计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2019,28(7):162−168 [doi: 10.15888/j.cnki.csa.006981]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于有效特征子集提取的高效推荐算法①

于　旭[1,2], 王前龙[1], 徐凌伟[1], 田　甜[3], 徐其江[4], 崔焕庆[2]

[1](青岛科技大学 信息科学与技术学院, 青岛 266061)
[2](山东科技大学 山东省智慧矿山信息技术重点实验室, 青岛 266590)
[3](山东建筑大学, 济南 250101)
[4](山东信息职业技术学院 软件系, 潍坊 261061)
通讯作者: 于　旭, E-mail: yuxu0532@163.com

摘　要: 推荐系统是根据用户的历史信息对未知信息进行预测. 用户项目评分矩阵的稀疏性是目前推荐系统面临的主要瓶颈之一. 跨域推荐系统是解决数据稀疏性问题的一种有效方法. 本文提出了基于有效特征子集选取的高效推荐算法 (FSERA), FSERA 是提取辅助域的子集信息, 来扩展目标域数据, 从而对目标域进行协同过滤推荐. 本文采用 K-means 聚类算法将辅助域的数据进行提取来降低冗余和噪声, 获取了辅助域的有效子集, 不仅降低了算法复杂度, 而且扩展了目标域数据, 提高了推荐精度. 实验表明, 此方法比传统的方法有更高的推荐精度.

关键词: 跨领域; 特征选择; 聚类; 协同过滤

引用格式: 于旭,王前龙,徐凌伟,田甜,徐其江,崔焕庆.基于有效特征子集提取的高效推荐算法.计算机系统应用,2019,28(7):162–168. http://www.c-s-a.org.cn/1003-3254/6981.html

Efficient Recommendation Algorithm Based on Feature Subset Extraction

YU Xu[1,2], WANG Qian-Long[1], XU Ling-Wei[1], TIAN Tian[3], XU Qi-Jiang[4], CUI Huan-Qing[2]

[1](School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266061, China)
[2](Shandong Key Laboratory of Wisdom Mine Information Technology, Shandong University of Science and Technology, Qingdao 266590, China)
[3](Shandong Jianzhu University, Jinan 250101, China)
[4](Software Engineering Department, Shandong College of Information Technology, Weifang 261061, China)

Abstract: The recommendation system predicts the unknown information according to the user's historical information. Sparsity of user item scoring matrix is one of the main bottlenecks faced by recommendation system. Cross-domain recommendation system is an effective method to solve the problem of data sparsity. In this study, an Efficient Recommendation Algorithm based on effective Feature Subset selection (FSERA) is proposed. FSERA extracts subset information of auxiliary domain to expand target domain data, so as to collaboratively filter recommendation for target domain. In this study, K-means clustering algorithm is used to extract data from the auxiliary domain to reduce redundancy and noise, and to obtain an effective subset of the auxiliary domain, which not only reduces the complexity of the algorithm, but also expands the target domain data and improves the recommendation accuracy. Experiments show that this method has higher recommendation accuracy than traditional methods.

Key words: cross domain; feature selection; clustering; collaborative filtering

① 基金项目: 国家自然科学基金 (61402246, 61503220); 山东省自然科学基金 (ZR2019MF014, ZR2017BF023); 光电技术与智能控制教育部重点实验室 (兰州交通大学) 开放课题基金 (KFKT2018-2)

Foundation item: National Natural Science Foundation of China (61402246, 61503220); Natural Science Foundation of Shandong Province (ZR2019MF014, ZR2017BF023); Open Fund of Key Laboratory of Opto-Technology and Intelligent Control (Ministry of Education), Lanzhou JiaotongUniversity (KFKT2018-2)

收稿时间: 2019-01-16; 修改时间: 2019-02-03; 采用时间: 2019-02-18; csa 在线出版时间: 2019-07-01

随着互联网的快速发展, 人们正处于一个信息爆炸性增长的时代. 人们从海量信息中获取对自己有价值的信息越来越困难. 推荐系统[1,2]正是当前解决这一问题最有效的技术之一. 当前在电子商务[3]、餐饮[4]、交通运输[5]等领域都存在着各种形式的推荐系统, 但人们对当前推荐系统的效果还不能完全满意, 而且大部分推荐系统仅服务于单一领域. 协同过滤 (collaborative filtering) 算法[6]是推荐系统中使用最广泛、最成功的方法之一, 它可以归结为分析表格数据, 即用户项目评分矩阵.

然而, 在现实生活中, 人们往往很少对项目进行评分, 这导致了用户项目评分矩阵非常稀疏. 稀疏化情况的加剧, 使得近邻寻找的复杂度急剧增加, 推荐系统的性能显著下降, 稀疏性已经成为大多数单领域域协同过滤算法的一个瓶颈[7]. 为了缓解这一问题, 最近人们提出了一些跨域推荐方法. 跨域推荐是将多个领域的数据联合起来, 共同作用于目标域的推荐系统. 这些方法有效的缓解了目标域的稀疏性问题. 目前对此方面的研究主要有: Berkovsky 等人[8]提到一个基于邻居的 CDCF(N-CDCF), 可以被视为基于内存的方法的跨领域扩展, 即 N-CF. Hu 等[9]提到基于矩阵分解的 CDCF(MF-CDCF), 其可以被视为矩阵分解方法的跨领域版本. Singh 和 Gordon[10]提出了一种集体矩阵分解 (CMF) 模型. CMF 将用户维度上所有域的评估矩阵相结合, 以便通过普通用户因子矩阵传递知识. Li 等人[11]提出了一种用于推荐系统的基于码本的知识转移 (CBT). 他们首先将辅助评级矩阵中的评级压缩为被称为码本的信息丰富且紧凑的集群级评分模式表示. 香港科技大学潘微科等人提出了一种对二元信息矩阵 (喜欢与不喜欢, 购买与不购买等) 和评分矩阵进行联合分解的方案[12], 也有效地降低了目标领域数据稀疏所带来了的问题, 但是该模型要求两个矩阵中的用户、项目必须严格一致. 文献[13]则同时对两个领域中的二元信息矩阵和用户评论信息进行联合简历模型, 得到用户的特征向量; 并训练得到两个非线性的映射函数, 一个用于将源领域中的用户偏好信息映射到目标领域中, 另一个则用于将源领域中的用户兴趣转换为目标领域中用户的兴趣. 然而, 他们仅仅是对特征进行随机或者是简单的选取, 并没有对选取的特征进行有效的评判.

但由于每个领域都有大量的评分数据, 当我们将这些领域联合进行处理的时候, 会有更高维度的特征空间, 这不仅产生高额的计算开销, 甚至可能会导致推荐系统崩溃. 因此进行合理有效的特征子集选取成为跨域推荐系统必须要解决的问题.

在本文的研究中, 我们假设两个具有强相关性的领域, 比如图书和电影, 他们具有相似的体裁和特性, 所以我偶们认为图书和电影是具有高相关性的领域. 本文我们采用在不同领域上共享用户的亚马逊数据集, 基于特征之间的相关性分析, 对辅助域特征以无监督聚类的方式进行特征子集选取, 删除掉相关性较大和冗余的特征, 筛选出对目标域最有价值的信息, 来提高推荐的准确率和效率. 实验结果表明, 我们提出的基于有效特征子集选取的高效推荐算法比目前的推荐算法有更好的性能.

本文的安排如下: 第 2 节我们介绍相关的背景知识, 第 3 节来详细阐述本文提出的基于有效子集提取的高效推荐算法, 第 4 节我们做了一些实验来测试我们提出的模型的性能, 我们将在第 5 节中给出结论.

1 相关的背景知识

1.1 特征子集选取的相关知识

特征选择[14,15](feature selection) 也称特征子集选择 (Feature Subset Selection, FSS), 或属性选择 (attribute selection). 是指从已有的 M 个特征 (feature) 中选择 N 个特征使得系统的特定指标最优化, 是从原始特征中选择出一些最有效特征以降低数据集维度的过程, 是提高学习算法性能的一个重要手段, 也是模式识别中关键的数据预处理步骤. 目前已有不少文献中提出了有监督学习的特征选择算法, 但是仅有少数文章对于无监督学习的特征选择问题做了研究. 无监督学习的特征选择问题就是依据一定的判断准则, 选择一个特征子集能够最好地覆盖数据的自然分类. 目前的方法有基于遗传算法的特征选择方法[16]、基于模式相似性判断的特征选择方法[17]和信息增益的特征选择方法[18], 这几种方法没有考虑特征之间的相关性和特征对分类的影响.

1.2 UV 分解技术

矩阵的 UV 分解技术[19]是将原始用户-项目评分矩阵 R 分解为用户特征矩阵 P 和项目特征矩阵 Q, 即每个用户 u(或者项目 i) 由一个实数向量 p_u(或者 q_i) 表

示, 而这个向量的维度远远小于用户或者商品的个数. 对于用户 u 来说, p_u 代表用户 u 的用户的喜好, 类似, 对于项目 i, q_i 代表项目 i 的特性. 他们的内积$q_i^{\mathrm{T}} p_u$表示用户 u 对项目 i 的评分, 所以, 用户的评分可以用如下公式表示:

$$\overline{R}=Q^{\mathrm{T}}P \tag{1}$$

$\overline{R}$是模型的预测评分矩阵.

为了得到 P 和 Q, 我们采用最小化的方法为:

$$\sum(r_{ui}-q_i p_u)+\lambda_1(|p_u|^2+|q_i|^2) \tag{2}$$

其中, r_{ui} 是评分矩阵 R 的第 u 行、第 i 列的已知打分值, p_u 是用户特征矩阵 P 的第 u 行; q_i 是项目特征矩阵 Q 的第 i 行; $\lambda_1\left(|p_u|^2+|q_i|^2\right)$是为了避免过拟合而附加的正则项.

由于辅助域是稀疏度相对较低, 所以我们可以利用 UV 分解技术对辅助域进行数据的填充, 这非常有利于对辅助域进行聚类, 提取更好的有用信息.

1.3 协同过滤算法的相似度计算

我们采用基于用户的协同过滤算法[20]进行推荐, 首先要计算出用户 (项目) 之间的相似度. 常用的计算相似度的方法主要有欧式距离、余弦相似度和皮尔逊相关系数等. 本文采用的皮尔逊相关系数作为用户相似度的度量方法. 皮尔逊相关系数其度量方法表示为:

$$sim(u,v)=\frac{\sum\limits_{i\in I_{uv}}(r_{ui}-\bar{r}_u)(r_{vi}-\bar{r}_v)}{\sqrt{\sum\limits_{i\in I_{uv}}(r_{ui}-\bar{r}_u)^2}\sqrt{\sum\limits_{i\in I_{uv}}(r_{vi}-\bar{r}_v)^2}} \tag{3}$$

其中, I_{uv} 表示用户 u 和 v 共同评分的项目, r_{ui}, r_{vi} 分别表示用户 u 和 v 对项目 i 的评分, $\bar{r}_u$和$\bar{r}_v$分别表示用户 u 和 v 对项目的平均评分.

推荐系统就是为用户提供可靠准确的服务, 使用户可以高效的从海量信息中快速得到自己想要的信息. 推荐系统算法常用的标准之一就是平均绝对误差 *MAE*, 它通过计算真实评价与预测评价之间的误差来衡量推荐结果的准确性. 平均绝对误差越小, 推荐结果的准确性就越高. 假设用户的评分集合表示为$\{v_1, v_2, \cdots, v_N\}$, 对应的预测评分集合为$\{p_1, p_2, \cdots, p_N\}$则具体的 *MAE* 计算公式为:

$$MAE=\sum_{i=1}^{N}\frac{|v_i-p_i|}{N} \tag{4}$$

2 基于有效特征子集选取的高效推荐算法

2.1 有效特征子集的选取

传统的跨域推荐系统在进行特征选择时, 往往是将辅助域和目标域特征进行联合选取, 这样会使得在目标域上稀疏数据的有效信息进一步减少, 这将会大大降低推荐精度. 为了保护目标域数据, 我们的模型仅对辅助域进行特征选取, 尽可能的提取辅助域上对目标域有用的信息, 提高推荐的准确率和效率. 由于目标域和辅助域不存在明显的指标, 所以我们要进行无监督的特征提取. 为了获取辅助域上最有效的信息, 最大限度的降低冗余和相关特征, 我们采用 K-means 算法来获取特征子集.

2.1.1 填充辅助域的缺失值

由于提取的辅助域信息存在有很多缺失值, 如果对这些数据直接进行聚类会大大降低聚类的效果, 所以我们要对缺失值进行填充, 处理缺失值的方法有很多, 本文采用矩阵 UV 分解的方法对缺失值进行填充. 将填充后的矩阵进行下一步的处理.

2.1.2 估计聚类趋势

聚类趋势度量指数据集是否有聚类的价值, 如果数据集是随机均匀地分布, 则聚类的价值很低. 我们常用空间随机性的统计检验来实现, 基于这种思想, 我们采用一种简单有效的统计量, 霍普金斯统计量. 计算霍普金斯统计量的步骤是:

(1) 均匀的从 D 的空间中抽到 n 个点 $p_1, \cdots, p_n$. 也就是说, D 空间中的每个点都以相同的概率包含在这个样本中, 对于每个点 $p_i(1<=i<=n)$, 我们找出 p_i 在 D 中的最近邻, 并令 x_i 为 p_i 与它在 D 中的最近邻之间的距离, 即$x_i=\min(dist(p_i,v))$, v 属于 D.

(2) 均匀地从 D 中抽到 n 个点 $q_1, \cdots, q_n$. 对于每个点 $q_i(1<=i<=n)$, 我们找出 q_i 在 $D-\{q_i\}$中的最近邻之间的距离, 即$y_i=\min(dist(q_i,v))$, v 属于 D, v 不等于 q_i.

(3) 计算霍普金斯统计量:

$$H=\frac{\sum\limits_{i=1}^{n}y_i}{\sum\limits_{i=1}^{n}y_i+\sum\limits_{i=1}^{n}x_i} \tag{5}$$

2.1.3 确定聚类簇数 k 的大致区间

对于 K-means 算法来说, 聚类数 k 的确定无疑是最重要的一个步骤. 为了得到更好的分类簇数, 我们将选用肘方法来得到 k, 获取对目标域最有价值的信息.

肘方法的原理是, 当我们增加簇数的时候, 数据集的簇内方差之和会降低, 以簇数为自变量, 簇内方差为因变量做一条曲线, 这条曲线会随簇数的增加而下降. 生成的这条曲线会有一个明显的拐点, 这个拐点附近就是我们要找的 k.

2.1.4 利用 K-means 算法进行特征子集选取

为了最大可能的减少辅助域的相关特征, 提取对目标域最有价值的信息, 我们首先对辅助域进行聚类. 因为目标域和辅助域虽然在具体项目上可能不同, 比如图书和电影, 但是他们在题材标签上是非常相似的. 所以我们首先通过聚类方法来获取题材标签 tag=$\{t_1, t_2, \cdots, t_m\}$, 然后用每个标签的平均值作为用户 u_i 对此标签 t_j 的评分 v_{ij}.

对于初始点 k 的选取, 我们应做到尽可能的选择相互距离较远的点. 在选择 k 个初始点的时候, 首先随机选择一个点作为初始簇中心, 然后选择距离该点相对较远的那个店作为第二个初始类簇中心, 然后选择距离该两点相对较远的点作为第三个初始类簇中心, 以此类推, 直至找到 k 个初始类簇中心.

K-means 算法产生聚类簇的过程:

1) 在所有的项目中挑选 k 个项目作为初始聚类点;

2) 计算每个聚类中心与其他项目的相似度, 依据相似度将项目分配到相应的类簇;

3) 计算生成的类簇的中心点.

重复 2), 3) 操作, 直到各个类簇的中心点不再发生变化.

在图 1 中我们假设辅助域中用户 u 对项目 I 的评分为 r_{ij}, $\{I_1, I_3, \cdots, I_n\}$为一类, 我们通过聚类生成标签 t_1, 对于用户 u_i 对标签 t_1 的评分为 v_{i1}, v_{i1} 是 u_i 对 $I_1, I_3, \cdots, I_n$ 的平均值, 也就是说聚类后的项目标签评分矩阵为 $U\times T\text{–>}V$, 用户对标签的评分计算公式为:

$$v_{ij}=\frac{\sum_{I\in t} r_{ij}}{N} \tag{6}$$

这里 t 表示生成的标签, N 表示属于第 j 个标签的项目数.

为了得到用户标签评分矩阵, 我们首先将用户项目评分矩阵进行转置, 得到项目用户矩阵, 然后对项目进行聚类, 得到项目标签, 我们对提取的项目标签用这一标签内的项目评分均值作为标签矩阵, 也就是标签用户矩阵, 最后将用户标签矩阵对目标矩阵进行扩展, 作为推荐算法的输入数据.

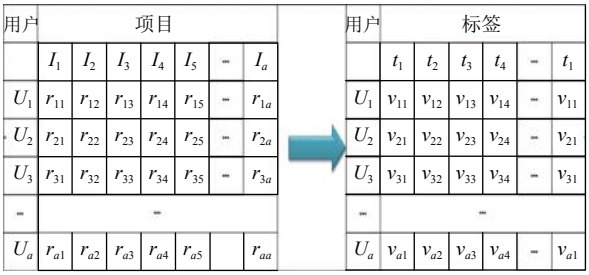

图 1 Kmeans 聚类模型

2.2 推荐算法

我们将得到的辅助域特征子集对目标域进行扩展, 得到扩展的用户项目评分矩阵, 用户还是$\{u_1, u_2, \cdots, u_m\}$, 我们对项目进行了扩展$\{i_1, i_2, \cdots, i_n, t_1, t_2, \cdots, t_k\}$共有 $m+k$ 个项目, 用户 u_i 对项目 j 的评分为 v_{ij}, 分值在 0–5 之间.

得到扩展的用户评分矩阵之后, 我们用式 (1) 来计算用户之间的相似度, 最后选择与目标用户最相似的 n 个用户为最近邻集合. 通过扩展后的用户项目评分矩阵获取最近邻之后, 我们对目标用户进行评分预测, 并向目标用户推荐前 N 项结果. 评分预测公式为:

$$pred(a,p)=\bar{r}_a+\frac{\sum_{b\in N} sim(a,b)*(r_{b,p}-\bar{r}_b)}{\sum_{b\in N} sim(a,b)} \tag{7}$$

其中, $sim(a, b)$ 表示用户 a 和 b 的相似性, $\bar{r}_b$是用户 b 的平均分, N 表示用户 a 的近邻集合. 这个公式考虑了与用户 a 最相似的 N 个近邻的评分偏差.

从算法的复杂度来看, 如果将未处理的目标域和辅助域进行联合, 那么我们采用基于用户的协同过滤方法来进行实验的算法复杂度为 $O(m^2\times(n+L))$, 其中 m 为用户数, n 为目标域的项目数, L 为辅助域的项目数. 本文将辅助域的项目进行了提取, 辅助域提取信息时, 矩阵分解的复杂度为 $O(m\times L\times k)$, 其中 L 为辅助域的项目数, k 为聚类的簇数, 可以视为常数. 辅助域进行 K-means 聚类的复杂度为 $O(m\times L)$, 进行辅助域的特征提取之后, 与目标域联合进行协同过滤的复杂度为 $O(m^2\times(n+k))$, 其中 k 为聚类的簇数, 视为常数, 所以复杂度为 $O(m^2\times n)$. 综上本文的计算复杂度为 $O(m^2\times n)$. 所以本文在计算复杂度上与已有的跨域推荐算法有一定的降低.

2.3 算法步骤

本文提出了基于有效特征子集选取的高效推荐算

法, 该算法有效的解决了推荐算法常见的数据稀疏性和特征冗余问题, 不仅提高了系统的运算效率而且也提高了推荐的准确率. 具体步骤如下:

1) 根据原始文本数据中提取目标域和辅助域的用户项目评分矩阵.

2) 对辅助域数据进行缺失值的填充.

3) 在辅助域上进行聚类并获得标签数据和评分, 并对目标域数据进行扩充.

4) 根据用户项目评分矩阵进行相似度计算, 创建目标用户的最近邻集合.

5) 对目标用户项目进行评分预测.

6) 对推荐结果进行分析, 计算得到 MAE.

算法流程图, 如下所示:

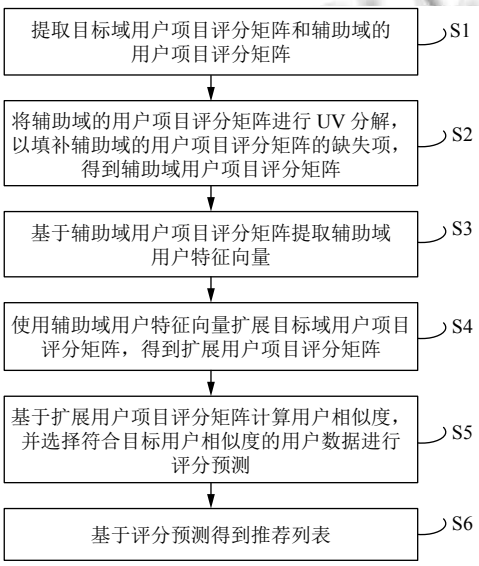

图 2 算法流程图

3 实验

3.1 数据准备

数据的完整性和真实性对实验的成功至关重要. 本文最终采用爬取的亚马逊数据集作为实验数据. 亚马逊数据集包含 1555 多万个用户对 753 243 个项目的评分, 其中项目包含图书, 电影, 音乐等方面. 这其中包含物品的 ID, 分类号 (ASIN), 标题 (title), 分类 (group), 用户的评分等信息.

我们对以上文本类数据进行了提取并合并了用户在每个领域的评分, 得到用户-项目评分矩阵 (u, i, r), r 是用户 u 对项目 i 的评分. 本实验的前提是数据集之间需要有较强的相关性, 所以我们采用评分数据相对较多的图书作为辅助域, 较稀疏的 DVD 作为目标域.

本文筛选出 1410 个用户 1000 本图书作为辅助域, 稀疏度为 23.3%, DVD 则选择 1410 个用户和 3000 个项目作为目标域, 稀疏度为 2.667%, 将以上数据作为本文的实验数据.

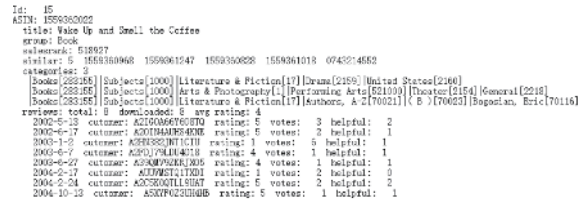

图 3 实验源数据格式

3.2 实验结果及分析

实验使用 10 折交叉验证的方法, 我们将数据集的 75% 作为训练集, 25% 作为测试集. 训练集的数据用来对算法各个步骤的参数进行计算和对测试集的评分进行预测; 测试集则用来衡量算法的推荐质量.

我们对辅助域进行聚类, 首先来估计辅助域的聚类趋势. 通过对辅助域的计算得到霍普金斯统计量为 0.2185<0.5, 说明辅助域的数据分布是不均匀的, 可以对辅助域进行聚类分析.

本实验采用肘方法来确定聚类数 k 的大致聚类区间, 最终我们得到如图 4 所示.

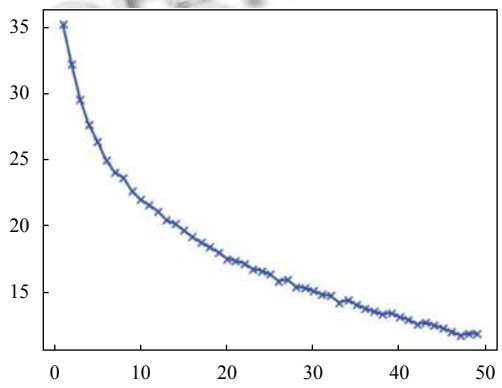

图 4 聚类数 k 的大致区间

由图 4 可以看出, k 的大致取值范围在 10 左右, 所以我们在聚类时, 在 k=10 周围进行选取最佳的值, 并与其他数据点进行比较.

本实验的目的是结合辅助域的信息标签, 对目标域采用协同过滤方法对目标用户进行推荐, 并与单域

上基于用户的协同过滤 (CF-U), 传统的跨域协同过滤 (CDCF) 和基于矩阵分解的协同过滤方法做比较. 我们分别选取聚类数 k={8, 10, 15, 25, 35}, 近邻数 N={5, 10, 20, 30, 40, 50, 60, 70, 80}来进行实验, 最终平均绝对偏差 (MAE) 随我们选择不同的近邻用户数 N 和聚类数 k 的变化如表 1 所示.

表 1 亚马逊数据集的实验结果

算法	k	N=5	N=10	N=20	N=30	N=40	N=50	N=60	N=70	N=80
CF-U	-	1.321	1.180	1.067	1.023	0.999	1.024	1.054	1.085	1.155
CDCF	-	0.912	0.845	0.837	0.833	0.83	0.836	0.846	0.862	0.867
MF-CDCF	-	0.855	0.841	0.826	0.821	0.82	0.822	0.837	0.845	0.855
FSERA	8	0.761	0.790	0.792	0.801	0.809	0.806	0.809	0.812	0.809
	10	0.721	0.779	0.790	0.8	0.804	0.803	0.805	0.809	0.809
	15	0.757	0.789	0.79	0.805	0.805	0.808	0.813	0.817	0.818
	25	0.759	0.789	0.823	0.835	0.843	0.845	0.839	0.838	0.836
	35	0.787	0.799	0.822	0.842	0.844	0.846	0.847	0.844	0.846

我们选择表 1 中具有代表性的结果以折线图的形式展现, 我们可以直观的看到 MAE 随着 k 和 N 值的变化曲线:

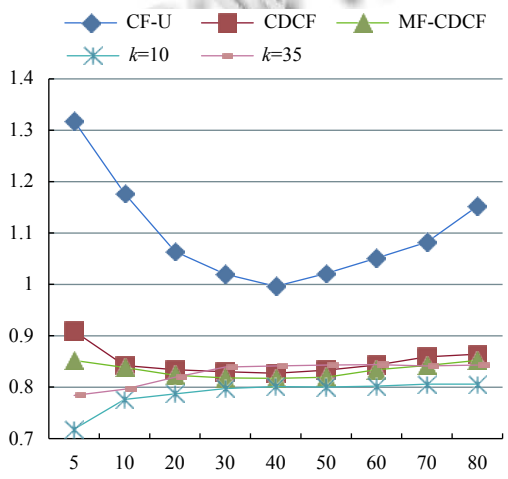

图 5 实验结果折线图

从图 5 看出, 单域上基于用户的协同过滤 (CF-U) 算法随着近邻数 N 的增加先减小后增加, 但是整体上, 单域的协同过滤算法 MAE 高于跨域的协同过滤算法. 说明引入辅助域信息可以提高推荐系统的准确率. 传统的跨域协同过滤算法 CDCF 和基于矩阵分解的跨域推荐算法 MF-CDCF 虽然比单域的推荐效果好, 但是 MAE 值还是较高. 从图 4 中可以看出, 本文所提出的算法依赖于所选择的聚类数 k 值, 如果 k 值选取不合适, 可能会使得推荐算法的质量下降. 基于有效子集选取的高效推荐算法 (FSERA) 只要选取合适的 k 值, 本系统的 MAE 值就远远低于其他算法, 这也表明了 FSERA 可以为用户提供较好的推荐.

随着聚类数 k 值的增加, 改进的协同过滤 MAE 会逐渐上升, 当 k=10 时, MAE 的曲线最低, 此时的算法质量最好; 当选取的近邻数 N 逐渐增加时, 跨域推荐系统的 MAE 值在不断增加. 我们从图 5 中也可以看出, 当引入辅助域后, 协同过滤的 MAE 会减小, 引入辅助域增加了系统的信息, 使得协同过滤算法在计算相似度时有了更多的有用数据. 辅助域聚类数为 10 时, 本文提出的 FSERA 算法的推荐质量最高.

从图 5 中我们可以看出, 基于有效子集提取的高效推荐算法在选取合适的参数时, MAE 可以达到 0.77 左右, 这说明本文的方案是可行的. 但是从特征选的的角度来看, 本文所采用的 K-means 方法来提取特征, 具有一定的局限性, 特征提取的方法有很多种, 在本文提取特征中, 选取 K-means 聚类方法有很好的效果, 但是也有一定的不足, 比如辅助域的数据也很稀疏, 聚类效果不佳, 可以考虑加入这种缺失信息的方法来达到更好的效果.

通过与其他推荐算法的比较可以看出, 本文所提出的 FSERA 在推荐质量上略优于其他算法, 但是此算法对辅助域进行了特征选择, 使得算法的复杂度大大降低, 极大的提高了系统的运算速度, 使推荐效率有了大幅度的提升.

4 结束语

传统的协同过滤方法在应用于推荐系统时, 存在着若干问题, 比如数据稀疏, 数据冗余等问题. 本文提出了基于有效特征子集选取的高效推荐算法, 有效的将辅助域信息迁移到目标域中, 对目标域数据进行了扩展. 并解决了丰富的辅助域信息直接对目标域进行

扩展带来的数据冗余等问题. 不仅降低了运算的复杂度, 并且提高了推荐效率. 在 amazon 数据集上的实验表明, 该算法很好的挖掘了用户的兴趣, 有效的降低了平均绝对偏差, 在一定程度上缓解了数据稀疏性问题, 该方法在电子商务系统中有一定的实用价值. 但是本文在特征选取时采用了 K-means 聚类方法有一定的局限性, 我们可以尝试更多的特征选择方法来进行选择, 找到更多能提高推荐效果的方法, 这也将是我以后研究的工作.

参考文献

1 Bobadilla J, Ortega F, Hernando A, *et al*. Recommender systems survey. Knowledge-Based Systems, 2013, 46: 109–132. [doi: 10.1016/j.knosys.2013.03.012]

2 He C, Parra D, Verbert K. Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities. Expert Systems with Applications, 2016, 56: 9–27. [doi: 10.1016/j.eswa.2016.02.013]

3 艾丹祥, 左晖, 杨君. 基于三维协同过滤的 C2C 电子商务推荐系统. 计算机工程与设计, 2013, 34(2): 702–706. [doi: 10.3969/j.issn.1000-7024.2013.02.060]

4 熊聪聪, 邓滢, 史艳翠, 等. 基于协同过滤的美食推荐算法. 计算机应用研究, 2017, 34(7): 1985–1988.

5 邵阔义, 班晓娟, 王笑琨, 等. 基于交通网络数据优化的地理信息推荐系统. 工程科学学报, 2015, 37(12): 1651–1657.

6 Lanier CR. Problem solving in user networks: Complex communication issues and item-to-item collaborative filtering. Communication Design Quarterly Review, 2015, 3(3): 33–39. [doi: 10.1145/2792989]

7 Pan WK, Xiang EW, Liu NN, *et al*. Transfer learning in collaborative filtering for sparsity reduction. Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence. Atlanta, GA, USA. 2010. 230–235.

8 Berkovsky S, Kuflik T, Ricci F. Cross-domain mediation in collaborative filtering. In: Conati C, McCoy K, Paliouras G, eds. Lecture Notes in Computer Science, Vol.4511. Springer, Berlin, Heidelberg. 2007: 355–359.

9 Hu L, Cao J, Xu GD, *et al*. Cross-domain collaborative filtering via bilinear multilevel analysis. Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence. Beijing. 2013. 2626–2632.

10 Singh AP, Gordon GJ. Relational learning via collective matrix factorization. Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Las Vegas, NV, USA. 2008. 650–658.

11 Li B, Yang Q, Xue XY. Can movies and books collaborate?Cross-Domain collaborative filtering for sparsity reduction. Proceedings of the 21st International Jont Conference on Artifical Intelligence. Pasadena, CA, USA. 2009. 2052–2057.

12 Pan WK, Yang Q. Transfer learning in heterogeneous collaborative filtering domains. Artificial Intelligence, 2013, 197: 39–55. [doi: 10.1016/j.artint.2013.01.003]

13 Xin X, Liu ZR, Lin CY, *et al*. Cross-domain collaborative filtering with review text. Proceedings of the 24th International Conference on Artificial Intelligence. Buenos Aires, Argentina. 2015. 1827–1833.

14 Li JD, Cheng KW, Wang SH, *et al*. Feature selection: A data perspective. ACM Computing Surveys, 2018, 50(6): 94.

15 黄丽萍. 不完备序信息系统的集对优势度粗糙集模型. 聊城大学学报 (自然科学版), 2017, 30(1): 97–101.

16 Morita M, Sabourin R, Bortolozzi F, *et al*. Unsupervised feature selection using multi-objective genetic algorithms for handwritten word recognition. Proceedings of 7th International Conference on Document Analysis and Recognition. Edinburgh, UK. 2003. 666–670.

17 Basak J, De RK, Pal SK. Unsupervised feature selection using a neuro-fuzzy approach. Pattern Recognition Letters, 1998, 19(11): 997–1006. [doi: 10.1016/S0167-8655(98)00083-X]

18 Dash M, Liu H, Yao J. Dimensionality reduction of unsupervised data. Proceedings of 9th IEEE International Conference on Tools with Artificial Intelligence. Newport Beach, CA, USA. 1997. 532–539.

19 Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. Computer, 2009, 42(8): 30–37. [doi: 10.1109/MC.2009.263]

20 Sarwar B, Karypis G, Konstan J, *et al*. Item-based collaborative filtering recommendation algorithms. Proceedings of the 10th International Conference on World Wide Web. Hong Kong, China. 2001. 285–295.