法。矩阵分解的痛点在于隐含特征值的确定，本文将 k-means 聚类算法引入矩阵分解，确定隐含特征值。主要的算法流程包括以下两个模块：一是 k-means 聚类模块。在项目潜在特征提取方面，通过对电影标签文本进行量化处理，采用 TF-IDF(Term Frequency-inverse Document Frequency)统计方法构建电影特征画像，利用 k-means 聚类方法获取电影的簇，即项目的潜在特征；在用户潜在兴趣提取方面，将用户对电影的点评标签作为用户兴趣标签，构建用户的兴趣画像，利用 k-means 聚类方法获取用户的潜在兴趣。二是矩阵分解模块。根据找到的项目潜在特征或者用户潜在兴趣将用户-评分矩阵进行分解，主要算法流程如图 1 所示。

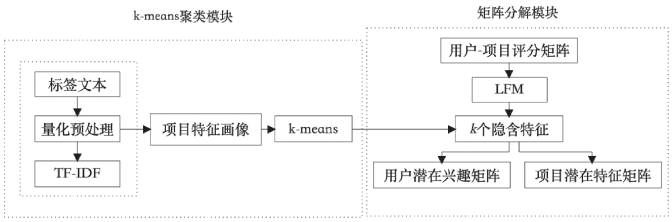


图 1　算法流程图

Fig. 1 Algorithm flow chart

2.1　基于 k-means 的项目潜在特征提取

2.1.1 基于 TF-IDF 的项目标签文本量化

TF-IDF 是一种针对关键词的统计分析方法，用于评估一个词对一个文件集或者一个语料库的重要程度。一个词的重要程度跟它在文章中出现的次数成正比，跟它在语料库出现的次数成反比。

首先构建项目特征画像和用户兴趣画像，为将项目的文本标签转化为计算机可以识别处理的数据结构，然后利用 TF-IDF 分析提取 TOP-n 个关键词。TF-IDF 计算公式如下：

$$TF_{ij}=\frac{f_{ij}}{f_{dj}} \tag{1}$$

$$IDF_i=\ln\frac{N}{N_i} \tag{2}$$

$$w_{ij}=TF_{ij}\times IDF_i=\frac{f_{ij}}{f_{dj}}\times\ln\frac{N}{N_i} \tag{3}$$

其中，N 表示总的文档数，N_i 表示包含关键词的文档数，f_{ij} 表示关键词在文档中出现的次数，f_{dj} 表示文档中的词语总数。通过计算每个项目所有标签词的 TF-IDF 值，进行对比，取 TF-IDF 值中最大的 n 个关键词作为目标文档的特征向量，组成项目的特征画像。

2.1.2 项目标签内容的 k-means 聚类

根据项目特征画像，将属性特征相似的项目聚在同一簇内，k-means 聚类首先选取任意样本点作为聚类初始中心点，对于每个样本点，通过计算欧式距离计算该样本点到每个聚类初始中心点的距离，将其划分至距离最近的簇中。计算样本点到中心点的距离公式如下：

$$d(m_i,c_i)=\sqrt{\sum_{j=1}^{n}(m_{ij}-c_{ij})^2} \tag{4}$$

其中，$d(m_i,c_i)$ 表示样本点 m_i 到聚类中心 c_i 的距离，n 表示簇内项目个数。计算每个类别中数据点的平均值，并将得到的平均值作为新的聚类中心，再利用上述公式计算样本点到新的聚类中心的距离，重新划分项目簇；经过若干次迭代之后，如果聚类中心不再变化或者变化很小，则可确最终的聚类簇。

对于聚类效果的评估，选择轮廓系数，其公式如下：

$$SC_i=\frac{b_i-a_i}{\max(b_i,a_i)} \tag{5}$$

其中，a_i 表示样本点 i 与同一簇中其他点的平均距离，即样本点 i 与同一簇中其他点的相似度；b_i 表示样本点 i 到其他簇中所有点的平均距离，即轮廓系数衡量的是内部距离最小化，外部距离最大化。

利用 TF-IDF 进行关键词量化后，选择 TF-TDF 值最大的 30 个关键词，聚类区间选择 2—50，观察聚类效果，如图 2 所示，当聚类数为 25—40 时，聚类效果较好，为检验聚类数为 25—40 的模型评估效果，下面的实验中将 k 值取 25—40，观察不同 k 值的效果。

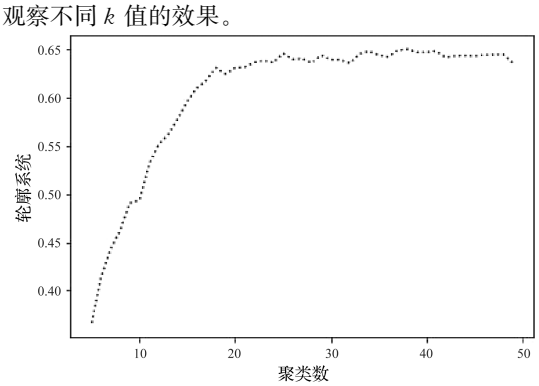


图 2　轮廓系数图

Fig. 2 Diagram of contour coefficient

2.2 基于 LFM 项目潜在特征矩阵生成

2.2.1 矩阵分解

矩阵分解算法是将用户-项目评分矩阵进行分解，为每一个用户和项目生成一个隐向量，将用户和项目定位到隐向量表示的空间上，如图 3 所示。将($m\times n$)维的共现矩阵 $\boldsymbol{R}$ 分解为($m\times k$)维的用户矩阵 $\boldsymbol{P}$ 和($k\times n$)维的项目矩阵 $\boldsymbol{Q}$ 相乘的形式，使得 $\boldsymbol{R}_{m\times n}=\boldsymbol{P}_{m\times k}\times\boldsymbol{Q}_{k\times n}$。其中，$m$ 是用户数量，n 是项目数量，k 是隐向量的维度。k 的大小决定了隐向量表达能力的强弱。k 的取值越小，隐向量包含的信息越少，模型的泛化程度越高；反之，k 的取值越大，隐向量的表达能力越强，但泛化程度相应降低。因此，最终目标是求出用户矩阵 $\boldsymbol{P}$ 和项目矩阵 $\boldsymbol{Q}$ 中的每一个值，然后对用户-项目评分矩阵进行预测。

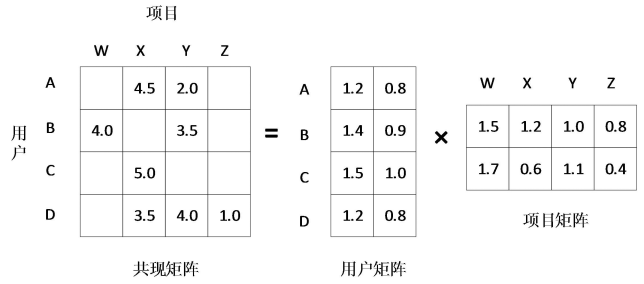


图 3 矩阵分解过程图

Fig. 3 Diagram of matrix decomposition process

基于用户矩阵 $\boldsymbol{P}$ 和项目矩阵 $\boldsymbol{Q}$，用户 u 对项目 i 的预估评分如下：

$$\hat{r}_{ui}=\boldsymbol{P}_u\boldsymbol{Q}_i=\sum_{j=1}^{k}\boldsymbol{P}_{uj}\boldsymbol{Q}_{ji} \tag{6}$$

其中，$\boldsymbol{P}_{uj}$ 是用户 u 在用户矩阵 $\boldsymbol{P}$ 中的对应行向量的第 j 维，$\boldsymbol{Q}_{ji}$ 是项目 i 在项目矩阵 $\boldsymbol{Q}$ 中的对应列向量的第 j 维，k 表示隐向量维度。

2.2.2 随机梯度下降

矩阵分解的标准是减少预测评分 $\hat{r}_{ui}$ 与真实评分 r_{ui} 之间的误差，利用平方差构建损失函数：

$$Cost=\sum_{\langle u,i\rangle\in\mathbb{R}}(r_{ui}-\hat{r}_{ui})^2+\lambda\sum_{j=1}^{k}(\|\boldsymbol{P}_u\|^2+\|\boldsymbol{Q}_i\|^2) \tag{7}$$

损失函数是指用户 u 对项目 i 的真实喜爱程度与预测出来喜爱程度的误差，要是模型合理就是误差达到最小值。公式(7)中的 $\lambda\sum_{j=1}^{k}(\|\boldsymbol{P}_u\|^2+\|\boldsymbol{Q}_i\|^2)$ 是正则项，用来防止训练结果出现过拟合。

损失函数的优化使用随机梯度下降算法，梯度下降的递推公式如下：

$$\boldsymbol{P}'_{uj}=\boldsymbol{P}_{uj}+\alpha\left[(r_{ui}-\sum_{j=1}^{k}\boldsymbol{P}_{uj}\boldsymbol{Q}_{ji})\boldsymbol{Q}_{ji}-\lambda\boldsymbol{P}_{uj}\right] \tag{8}$$

$$\boldsymbol{Q}'_{ji}=\boldsymbol{Q}_{ji}+\alpha\left[(r_{ui}-\sum_{j=1}^{k}\boldsymbol{P}_{uj}\boldsymbol{Q}_{ji})\boldsymbol{P}_{uj}-\lambda\boldsymbol{Q}_{ji}\right] \tag{9}$$

其中，$\boldsymbol{P}'_{uj}$ 和 $\boldsymbol{Q}'_{ji}$ 分别表示经过一次梯度下降后的 $\boldsymbol{P}_{uj}$ 和 $\boldsymbol{Q}_{ji}$ 值，α 是在梯度下降过程中的步长，也称作学习速率，表示每次更新的快慢。在用户-项目评分矩阵 $\boldsymbol{R}$ 中，通过迭代可以求得每个用户对每个项目的喜爱程度，选取喜爱程度最高且用户没有反馈过的项目进行推荐。

3 实验设计(Experimental design)

3.1 实验数据集描述

本文是以电影推荐为具体场景，构建基于电影潜在特征和用户潜在兴趣的推荐模型。本文采用推荐算法中常用的电影数据集 MovieLens 中的 ml-latest-small 数据集，包括用户-电影评分数据集 ratings. csv、电影信息数据集 movies. csv 及电影标签数据集 tags. csv。k-means 聚类模块使用到电影信息数据集 movies，格式包括电影 ID、标题、类型；电影标签数据 tags. csv 包括用户 ID、电影 ID、用户给电影打的标签、时间戳。矩阵分解模块应用到用户-电影评分数据集 ratings。数据集描述如表 1 所示。

表1 数据集描述

Tab.1 Dataset description

电影数/部	用户数/名	评分数/个	稀疏度/%
9 742	610	100 836	98.3

3.2 评估指标

3.2.1 均方根误差(*RMSE*)和绝对平均误差(*MAE*)

RMSE 是精确度的度量，用于比较特定数据集的不同模型的预测误差，*RMSE* 值越小，说明模型具有更好的精确度。

$$RMSE=\sqrt{\frac{1}{|T|}\sum_{\langle u,i\rangle\in T}(\hat{r}_{ui}-r_{ui})^2} \tag{10}$$

$$MAE=\frac{1}{|T|}\sum_{\langle u,i\rangle\in T}|\hat{r}_{ui}-r_{ui}| \tag{11}$$

其中，T 是测试集样本数，$\hat{r}_{ui}$是模型预测评分，r_{ui}是测试集的用户真实评分。

3.2.2 准确率(*Precision*)和召回率(*Recall*)

准确率(*Precision*)和召回率(*Recall*)的公式如下：

$$Precision=\frac{TP}{TP+FP} \tag{12}$$

$$Recall=\frac{TP}{TP+FN} \tag{13}$$

其中，*TP*(True Positive)表示将正类预测为正类；*FP*(False Positive)表示将负类预测为正类；*FN*(False Negative)表示将正类预测为负类。

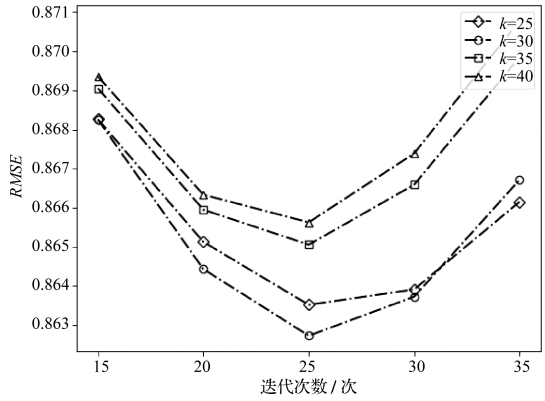
3.3 参数分析

为了得到本文算法的最优参数，首先根据 k-means 的评估指标轮廓系数选定效果较好的 k 值为 25—40，其余参数的确定进行控制变量实验，测试重要参数对本文算法效果的影响。实验随机选取用户-评分数据集的 80%作为训练集，剩余 20%作为测试集，为防止每次运行结果不一致，设置随机数种子 *random _ state*=1，具体实验过程如下。

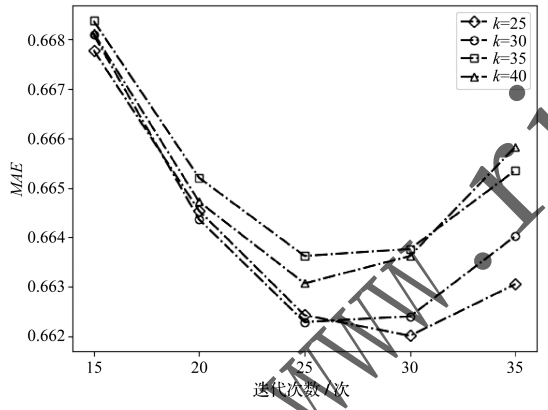
3.3.1 不同 *k* 值下模型表现

固定学习率是 0. 005，正则项系数是 0. 02，设定迭代次数为 15—35 次，比较不同 k 值(取值区间为 25—40)的效果。图 4 是 ml-latest-small 数据集迭代次数在[15，35]范围内，k

取不同值对模型效果的影响。由图 4(a)可以看出，不同的 k 值，迭代次数在[15,25]区间，*RMSE* 值呈下降趋势，模型效果变好；在[25,35]区间，*RMSE* 值呈上升趋势，模型效果下降；当 k 取值为 30 时，算法的推荐效果最佳。由图 4(b)可以看出，迭代次数在[15,25]区间，*MAE* 值呈下降趋势，模型效果变好；在[25,35]区间，*MAE* 值呈上升趋势；k 取值为 25 时，在迭代 15 次、30 次、35 次时效果优于 k 取值为 30，但总体而言，k 取值为 30 效果更优，并且在迭代 25 次时，模型效果最优。因此，根据 k-means 聚类得出 30 个聚类时效果最优是成立的，并且此时的最优迭代次数为 25 次。



(a)不同 k 值下的 *RMSE*



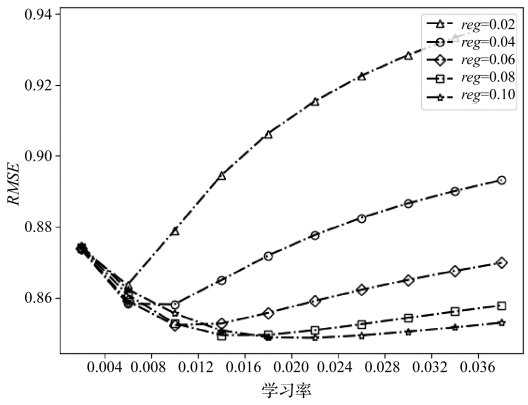
(b)不同 k 值下的 *MAE*

图 4　不同 k 值下的模型表现

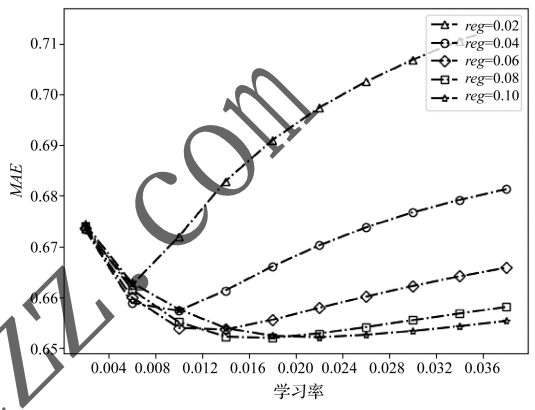
Fig. 4 Model performance under different k values

3.3.2 不同正则项系数下模型表现

固定迭代次数为 25 次，k 取值为 30 时，比较不同学习率及不同正则项系数对模型的影响。图 5 是学习率取值在[0.002,0.04]区间，正则项系数的取值不同对模型效果的影响。由图 5(a)可以看出，正则项系数取 0.02、学习率在 0.005 处时，模型效果最优，当学习率大于 0.005 时，*RMSE* 值呈上升趋势；正则项系数取 0.04 或 0.06、学习率在 0.01 处时，模型效果达到最优；正则项系数取 0.08 或 0.1、学习率在 0.022 处时，与其他正则项系数取值相比，模型效果最好。同样，由图 5(b)可以看出，正则项系数取 0.08 或 0.1 时，模型效果较好。综合考虑认为，正则项系数取 0.1、学习率取 0.022 时，模型效果最优。



(a)不同正则项系数下的 *RMSE*



(b)不同正则项系数下的 *MAE*

图 5　不同正则项系数下的模型表现

Fig. 5 Model performance under different regularization coefficient

4　实验结果与分析(Experimental results and analysis)

4.1　比较方法

为了验证本文提出算法的有效性，选取 3 种推荐算法模型与本文提出算法进行比较。①非负矩阵分解(NMF)基于传统的矩阵分解模型，是一种有效的数据分解方法，将用户-评分矩阵分解为 2 个非负的小矩阵[12]。②SVD++，在矩阵分解的基础上考虑用户浏览的历史记录，认为用户浏览的历史记录对当前项目的评分有一定的影响，将项目间的关联考虑到模型的评估中[13]。③基于内容的推荐(Baseline)是根据项目本身的标签信息构建项目特征向量，根据项目间的相似性为用户做推荐[14]。

在属性信息层面，非负矩阵分解(NMF)和 SVD++两种方法均属于矩阵分解，并且没有考虑项目本身的属性信息，可以在融合标签文本的属性信息层面对本文算法进行有效性验证。在聚类方法层面，基于内容的推荐仅考虑了项目本身的标签属性信息而未引入聚类算法，可以对本文算法引入聚类信息进行有效性验证。

4.2　比较结果

4.2.1 不同模型的 *RMSE* 和 *MAE* 对比

对比实验选择 ml-latest-small 数据集，根据控制变量得

出的参数结果，选取潜在特征 $k=30$，迭代次数为 25，学习率为 0.022，正则项系数为 0.1。实验结果如表 2 所示。非负矩阵分解 NMF 和 SVD++都属于矩阵分解，但是没有考虑项目本身的属性信息；而基与内容的推荐(Baseline)仅考虑项目本身的标签信息，根据项目间的相似度推荐，本文模型 KLFM 采用聚类方法，更好地将属性相似的项目聚类，从而形成项目特征画像和用户兴趣画像，能够更好地为用户做出推荐。由表 2 可知，本文模型 KLFM 的 *RMSE* 和 *MAE* 均表现较好。

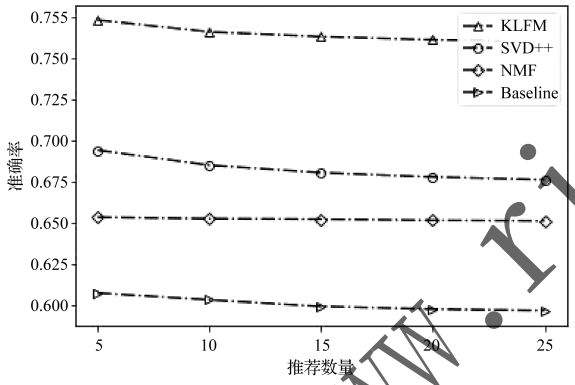
表2 算法性能对比

Tab.2 Algorithm performance comparison

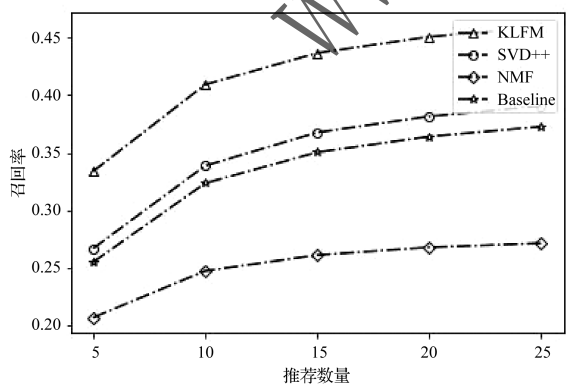
模型	*RMSE*	*MAE*
KLFM	0.847 7	0.651 5
SVD++	0.857 7	0.657 7
NMF	0.915 7	0.703 1
Baseline	0.986 5	0.784 4

4.2.2 不同模型的准确率和召回率对比

本实验采用准确率和召回率对各个推荐算法进行评估，具体结果如图 6 所示。横轴表示推荐数量，在 ml-latest-small 数据集上，本文模型 KLFM 在推荐数量为[5,25]区间上表现优秀，在推荐的准确率和召回率方面都有了较大幅度的提升，较次优算法分别提升了 14.5%和 20.7%，模型表现较优。



(a)准确率对比结果



(b)召回率对比结果

图 6 ml-latest-small 数据集上性能对比结果

Fig. 6 Algorithm performance comparison on ml-latest-small

5 结论(Conclusion)

本文针对推荐系统中用户对项目评分信息的数据稀疏性问题，提出一种融合标签文本的 k-means 聚类和矩阵分解的推荐算法。该模型首先对项目信息构建项目特征画像，利用 k-means 聚类找出项目的潜在特征数量；然后根据轮廓系数找出最优的 k 值，利用隐语义模型 LFM 进行矩阵分解，将用户-评分矩阵进行分解重构；最后通过实验分析不同参数对模型效果的影响，根据控制变量法找出最优的模型参数。将本文算法 KLFM 与其他推荐模型算法进行对照实验，得出结论：本文模型的 *RMSE* 和 *MAE* 表现较好，在 ml-latest-small 数据集中准确率和召回率上较次优算法分别提升了 14.5%和 20.7%，有效地改善了推荐算法的效果，可以推广到电商、新闻、社交媒体等其他推荐场景。但是，本文构建项目特征矩阵仅考虑了用户评论标签这一属性，后续研究可以考虑项目的其他属性，能够更好地表征项目潜在特征。

参考文献(References)

[1] SLAVAKIS K, GIANNAKIS G B, MATEOS G. Modeling and optimization for big data analytics: (statistical) learning tools for our era of data deluge[J]. IEEE Signal Processing Magazine, 2014, 31(5): 18-31.

[2] JAVED U, SHAUKAT K, HAMEED I A, et al. A review of content-based and context-based recommendation systems[J]. International Journal of Emerging Technologies in Learning, 2021, 16(3): 274-306.

[3] ZHOU X, HE J, HUANG G, et al. SVD-based incremental approaches for recommender systems[J]. Journal of Computer and System Sciences, 2015, 81(4):717-733.

[4] GOLDBERG K, ROEDER T, GUPTA D, et al. Eigenstate: a constant time collaborative filtering algorithm[J]. Information Retrieval, 2001, 4(2):133-151.

[5] LIU J, WU C, XIONG Y, et al. List-wise probabilistic matrix factorization for recommendation[J]. Information Sciences, 2014, 278: 434-447.

[6] BERRY M W, BROWNE M, LANGVILLE A N, et al. Algorithms and applications for approximate nonnegative matrix factorization[J]. Computational Statistics & Data Analysis, 2007, 52(1): 155-173.

[7] YAO Z, GAO K. User recommendation method based on joint probability matrix decomposition in CPS networks[J]. Computer Communications, 2020, 157:221-231.

[8] 杨辰,陈晓虹,王楚涵,等.基于用户细粒度属性偏好聚类的推荐策略[J].数据分析与知识发现,2021,5(10):94-102.

[9] YUAN H, CHEN Z, YANG J, et al. A hybrid aspect based latent factor model for recommendation[J]. Chinese Journal of Electronics, 2020, 29(3):482-490.

[10] 李淑芝,余乐陶,邓小鸿.融合深度情感分析和评分矩阵的推荐模型[J].电子与信息学报,2022,44(1):245-253.

[11] 邢长征,杨晓婷.基于 SVD++ 与标签的跨域推荐模型[J].计算机工程,2018,44(4):225-230.

[12] JIA Y, KWONG S, HOU J, et al. Semi-supervised non-negative matrix factorization with dissimilarity and similarity regularization[J]. IEEE Transactions on Neural Networks and Learning Systems, 2019, 31(7):2510-2521.

[13] SHI W, WANG L, QIN J. User embedding for rating prediction in SVD++ -based collaborative filtering[J]. Symmetry, 2020, 12(1):121.

[14] BOUGIATIOTIS K, GIANNAKOPOULOS T. Enhanced movie content similarity based on textual, auditory and visual information[J]. Expert Systems with Applications, 2018, 96:86-102.

作者简介：

居晓媛(1996-)，女，硕士生. 研究领域：数字营销，数据分析.

汪明艳(1975-)，女，博士，教授. 研究领域：数字营销，数据分析，舆情治理.

(上接第 29 页)

[21] YANG J, ZHANG Q, NI B, et al. Modeling point clouds with self-attention and gumbel subset sampling[C]// BRENDEL W. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019:3323-3332.

[22] LI Y, BU R, SUN M, et al. Pointcnn: Convolution on x-transformed points[J]. Advances in Neural Information Processing Systems, 2018, 31:820-830.

[23] WANG Y, SUN Y, LIU Z, et al. Dynamic graph cnn for learning on point clouds[J]. Acm Transactions on Graphics, 2019, 38(5):1-12.

[24] ZHAO H, JIANG L, FU C W, et al. Pointweb: Enhancing local neighborhood features for point cloud processing[C]// BRENDEL W. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019:5565-5573.

[25] WU W, QI Z, FUXIN L. Pointconv: Deep convolutional networks on 3d point clouds[C]// BRENDEL W. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019:9621-9630.

[26] LIU X, HAN Z, LIU Y S, et al. Point2sequence: Learning the shape representation of 3d point clouds with an attention-based sequence to sequence network[C]// STONE P. Proceedings of the AAAI Conference on Artificial Intelligence. Honolulu: AAAI Press, 2019, 33(1):8778-8785.

[27] THOMAS H, QI C R, DESCHAUD J E, et al. Kpconv: Flexible and deformable convolution for point clouds[C]// MORTENSEN E. Proceedings of the IEEE/CVF International Conference on Computer Vision. New York: IEEE, 2019:6411-6420.

[28] MA X, QIN C, YOU H, et al. Rethinking network design and local geometry in point cloud: A simple residual mlp framework [DB/OL]. (2022-02-15) [2022-07-12]. https://arxiv.53yu.com/abs/2202.07123.

[29] LI J, CHEN B M, LEE G H. So-net: Self-organizing network for point cloud analysis[C]// MORTENSEN E. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018:9397-9406.

[30] ZENG W, GEVERS T. 3Dcontextnet: K-d tree guided hierarchical learning of point clouds using local and global contextual cues[C]// MORTENSEN E. Proceedings of the European Conference on Computer Vision Workshops. Munich: Springer Verlag, 2018:19-41.

[31] 许嘉麟,姚双,张蕊华,等.基于亲疏度矩阵的点云置换不变特征提取方法[J].计算机工程,2022,48(6):243-250.

[32] TANG L, ZHAN Y, CHEN Z, et al. Contrastive boundary learning for point cloud segmentation[C]// DANA K. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022:8489-8499.

[33] QIAN G, LI Y, PENG H, et al. PointNeXt: Revisiting PointNet++ with improved training and scaling strategies[DB/OL]. (2022-06-09) [2022-07-12]. https://arxiv.53yu.com/abs/2206.04670.

作者简介：

张海博(1997-)，男，硕士生. 研究领域：人工智能，点云处理.

沈　洋(1975-)，男，博士，教授. 研究领域：深度学习，点云分析，图像处理.

许　浩(1988-)，男，博士，讲师. 研究领域：深度学习与优化方法，计算机图像处理.

包艳霞(1980-)，女，博士，讲师. 研究领域：优化算法理论，人工智能.

刘　江(1975-)，男，硕士，工程师. 研究领域：深度学习.