doi:10.3969/j.issn.1672-4348.2016.03.017

基于概念分层的估值填充推荐算法

蒋雯

(福州理工学院 管理工程系, 福建 福州 350506)

摘要: 为解决传统协同过滤算法中存在的数据稀疏性问题,在原有估值公式的基础上对传统的协同过滤算法进行改进,提出一种基于概念分层的估值填充推荐的改进算法,并对此算法进行仿真实验。结果表明,该算法在稀疏数据集上有着良好的推荐效果。

关键词: 协同过滤; 项目推荐; 概念分层; 估值填充

中图分类号: TB23　　**文献标志码**: A　　**文章编号**: 1672-4348(2016)03-0302-05

A recommended algorithm of valuation filling based on concept hierarchy

Jiang Wen

(Management Engineering Department, Fuzhou Polytechnic College, Fuzhou 350506, China)

Abstract: To deal with data sparseness problems in the traditional collaborative filtering algorithm, a new recommendation algorithm of valuation filling based on concept hierarchy was proposed through improving the traditional collaborative filtering algorithm. Simulation experiments of the new recommendation algorithm were conducted. The results indicate that the proposed algorithm has favourable recommendation effect in sparse data sets and can improve the quality of recommendations.

Keywords: collaborative filtering; projects recommendation; concept hierarchy; valuation filling

协同过滤推荐系统是电子商务网站提高经济效益的有效技术手段,能够主动快速地挖掘出潜在的购买用户,并帮助他们找到感兴趣的商品,在增加网站的商品销量的同时也增加了用户对商品网站的忠诚度。但随着电子商务规模的不断壮大,用户和项目数据急剧增加,用户评分数据变得极端稀疏[1],严重影响着推荐结果的准确性。针对数据稀疏问题,研究学者已提出了众多解决方法,但至今没有一种方法能够彻底解决此问题。Sarwar[2]等人提出了利用单值分解的方法对原始稀疏数据进行数据处理,这种方法能够提高推荐效果,但是在分解过程中却造成了数据的遗失。BP 神经网络的填充方法、聚类技术近年来也被用于解决数据稀疏性问题,但这些方法的最大缺点是可扩展性问题,其计算量会随着数据量增大而急剧增加,推荐速度变慢。鉴于此,针对稀疏性问题,协同过滤推荐算法依然有很大的改进空间。

1　协同过滤推荐概述

协同过滤推荐(collaborative filtering)[3]又称为社会过滤。Goldberg 等学者于 1992 年首次提出了该概念。协同过滤推荐的关键在于假设目标客户的兴趣可以根据其他类似客户的兴趣进行预测推荐,突出了人与人之间的协作。传统的协同过滤推荐算法是指基于用户的协同过滤算法,其基本思想是通过计算用户间相似度,找出目标用户的邻居用户,基于最近邻居的评分数据对用户未购买的项目进行预测评分,并向目标用户产生推荐。整个算法可以概括为 3 个阶段:建立用户模型、获取最近邻居、产生推荐列表[4]。

收稿日期: 2016-04-27
基金项目: 福建省中青年教师教育科研项目(JAS150738)
作者简介: 蒋雯(1983-),女,福建漳州人, 讲师,硕士,研究方向:营销管理。

1)建立用户模型。协同过滤算法中用户评分数据可以用一个 $m \times n$ 维的用户-项目矩阵 $\boldsymbol{R}_{st}$ 描述，

$$\boldsymbol{R} = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{12} & \cdots & r_{1n} \end{bmatrix} \tag{1}$$

式中 m 是用户数，n 是项目数，r_{st} 是用户 s 对项目 t 的评分，$\boldsymbol{R}_s$ 表示用户 u_s 在 n 维项目空间上的评分向量，以表 1 为例。

表 1　协同过滤举例

Tab.1　Examples of collaborative filtering

用户	项目 1	项目 2	项目 3	项目 4
用户 A	2	4	3	1
用户 B	2	r_{ij}	3	1
用户 C	1	5		3
用户 D	2	3	4	1

2)获取最近邻居。利用用户-项目评分矩阵 $\boldsymbol{R}_{st}$ 计算用户之间的相似度，找出目标用户的最近邻居集合。

3)产生推荐。根据最近邻居已购买(浏览或评价)但目标用户 u_s 尚未发现的项目形成候选项目集合，然后预测目标用户对候选项目的评分，产生 top-**N** 项目推荐集，进而产生推荐，如图 1 所示。

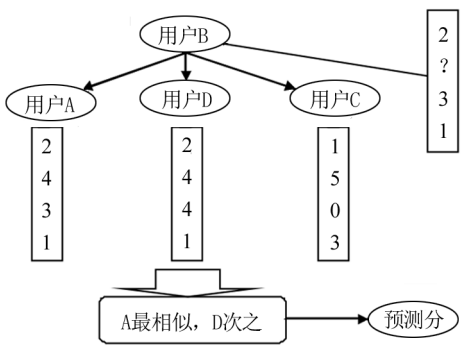

图 1　协同过滤推荐过程

Fig.1　Process of collaborative filtering

2　基于概念分层的估计填充商品推荐算法

鉴于数据稀疏性问题对提高推荐算法的准确性有重大影响，提出了基于概念分层的估值填充商品推荐算法。首先，在原有的用户估值公式基础上利用概念分层思想进行了改进，使得该算法在各项目种类上确认“用户打分尺度”和“商品受欢迎度”，并生成新的用户模型；然后，利用 Pearson 相关性方法[2]进行用户相似性计算，结合 top-**N** 推荐产生用户邻居集合；再从邻居的历史记录找出候选项目集，通过计算用户对候选项目集的兴趣度，并按降序排列，结合 top-**N** 进行项目推荐；最后，利用仿真实验验证了该算法在稀疏数据集上有着良好的推荐效果。

2.1　用户估值填充公式

针对用户评分矩阵的数据稀疏性问题，原有估值填充公式是在“用户评分尺度”和“商品受欢迎度”的基础上提出来的[5]，填充项 r_{st} 为，

$$r_{st} = \overline{R_s} + \frac{1}{|U'|}\sum_{s' \in U'} (r_{s't} - \overline{R_{s'}}) \tag{2}$$

其中，用户 s 评分尺度为 $\overline{R_s} = \dfrac{1}{|R_s|}\sum_{t=1}^{n} r_{st}$；项目 t 受欢迎度为 $\dfrac{1}{|U'|}\sum_{s' \in U'} (r_{s't} - \overline{R_{s'}})$ 。

公式(2)在一定程度上能够解决用户评分矩阵数据稀疏性问题，但在填充过程中并未考虑项目所属类别。项目种类不同，用户感兴趣程度不同，用户打分尺度也不尽相同。因此，利用该估值公式填充矩阵产生的推荐结果不够精准。即，用户 u_s 对与项目 t 所属类别相差比较大的其他项目的评分不具参考性，若以用户 u_s 对所有项目的平均打分尺度 $\overline{R_s}$ 来衡量用户 u_s，则对商品 t 的评分明显不够精准。

2.2　基于概念分层的用户估值填充

概念分层是一种广泛应用于数据挖掘领域的数据分类方法。它定义一个映射序列，将低层概念映射到更一般的较高层概念[6]。它实质上是以层次的形式、偏序的关系来表示数据或概念。一般用树结构来表示，其中树的节点代表概念，树枝代表偏序关系。概念分层可以由领域专家人工地提供，或根据数据分布的统计分析自动生成。

考虑维 location(地点)的概念分层。location(地点)的城市值包括温哥华、多伦多、纽约和芝加哥。每个 city(城市)可以映射到它们所属的省或州。比如，温哥华可以映射到不列颠哥伦比亚；芝加哥映射到伊利诺伊；这些省和州又可以映射到它们所属的国家，如加拿大或美国。这些映射

形成维 location(地点)的概念分层,将低层概念(城市)映射到更一般的较高层概念(国家)。维 location(地点)的概念分层树详见图 2。

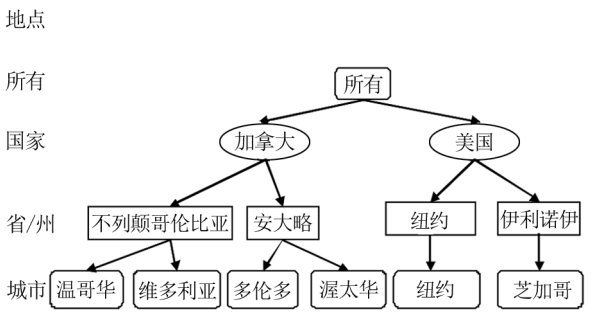

图 2 维 location(地点)的概念分层树

Fig.2 Concept hierarchy tree of location

针对稀疏性问题,原有估值填充公式预测结果不够精准。为避免其缺陷,文中对算法进行了改进,运用基于概念层次树的用户-项目种类估值填充数据矩阵。改进的思路大致为:利用概念分层思想引入了项目分类,在项目种类上确认“用户评分尺度”和“商品受欢迎度”,以完成更加精准的估值计算,进而提高商品推荐质量。改进后的基于概念分层评分估值填充公式如下,

$$r_{st} = \underset{t \in c_t}{R_{sc_t}} + \frac{1}{|U'|}\sum_{\substack{s' \in U' \\ t \in c_t}} (r_{s't} - R_{s'c_t}) \tag{3}$$

其中,将用户 u_s 对项目 t 所属种类 c_t 的评分作为用户评分尺度 R_{sc_t};

商品受欢迎度为 $\frac{1}{|U'|}\sum_{\substack{s' \in U' \\ t \in c_t}} (r_{s't} - R_{s'c_t})$,其中 U' 表示评论过项目 t 的用户集合。

2.3 评分数据转换模型

一般而言,协同过滤推荐电子商务系统只提供用户-项目评分数据,并没有用户-项目种类评分资料。如何将用户-项目评分转换为用户-项目种类评分数据成为问题关键所在,因此,建立了数据转换模型。

2.3.1 评分数据转换的假定前提

为了使该模型更科学地进行数据转换,设计了以下假定前提:

1)每位用户对所有项目种类的总评分都是 S。统一总评分的设定保证了各用户在同一评分范围内对所有项目种类进行评分预测。

2)因为一个项目可能同时归属于多个种类。假定在项目 t 所属的 $|f(t)|$ 个种类中,每个种类分摊的评分值相等。其中,$f(t) \subseteq C$ 是项目 t 所属种类的子集。

3)概念层次树中的各路径结点的分值自底向上逐层递减。因为在概念层次树中,路径结点越往下项目种类就越细致。而数据填充时,评估的种类越细致,受用户关注程度就越高。所以,在概念层次树中路径结点越往下分配的评分值自然就越多。

2.3.2 评分数据转换过程

总评分 S 分配到概念层次树各结点(即各项目种类)的过程如下:

1)对于稀疏的评分矩阵的空白项 r_{st},在进行估值填充时,按照分摊比例从 S 中分享评分。即,空白项 r_{st} 获取的 R_{sc_t} 的初始分值 $S(c_t)$,其公式为:

$$\underset{c_t \in f(t)}{S(c_t)} = S \times \left(\sum_{t \in c_t} r_{st} \Big/ \sum_{t=1}^{n} r_{st} \right) \tag{4}$$

2)将获取的用户评分尺度 R_{sc_t} 的初始分值 $S(c_t)$ 按照一定规则分配到概念层次树各路径的各结点,各结点获取的分值 $s(p_t)$,其公式为:

$$\sum_{t=0}^{k} s(p_t) = S(c_t) \tag{5}$$

$$s(p_t) = \frac{s(p_{t+1})}{b(p_{t+1}) + 1} \tag{6}$$

其中,$(p_0, p_1, \ldots, p_k)$ 表示概念层次树中自顶向下的一条路径,p_j 为路径上的各结点,$b(p_t)$ 为结点 p_t 的姐妹结点个数。

3)汇总种类 c_t 中的各结点获取的分值,即可得出用户评分尺度 R_{sc_t} 的最终分值。

4)重复上述步骤 1~3 获取 $R_{s'c_t}$ 的最终分值。

5)重复上述步骤 1~4,再结合基于概念层次的估值公式完成整个稀疏矩阵的填充工作。

2.4 生成最近邻居

协同过滤算法的关键在于寻找与目标用户兴趣爱好相似的邻居,根据邻居已经浏览或评价或购买但目标用户还未发现的项目向目标用户产生推荐。在寻找邻居的过程中:首先,利用 Pearson 相关方法计算用户之间的相似度,并将计算出的相似度按照从高到低的顺序进行排列,形成目标用户的相似度集合;然后,根据预定的相似度阈值或预定的邻居个数进行 top-**N** 选择,以确定目标用户 u_s 的邻居集合。

利用 Pearson 相关方法计算的用户 u_i 和用户 u_j 的相似度 $\mathrm{sim}(u_i,u_j)$，公式如下：

$$\mathrm{sim}(u_i,u_j)=\frac{\sum_{i_k\in I_{ij}}(r_{ik}-R_{ic_k})(r_{jk}-R_{jc_k})}{\sqrt{\sum_{i_k\in I_{ij}}(r_{ik}-R_{ic_k})^2}\sqrt{\sum_{k\in I_{ij}}(r_{jk}-R_{jc_k})^2}} \quad (7)$$

其中，I_{ij} 为用户 u_i 和用户 u_j 共同评分过的项目集合；R_{ic_k} 或 R_{jc_k} 为用户 u_i 或 u_j 获取的对项目所属种类 c_k 的评分。

2.5　形成推荐

邻居集合 $\mathbf{N}(u_s)=\{n_1,n_1,\ldots n_p \mid p\leqslant m\}$ 形成以后，首先将集合 $\mathbf{N}(u_s)$ 中的邻居用户已经浏览或评价或购买但目标用户 u_s 还未发现的项目组成推荐候选项目集合 $\mathrm{CI}(u_s)$，然后目标用户对候选集合 $\mathrm{CI}(u_s)$ 中的各项目进行评估值预测，最后将预测的评估值从大到小排列，选出 top-**N** 项目并产生最终推荐。

$$\mathrm{Pre}(i_{t,n_j})=R_{sc_t}+\frac{\sum_{n_j\in N(u_s)}\mathrm{sim}(u_s,n_j)(r_{jt}-R_{jc_t})}{\sum_{n_j\in N(u_s)}\mathrm{sim}(u_s,n_j)} \quad (8)$$

其中，$i_{t,n_j}\in \mathrm{CI}(u_s)$ 表示邻居 n_j 已经浏览或评价或购买但目标用户 u_s 还未发现的项目；R_{sc_t} 或 R_{jc_t} 表示目标用户 u_s 或邻居 n_j 对项目 i_{t,n_j} 所属种类的评分（据基于概念分层的估值方法评估）；r_{jt} 表示邻居 n_j 对项目 i_{t,n_j} 的评分。

3　实验设计

利用 MovieLens 网站 www.grouplens.org 提供的数据集，并参考 eBay 网中的电影分类构建概念层次树进行实验，用户对电影的评分值为 1~5 的整数。实验中，将提出的基于概念分层的估计填充商品推荐算法 CF1 与原有的评估填充算法 CF 从推荐准确性以及推荐全面性角度进行了对比分析。

3.1　实验对比指标

（1）准确率（100%）指标

用均值绝对误差（mean absolute error，MAE）衡量整个检验集合中的平均误差，公式如下：

$$\mathrm{MAE}=\frac{1}{|\mathrm{CI}'_s|}\sum_{t=1}^{|\mathrm{CI}'_s|}|p_{st}-r_{st}|\times 100\% \quad (9)$$

式中，$\mathrm{CI}'_s\subseteq \mathrm{CI}_s$ 为 top-**N**=3 时进行推荐的项目集合；p_{st} 为预测的用户 u_s 对项目 $i_t\in \mathrm{CI}'_s$ 的评分；r_{st} 为用户 u_s 对项目 i_t 的真实评分。

（2）查全率（100%）指标

为了对推荐算法的全面性进行验证，引入了查全率，主要验证推荐项目占用户实际感兴趣项目的比重。

$$\mathrm{Recall}=\frac{|\,|\mathrm{CI}'_s|\cap \mathrm{TR}_s\,|}{|\mathrm{TR}_s|}\times 100\% \quad (10)$$

式中，TR_s 为用户 u_s 的真实感兴趣的项目集合，在测试数据集中体现为评分 ≥4 的项目集合；$|\mathrm{CI}'_s|\cap \mathrm{TR}_s$ 表示对用户 u_s 的推荐与其真实感兴趣相重叠的项目集合。

3.2　实验方法

提取的实验数据针对的是评分项目数为 0~160 时的用户。因为在该区间内，用户数随着用户评分项目的增加而增加，该区间的数据具有层次性，实验效果比较明显。

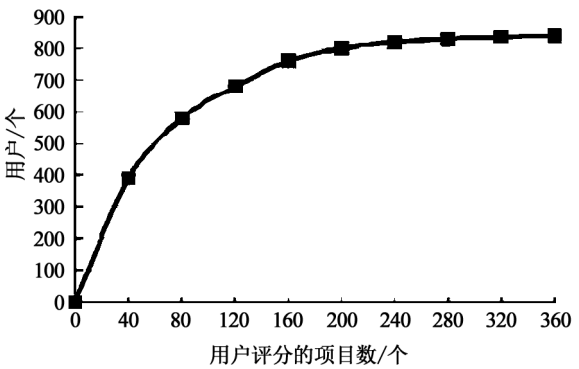

图 3　用户评分分布图

Fig.3　User rating distribution

实验中，为简便计算，分别从用户评分的项目数位于 0~40、40~80、80~120、120~160 区间内的用户集合中各选取 3 位代表用户，4 个区间总共产生 12 位代表用户。

然后，对每位代表用户在各自的评分项目数区间内，采用 4 折交叉验证技术[6]，产生 4 次推荐，取其平均值作为该代表用户的推荐评估结果；按照同样方法，计算出 12 位代表用户在各评分项目数区间内的推荐评估结果；最后，将这 12 位代表用户的推荐评估结果平均化，以作为推荐算法在该用户集合（用户评分的项目数≤160）上的最终推荐评估结果。

3.3　实验结论及分析

将提出的基于概念分层的估计填充商品推荐

算法 CF1 与原有的评估填充算法 CF 从推荐准确性以及推荐全面性角度进行了对比分析。

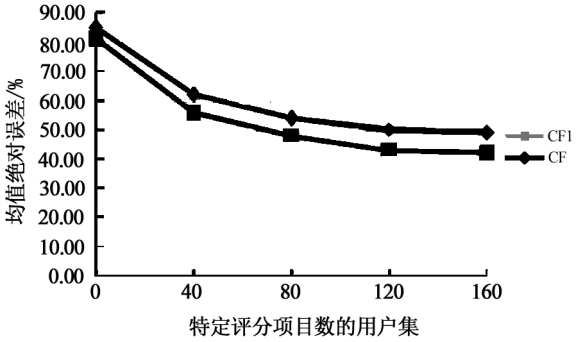

图 4 算法 CF1 与 CF 的均值绝对误差 MAE 对比结果

Fig.4 The comparison between mean absolute error (MAEtive) results of algorithm CF1 and CF

如图 4 所示,所提出的基于概念分层的估计填充商品推荐算法 CF1 的均值绝对误差 MAE 相比算法 CF 更小一些;并且,随着用户评分项目的增多,算法 CF1 与 CF 的 MAE 曲线均呈现下降趋势(用户评分项目越多,项目种类也就越多,寻找的邻居也就越准确,从而算法的均值绝对误差也就越小)。

从图 5 可知,所提出的推荐算法 CF1 的查全率相比算法 CF 更高一些。并且,随着用户评分项目的增多,算法 CF1 与 CF 的查全率曲线均呈现上升趋势(用户评分项目越多,项目种类也就越多,算法的查全率也就越高)。

通过以上对比分析,显然所提出的基于概念分层的估计填充推荐算法能够提高推荐质量,对于数据稀疏问题起到了一定的改善作用。

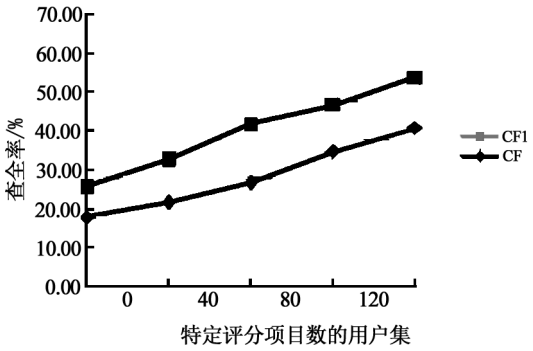

图 5 算法 CF1 与 CF 的查全率对比结果

Fig.5 The comparison of recall rate between the results of algorithm CF1 and CF

4 结语

针对推荐系统中的数据稀疏性问题,提出了一种基于概念分层的估值填充推荐改进算法,在项目种类上确认“用户评分尺度”和“商品受欢迎度”,以提高项目推荐结果的质量。最后,通过实验验证了该算法的可行性以及有效性。然而该问题依然有很大的研究空间,比如:

1) 将理论运用到实际电商网站,通过线上反馈进行进一步的算法优化。

2)数据填充虽然在一定程度上解决了数据稀疏性问题,但不能解决高维矩阵的降维问题,即数据可扩展性问题仍待解决。

3)所提出的算法主要是利用用户-项目评分数据等显性信息进行推荐,在隐性数据挖掘方面尚待研究。

参考文献:

[1] Samak A C. An experimental study of reputation with heterogeneous goods[J]. Decision Support Systems,2013,54(2):1134-1149.

[2] 工业和信息化部.电子商务“十二五”发展规划[R].北京:工业和信息化部,2012.

[3] 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.

[4] 朱郁筱,吕琳媛.推荐系统评价指标综述[J].电子科技大学学报,2012,41(2):163-175.

[5] 姜锦虎,李皓,袁帅.基于多智能体系统的分布式信誉机制研究[J].管理工程学报,2013,27(1):77-87.

[6] Tan P N, Steinbach M, Kumar V.数据挖掘导论[M].北京:人民邮电出版社,2006.

(责任编辑:肖锡湘)