嵌入LDA主题模型的协同过滤推荐算法

高 娜 杨 明
(南京师范大学计算机科学与技术学院 南京 210046)

摘 要 协同过滤推荐算法由于其推荐的准确性和高效性已经成为推荐领域最流行的推荐算法之一。该算法通过分析用户的历史评分记录来构建用户兴趣模型,进而为用户产生一组推荐。然而,推荐系统中用户的评分记录是极为有限的,导致传统协同过滤算法面临严重的数据稀疏性问题。针对此问题,提出了一种改进的嵌入LDA主题模型的协同过滤推荐算法(ULR-CF算法)。该算法利用LDA主题建模方法在用户项目标签集上挖掘潜在的主题信息,进而结合文档-主题概率分布矩阵和评分矩阵来共同度量用户和项目相似度。实验结果表明,提出的ULR-CF算法可以有效缓解数据稀疏性问题,并能显著提高推荐系统的准确性。

关键词 协同过滤,稀疏性,主题模型
中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.3.011

Topic Model Embedded in Collaborative Filtering Recommendation Algorithm

GAO Na YANG Ming
(School of Computer Science and Technology, Nanjing Normal University, Nanjing 210046, China)

Abstract Collaborative filtering(CF) recommendation algorithm has become one of the most popular algorithms in the field of recommendation due to its accuracy and efficiency. CF algorithm constructs interest models of users through analyzing their history rating records. Then it generates a set of recommendations for users. While the rating records of users in the recommendation system are limited, it results in the traditional CF algorithm facing with serious problem of data sparsity. Therefore, to address the problem of sparsity, we proposed an improved collaborative filtering recommendation algorithm that embeds the LDA topic model, named LDA-CF. This algorithm utilizes LDA topic model method to discover latent topics information in tags of users and items. Then it unifies both the document-topic probability distribution matrix and rating matrix simultaneously to measure the similarities between users and items. The experiment results indicate that the developed ULR-CF algorithm can alleviate the sparsity problem, and improve the accuracy of recommendation system simultaneously.

Keywords Collaborative filtering, Sparsity, Topic model

1 引言

近年来,Internet的飞速发展和广泛普及使得用户生活中充斥着大量的选择。电子零售商和产品提供商提供数量庞大的产品和服务来满足不同用户的不同需求,使得用户可以根据自己的兴趣爱好来选择不同的产品。但是面对海量信息,用户很难快速发现自己真正感兴趣的东西;并且,电子商务提供商也很难在短时间内把自己生产的信息推送到对其真正感兴趣的用户面前。推荐系统(Recommendation System, RS)正是解决这一矛盾的有效途径,它能为用户快速推荐满足其兴趣爱好的产品,并帮助提供商寻找到自己的忠实客户。

协同过滤(Collaborative Filtering, CF)算法[1,2]是推荐系统中最流行的推荐算法之一,它通过分析用户的历史行为记录来获取用户兴趣模型,进而为用户做出推荐。传统协同过滤算法基于用户-项目评分矩阵(Rating Matrix)进行推荐,利用评分矩阵中的已知值来估计缺失值,如图1所示。根据推荐方式的不同,协同过滤算法分为基于邻域的协同过滤算法(Neighborhood-based CF Approach)[3,4]和矩阵分解(Matrix Factorization)的协同过滤算法[5]。

	Item1	Item2	Item3	Item4	Item5
User1	2	?	3	?	5
User2	?	3	?	2	?
User3	1	?	?	?	3
User4	?	4	?	?	?
User5	?	?	3	?	4

图1 CF的任务是预测用户-项目评分矩阵中的缺失评分值

基于邻域的协同过滤算法的核心环节是计算用户或项目相似度,依赖用户或项目的邻域信息进行推荐。该算法又可以分为两类:基于用户的协同过滤算法(User-based CF, UserCF)[1]和基于项目的协同过滤算法(Item-based CF, ItemCF)[6]。这两种算法分别通过计算用户和项目相似度,将原始的用户-项目评分空间分别转化为用户-用户相似度空

到稿日期:2015-03-18 返修日期:2015-06-08 本文受国家自然科学基金(61272222),国家自然科学基金重点项目(61432008)资助。
高 娜(1989-),女,硕士,主要研究方向为机器学习、模式识别,E-mail:gaonahao@126.com;杨 明(1964-),男,博士,教授,主要研究方向为机器学习、模式识别。

间和项目-项目相似度空间，进而利用用户之间或项目之间的关系来产生推荐。

基于矩阵分解的协同过滤算法[5]实为降维的方法，该算法通过将原始的高维评分矩阵分解为两个低维矩阵乘积的形式，把用户和项目映射到同一个 f 维的隐空间。用户对项目的预测评分就表示为两个矩阵的乘积的形式。

推荐系统中用户和项目数量非常庞大，而用户往往只对其中很少一部分项目进行评分。因此，推荐系统面临严重的数据稀疏性问题。这将导致在利用 Neighborhood-based CF 算法进行推荐时，计算得到的相似度不准确，严重影响推荐系统的推荐质量。已有学者提出不同方法来解决这一问题。例如，Yifan Hu 等人在文献[7]中提出基于隐式反馈的协同过滤算法，其通过挖掘用户的隐式反馈信息来推断用户的兴趣爱好，解决了部分稀疏性问题。针对稀疏性问题，本文提出一种嵌入 LDA 主题模型的协同过滤算法（Unifying LDA and Ra-tings CF，ULR-CF）。该算法结合文档-主题概率分布矩阵和评分矩阵共同度量相似度，弥补了传统的仅利用评分矩阵度量相似度的不足。实验结果表明，本文提出的 ULR-CF 算法可以有效缓解数据稀疏性问题，并显著提高推荐系统的推荐准确性。

2 传统协同过滤推荐算法

传统协同过滤算法基于用户的历史行为记录，为用户兴趣建模并做出推荐，这些历史行为通常通过收集用户的评分记录来获得。协同过滤算法的步骤可概括为：(1)收集用户的历史行为记录，获得用户的偏好信息；(2)基于评分计算相似度，搜索目标对象的 K 近邻；(3)根据邻域信息产生推荐。

对本文涉及到的符号变量进行解释：u、v 表示用户，i、j 表示项目，r_{ui} 表示用户 u 对项目 i 的评分的真实值，$\hat{r}_{ui}$ 表示用户 u 对项目 i 的评分的预测值。评分的范围为[1,5]，值越大表示用户对项目越喜爱。对于评分已知的(u,i)二元组，将其放到集合 κ 中，即 $\kappa=\{(u,i)|r_{ui}\text{已知}\}$。

由引言部分可知，协同过滤算法包括基于邻域的算法和矩阵分解算法，本文主要研究前者。因此，下面将对基于用户的协同过滤算法和基于项目的协同过滤算法做详细介绍。

2.1 基于用户的协同过滤算法

基于用户的协同过滤算法认为，一个用户会喜欢和他有相似兴趣爱好的邻居用户所喜欢的东西。因此，UserCF 算法预测目标用户对一个项目的未知评分是基于目标用户的邻居集对该项目的已知评分进行的。在利用 UserCF 算法进行推荐时，首先要找到与这个用户有相似兴趣爱好的邻居用户，进而基于这些邻居用户的行为信息来为用户产生推荐。因此，基于用户的协同过滤算法的核心是计算用户相似度，并找到与目标用户相似度最大的前 K 个好友来构造其邻居集。基于用户的协同过滤算法的步骤描述如算法 1 所示。

算法 1 基于用户的协同过滤算法（UserCF）

步骤 1：计算用户之间的相似度，构造用户相似度矩阵；

步骤 2：选择与目标用户相似度最大的前 K 个用户作为其邻居集；

步骤 3：利用邻居集的已知评分的加权和来预测目标用户的未知评分。

这里，用户相似度的计算是基于共同评分的项目集进行的，通常利用皮尔逊相关系数[6]进行计算。假设 I 表示用户 u 和用户 v 共同评分的项目集，$\bar{r}_u$ 表示用户 u 在共同评分项目集 I 上的平均评分，$\bar{r}_v$ 表示用户 v 在共同评分项目集 I 上的平均评分。那么利用皮尔逊相关系数法计算用户 u 和 v 的相似度，如式(1)所示：

$$sim(u,v)=\frac{\sum_{i\in I}(r_{ui}-\bar{r}_u)(r_{vi}-\bar{r}_v)}{\sqrt{\sum_{i\in I}(r_{ui}-\bar{r}_u)^2}\sqrt{\sum_{i\in I}(r_{vi}-\bar{r}_v)^2}} \tag{1}$$

利用式(1)计算得到的相似度可以构造用户相似度矩阵。从用户相似度矩阵可以得出与目标用户 u 相似度最大的前 K 个用户，把它们称为 u 的邻居集，表示为 $N(u)$。则利用 UserCF 算法预测用户 u 对项目 i 的评分就可以表示为用户 u 的邻居集 $N(u)$对项目 i 的真实评分的加权平均值，如式(2)所示：

$$\hat{r}_{ui}=\bar{r}_u+\frac{\sum_{v\in N(u)}sim(u,v)(r_{vi}-\bar{r}_v)}{\sum_{v\in N(u)}sim(u,v)} \tag{2}$$

其中，$sim(u,v)$即为通过式(1)计算得到的用户 u 和 v 之间的相似度。引入用户 u 和用户 v 的平均评分 $\bar{r}_u$ 和 $\bar{r}_v$ 是为了消除不同用户之间评分尺度的差异。

2.2 基于项目的协同过滤算法

与基于用户的协同过滤算法不同，基于项目的协同过滤算法基于这样一个假设：一个用户会喜欢与他之前喜欢的东西相类似的东西，因此它是基于目标项目的邻居集进行推荐的。其核心是计算项目之间的相似度，并利用与目标项目相似度最大的前 K 个项目构造其邻居集。ItemCF 算法的步骤描述如算法 2 所示。

算法 2 基于项目的协同过滤算法（ItemCF）

步骤 1：计算项目之间的相似度，构造项目相似度矩阵；

步骤 2：选择与目标项目相似度最大的前 K 个项目构造其邻居集；

步骤 3：利用邻居集的已知评分的加权和来预测目标项目的未知评分。

项目之间相似度的计算是基于共同评分的用户集进行的。假设 U 表示对项目 i 和项目 j 都有评分的用户集，$\bar{r}_i$ 和 $\bar{r}_j$ 分别表示项目 i 和 j 在 U 上的平均评分，r_{ui} 和 r_{uj} 表示用户 u 对项目 i 和 j 的真实评分。则利用皮尔逊相关系数计算项目 i 和 j 之间相似度，如式(3)所示：

$$sim(i,j)=\frac{\sum_{u\in U}(r_{ui}-\bar{r}_i)(r_{uj}-\bar{r}_j)}{\sqrt{\sum_{u\in U}(r_{ui}-\bar{r}_i)^2}\sqrt{\sum_{u\in U}(r_{uj}-\bar{r}_j)^2}} \tag{3}$$

与目标项目 i 相似度最大的且被用户 u 评过分的前 K 个项目构成它的邻居集，表示为 $N(i)$。那么，利用基于项目的协同过滤算法计算用户 u 对项目 i 的预测评分就可以表示为用户 u 对项目 i 的邻居集 $N(i)$的真实评分的加权平均值，并且通过基线估计来调整用户和项目的影响，如式(4)[8]所示：

$$\hat{r}_{ui}=b_{ui}+\frac{\sum_{j\in N(i)}sim(i,j)(r_{uj}-b_{uj})}{\sum_{j\in N(i)}sim(i,j)} \tag{4}$$

因为推荐系统中的评分数据表现出强烈的用户和项目效应，所以式(4)通过加入基线估计来调整用户和项目的影响。其中，b_{ui} 和 b_{uj} 分别表示用户 u 对项目 i 和 j 的基线估计，$sim(i,j)$为利用式(3)计算得到的项目 i 和 j 之间的相似度。这个公式的物理含义是用户对目标项目的未知评分可以利用用户对目标项目邻居集的已知评分的加权和来表示。关于基线估计的详细介绍请参考文献[8]。

3 嵌入 LDA 主题模型的协同过滤推荐算法

3.1 主题模型概念

主题模型是对文字中隐含主题的一种建模方法，它可以发现文档中的一些潜在主题。由于不同的单词可能隐含了相同的主题，因此比较两篇文档的相似性不再是简单地比较共同出现的单词数，而是比较两篇文档中单词所隐含的主题之间的相似性。主题表现为一系列语义相关的词语，它可以理解为词汇表上词语的条件概率分布，即与主题关系越密切的词语，它的条件概率越大；反之则越小。从不同的层次来看，一篇文章以不同的概率包含了不同的主题，而一个主题又以不同的概率包含了很多不同的单词。

主题模型的发展最早可以追溯到 20 世纪 70 年代提出的向量空间模型[9]。它将文本内容映射到向量空间以方便运算，用向量空间中的向量相似度表示文档中语义的相似度。后来，学者又提出隐马尔科夫模型、最大熵模型等统计语言模型，直到近年来流行的潜在语义分析，这些都为主题模型的发展奠定了基础。

3.2 LDA 主题模型

Blei 等人在文献[10]中提出的潜在狄利克雷分配(Latent Dirichlet Allocation，LDA)是主题模型中最经典的一种算法。它是一个生成式模型，可以看作是一个文档的产生过程。它认为一篇文章中的每个词都是通过“以一定概率选择了某个主题，并从这个主题中以一定概率选择某个词语”这样一个过程得到的。或者说是，主题是词语上的多项分布，文档是主题上的多项分布。因此，如果要生成一篇文档，它里面每个词语出现的概率如式(5)所示。

$$p(\text{词语}|\text{文档})=\sum_{\text{主题}} p(\text{词语}|\text{主题})\times p(\text{主题}|\text{文档}) \quad (5)$$

LDA 模型的输入是大规模的文档集合，经过 LDA 模型的训练之后得到两个分布，分别为文档在主题上的概率分布和主题在单词上的概率分布。

将 LDA 模型的生成过程用概率图模型描述，如图 2 所示。图中 α 为 Dirichlet 分布的超参数，用于生成一个主题 θ 向量；β 为各个主题对应的单词概率分布矩阵 $p(w|z)$；θ 是一个主题向量，向量的每一列表示文档中每个主题被选择的概率。从图 2 可以看出，LDA 是一个 3 层的贝叶斯模型。这种方法首先选定一个主题向量 θ，确定每个主题被选择的概率；然后在生成每个单词时，从主题分布向量 θ 中选择一个主题 z，按主题 z 的单词概率分布生成一个单词。

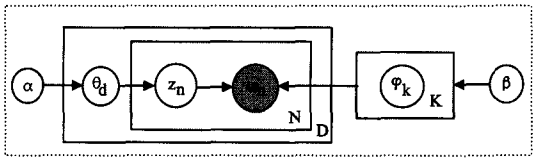

图 2 LDA 的概率图模型表示

因此，LDA 模型生成一篇文档 d 的步骤为(假设语料库中有 D 篇文档、K 个主题)：

(1)采样 $N\sim$Poisson(ε)；

(2)采样 $\theta_d\sim Dir(\alpha)$，其中 $d\in\{1,\cdots,D\}$；

(3)采样 $\varphi_k\sim Dir(\beta)$，其中 $k\in\{1,\cdots,K\}$；

(4)对于文档 d 中的第 n 个单词 w_n，$n\in\{1,\cdots,N_d\}$：

a)选择隐藏主题 $z_n\sim Mult(\theta_d)$；

b)生成单词 $w_n\sim Mult(\varphi_z)$。

在给定先验参数 α 和 β 的条件下，LDA 的联合概率分布如式(6)所示：

$$p(\theta,\varphi,\boldsymbol{z},\boldsymbol{w}|\alpha,\beta)=p(\varphi|\beta)\prod_{n=1}^{N}p(w_n|\varphi_{zn})p(z_n|\theta)p(\theta|\alpha) \quad (6)$$

根据式(6)可以估计出两个参数 θ 和 φ 的值，本文实验用到的参数是文档-主题概率分布矩阵 θ。LDA 模型参数的估计有多种方法，如变分贝叶斯推断、吉布斯采样以及最大似然估计等。本文实验采用变分贝叶斯推断算法对模型参数进行估计。关于变分贝叶斯推断算法的详细介绍请参考文献[10]。

3.3 嵌入 LDA 主题模型的协同过滤推荐算法

本文实验使用美国 GroupLens 研究项目组提供的 MovieLens数据集[11]。数据集中不仅包含用户对电影的评分记录，还包括用户和电影的标签信息，这些标签信息是领域专家为对象预先指定的。标签是一种描述事物多重属性的分类工具，每个用户和每个电影都有一个标签集来描述它的特征属性。MovieLens 数据集中用户的标签包括用户的性别、年龄、职业等，描述了用户的人口统计学特征；电影的标签为电影的类型，描述了该电影可能属于的类别。传统协同过滤算法基于评分计算相似度，因此存在稀疏性问题。然而，不同标签可能隐藏了相同主题，因此利用用户和电影标签集中隐藏的潜在主题也可以很好地度量用户和电影相似度。因此，本文针对传统协同过滤算法存在的稀疏性问题，提出一种结合文档-主题概率分布矩阵和评分矩阵来共同度量相似度的新算法，即嵌入 LDA 主题模型的协同过滤算法(ULR-CF)。这里的文档-主题分布是在文档-标签分布之上利用 LDA 主题模型计算得到的。

3.3.1 利用文档-主题分布计算相似性

把每个用户和电影分别看成一个文档，用户和电影的标签看成文档中的单词，从中挖掘潜在的主题信息。即在文档-标签分布上利用 LDA 主题模型进行建模，得到文档-主题概率分布，进而利用这个分布来计算文档相似性(即用户相似性和电影相似性)。下面给出一个利用 LDA 主题模型从文档-标签分布诱导出文档-主题分布的例子。我们抽取 6 个电影(i_1-i_6)的文档-标签分布，如图 3(a)所示。在文档-标签分布上利用 LDA 主题模型进行建模时设置主题个数为 3，从而得到文档-主题概率分布矩阵 Θ，如图 3(b)所示。

$$\begin{cases} i_1: \text{Horror Action Children Comedy} \\ i_2: \text{Horror Fantasy Crime Action} \\ i_3: \text{Romance Fantasy Drama Crime} \\ i_4: \text{War Adventure Fantasy Crime} \\ i_5: \text{Mystery Thrill Drama Children} \\ i_6: \text{Thrill Adventure Drama Action} \end{cases}$$

(a)文档-标签分布

$$\Theta=\begin{bmatrix} 0.8182 & 0.0909 & 0.0909 \\ 0.4545 & 0.4545 & 0.0909 \\ 0.2727 & 0.6364 & 0.0909 \\ 0.2727 & 0.4545 & 0.2727 \\ 0.2727 & 0.0909 & 0.6364 \\ 0.2727 & 0.2727 & 0.4545 \end{bmatrix}$$

(b)文档-主题分布

图 3 文档-标签分布与文档-主题分布示例图

从文档-主题分布 Θ 可以看出，Movie1 与 Movie2 有很高的相似性，因为它们第一列的值是所有 3 列值中最大的，说明它们属于第一个主题的概率是最大的；同样地，Movie3 与 Movie4 是相似的，它们属于第二个主题的概率最大；Movie5 和 Movie6 是相似的，它们属于第三个主题的概率最大。

因此，在这个概率分布 Θ 上计算文档之间的相似度就变得非常容易。利用 LDA 主题模型实现了把难以处理的文本信息转化为方便计算的数字向量，每篇文档表示为一个归一化的数字向量，向量中的每个元素描述了这个文档包含不同主题的概率大小。本文采用余弦相似性方法计算向量之间的相似度。假设两个电影 i 和 j 的文档-标签分布在利用 LDA 主题模型进行建模后得到的两个文档-主题概率分布向量分别为 t_i 和 t_j，那么基于文档-主题概率分布，利用余弦相似性方法计算电影 i 和 j 之间的相似度可表示为式(7)：

$$sim(i,j)=\cos(t_i,t_j)=\frac{t_i \cdot t_j}{\| t_i \|_2 \cdot \| t_j \|_2} \tag{7}$$

其中，$\cdot$ 表示两个向量之间的内积。基于文档-主题概率分布计算两个用户之间的相似度可以通过相同的方法得到。

3.3.2 结合文档-主题分布和评分矩阵共同计算相似度

基于用户-项目评分矩阵可以计算用户和项目的相似度，基于用户和项目的文档-主题分布同样可以计算出它们各自之间的相似度。因此，把二者结合起来通过联合学习的方式来共同度量相似度可得到更加准确的相似度，从而提高推荐结果的准确性。以计算项目 i 和 j 的相似度为例，本文提出的 ULR-CF 算法的核心思想为：

(1)基于评分矩阵计算项目 i 和 j 的相似度，记为 $simR(i,j)$；

(2)基于文档-主题概率分布矩阵计算项目 i 和 j 的相似度，记为 $simL(i,j)$；

(3)既考虑利用评分矩阵计算得到的相似度 $simR(i,j)$，又考虑利用文档-主题概率分布计算得到的相似度 $simL(i,j)$，在总的相似度 $uni\_sim(i,j)$ 中，二者所占的比重通过参数 λ 来调节，如式(8)所示：

$$uni\_sim(i,j)=\lambda simR(i,j)+(1-\lambda)simL(i,j) \tag{8}$$

类似地，用户 u 和 v 之间的相似度 $uni\_sim(u,v)$ 也可以通过类似的方法计算得到。

3.3.3 产生推荐

在结合评分矩阵和文档-主题概率分布矩阵共同计算得到总的相似度之后，就可利用基于用户的协同过滤算法和基于项目的协同过滤算法来产生推荐。将式(8)应用到模型(2)和模型(4)中，发展出如下两种改进的协同过滤算法。

(1)嵌入 LDA 主题模型的基于用户的协同过滤算法(记为 ULR-UserCF)，其评分预测形式如式(9)所示：

$$\hat{r}_{ui}=\bar{r}_u+\frac{\sum_{v\in N(u)} uni\_sim(u,v)(r_{vi}-\bar{r}_v)}{\sum_{v\in N(u)} uni\_sim(u,v)} \tag{9}$$

(2)嵌入 LDA 主题模型的基于项目的协同过滤算法(记为 ULR-ItemCF)，其评分预测形式如式(10)所示：

$$\hat{r}_{ui}=b_{ui}+\frac{\sum_{j\in N(i)} uni\_sim(i,j)(r_{uj}-b_{uj})}{\sum_{j\in N(i)} uni\_sim(i,j)} \tag{10}$$

实验结果表明，本文提出的 ULR-CF 算法可有效缓解数据稀疏性问题，同时可显著提高推荐系统的推荐准确性。

4 实验结果及分析

4.1 数据集与评价准则

本文实验使用的数据集来自美国 GroupLens 研究项目组提供的 MovieLens 数据集[11]。数据集中共有 943 个用户、1682 个电影、10 万个评分记录以及大量的标签信息。本文从数据集中抽取 80%作为训练集来计算相似度，20%作为测试集来检测算法的效果，使用 5 折叠交叉平均实验结果来减少误差。

本文使用的评价准则为平均绝对误差(Mean Absolute Error，MAE)，它是推荐领域中最常用的评价准则之一。*MAE* 评价准则如式(11)所示：

$$MAE=\frac{\sum_{u,i}|\hat{r}_{ui}-r_{ui}|}{N} \tag{11}$$

式中，N 为测试的总数量，$\hat{r}_{ui}$ 表示用户 u 对项目 i 的预测评分，r_{ui} 表示用户 u 对项目 i 的真实评分。平均绝对误差 *MAE* 的物理含义为取所有预测误差的平均值。显然，*MAE* 的值越小越好。

4.2 实验结果与分析

为验证本文提出的 ULR-CF 算法的有效性，设计如下两组对比实验。

4.2.1 ItemCF 算法与 ULR-ItemCF 算法

对于 ULR-ItemCF 算法，固定邻域大小 $K=20$，调节式(8)中的参数 λ。由式(8)可见，当参数 $\lambda=0$ 时，相似性退化为仅基于文档-主题分布的相似性计算；当参数 $\lambda=1$ 时，相似性退化为仅基于评分矩阵的相似性计算。在测试集上的实验结果如图 4 所示，它描述了 ULR-ItemCF 算法在不同 λ 值下的 *MAE* 值。

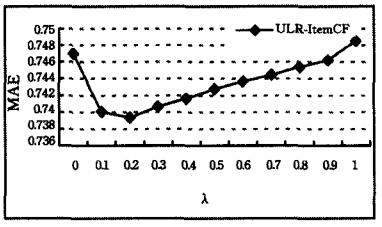

图 4 ULR-ItemCF 算法的 *MAE* 值随参数 λ 的变化情况

从图 4 可以看出，当 $\lambda=0.2$ 时，算法的 *MAE* 取得最小值。因此，ULR-ItemCF 算法中参数 λ 的最优值为 0.2。固定 $\lambda=0.2$，使邻域大小 K 从 10 变化到 30，比较传统 ItemCF 算法与本文提出的 ULR-ItemCF 算法的性能。表 1 给出了这两种算法在不同 K 值情况下的详细 *MAE* 值。由表 1 的 improvement 一列可知，本文提出的 ULR-ItemCF 算法相较于传统 ItemCF 算法在预测精度上有了很大提升，最多可提高 1.7%，这是因为在电影标签集上很容易发现潜在主题信息，而且把电影的标签进行主题划分也是很好理解的。例如，惊悚类型和恐怖类型的电影很可能被划分到同一个主题。

为了更清晰地观察二者的变化趋势，把实验数据用折线图的形式呈现，如图 5 所示。从图 5 可以看出，本文提出的结合评分矩阵和文档-主题概率分布矩阵共同度量相似度的 ULR-ItemCF 算法比传统的仅利用评分矩阵度量相似度的 ItemCF 算法具有更小的预测误差，性能得到明显优化。从图

5 也可清楚地看到，两种算法的 *MAE* 值随邻域大小 K 值的增加均呈下降的趋势，即推荐系统在进行推荐时所参考的邻居数目越多，得到的推荐结果越准确。

表 1　ItemCF 算法与 ULR-ItemCF 算法的 *MAE* 值比较

K	ItemCF	ULR-ItemCF	improvement
10	0.7692	0.7521	0.0171
12	0.7623	0.7476	0.0147
14	0.7574	0.7444	0.0130
16	0.7538	0.7420	0.0118
18	0.7509	0.7408	0.0101
20	0.7485	0.7393	0.0092
22	0.7468	0.7386	0.0082
24	0.7452	0.7377	0.0075
26	0.7439	0.7371	0.0068
28	0.7428	0.7367	0.0061
30	0.7419	0.7364	0.0055

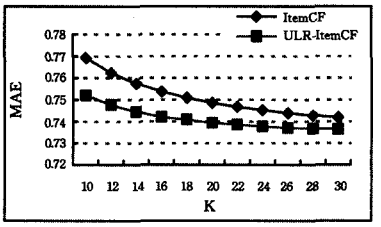

图 5　ItemCF 算法与 ULR-ItemCF 算法的 *MAE* 值随邻域大小 K 的变化情况

4.2.2　UserCF 算法与 ULR-UserCF 算法

对于 ULR-UserCF 算法，固定邻域大小 $K=20$，调节参数 λ。在测试集上的实验结果如图 6 所示，它描述了 ULR-UserCF 算法在不同 λ 值下的 *MAE* 值。由图 6 可知，当 $\lambda=0.6$ 时，算法的 MAE 取得最小值。因此，ULR-UserCF 算法中参数 λ 的最优值为 0.6。固定 $\lambda=0.6$，使邻域大小 K 从 10 变化到 30，比较传统 UserCF 算法与本文提出的嵌入 LDA 主题模型的 ULR-UserCF 算法的性能。表 2 给出了两种算法在不同 K 值情况下的详细 *MAE* 值。

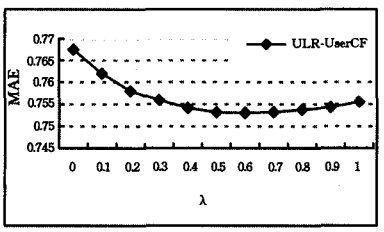

图 6　ULR-UserCF 算法的 *MAE* 值随参数 λ 的变化情况

表 2　UserCF 算法与 ULR-UserCF 算法的 *MAE* 值比较

K	UserCF	ULR-UserCF	improvement
10	0.7736	0.7697	0.0039
12	0.7677	0.7641	0.0036
14	0.7636	0.7598	0.0038
16	0.7601	0.7569	0.0032
18	0.7576	0.7547	0.0029
20	0.7555	0.7530	0.0025
22	0.7540	0.7517	0.0023
24	0.7530	0.7509	0.0021
26	0.7520	0.7500	0.0020
28	0.7511	0.7495	0.0016
30	0.7505	0.7488	0.0017

从实验结果可以看出，本文提出的 ULR-UserCF 算法可以取得比传统 UserCF 算法更小的预测误差，有效解决了协同过滤算法所面临的数据稀疏性和推荐准确性问题。与表 2 相对应的折线图如图 7 所示。从图 7 也可清晰地看出，ULR-UserCF 算法的性能要明显优于 UserCF 算法的性能，且两种算法的 *MAE* 值随邻域大小 K 的增加均呈下降趋势。对比 4.2.1 节的实验与 4.2.2 节的实验可看出，ULR-UserCF 算法相对于现有 UserCF 算法的改进程度不如 ULR-ItemCF 算法相对于现有 ItemCF 算法的改进程度大，这是因为对于用户的标签信息，我们较难发现其中隐含的潜在主题；而对于电影的标签信息，挖掘其中隐含的潜在主题相对容易，并且得到的主题信息也较为精确。

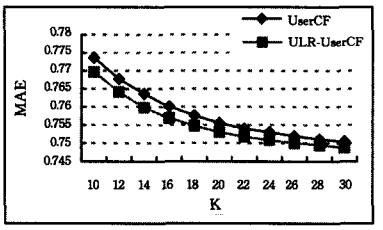

图 7　UserCF 算法与 ULR-UserCF 算法的 *MAE* 值随邻域大小 K 的变化情况

从以上两组实验结果可以看出，本文提出的结合评分和文档-主题概率分布共同度量相似度的 ULR-CF 算法可有效解决协同过滤算法所面临的数据稀疏性问题，并显著提高推荐系统的推荐准确性。

结束语　本文提出的嵌入 LDA 主题模型的协同过滤算法(ULR-CF)与传统协同过滤算法相比，融入了标签集中隐含的潜在主题信息，有效缓解了数据稀疏性问题，并使推荐系统的准确性得到显著提高。但推荐系统还存在很多挑战，如冷启动问题、可扩展性问题、安全性问题等。因此，本文以后的工作将致力于研究更多的算法来解决推荐系统所面临的各种挑战，以提高推荐系统的准确性。

参 考 文 献

[1] Rich E. User modeling via stereotypes [J]. Cognitive Science, 1979, 3(4): 329-354

[2] Nakamura A, Abe N. Collaborative Filtering Using Weighted Majority Prediction Algorithms [C] // Proceedings of the 15th International Conference on Machine Learning, 1998. San Francisco: Morgan Kaufmann, 1998: 395-403

[3] Linden G, Smith B, York J. Amazon. com recommendations: Item-to-item collaborative filtering [J]. Internet Computing, 2003, 7(1): 76-80

[4] Ji H, Li J, Ren C, et al. Hybrid collaborative filtering model for improved recommendation [C] // Service Operations and Logistics, and Informatics, 2013. Dongguan: IEEE, 2013: 142-145

[5] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems [J]. Computer, 2009, 42(8): 30-37

[6] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms [C] // Proceedings of 10th International Conference on World Wide Web, 2001. New York: ACM, 2001: 285-295

[7] Hu Y, Koren Y, Volinsky C. Collaborative filtering for implicit feedback datasets [C] // Proceedings of the 8th International Conference on Data Mining, 2008. Pisa: IEEE, 2008: 263-272

(下转第 79 页)

发和不确定性等特征，这与人脑思维过程中的某些认知活动类似，其参数也可进行学习训练[12-14]。本文方法通过把 Petri 网中的库所和变迁分别用属性粒及定性映射来模拟，使 Petri 网以属性粒计算的方式在知识表示、知识推理、学习模式和记忆模式等方面上初步表现出一个认知系统所需要具备的一些基本元素特征。

因此，与其他方法相比，扩展模糊 Petri 网在知识推理方面具有以下优势：(1)带基准变换算子的模糊 Petri 网变迁节点有利于权值参数的学习与调整；(2)定性映射的记忆模式能使模糊 Petri 网较好地克服过于依赖专家或单纯利用专家知识和经验的局限性，这里权值是由定性基准来确定的，且定性基准可不事先确定，而由学习获得；(3)相对人工神经网络，扩展模糊 Petri 网转移结点的每一个输入输出的值都有确定的含义，而在一般神经网络知识表示中只关注输入层与输出层的神经元之值，并不关注它的含义，而且对多层信息必须经过预处理才能进行推理，这里只须确定研究对象和它的层次属性，然后在定性基准下经定性映射进行属性整合操作推理，不仅能模拟数值推理情况，同时也能够模拟非数值推理情况；(4)这种扩充比较容易推广到以下应用领域：基于特征抽取的识别、判断、分类、评估、规划和决策领域，以及需表示事件状况、属性、属性间推理和动作间关系的领域等。

结束语 基于定性映射的扩展模糊 Petri 网可以有效扩充网系统的表达能力。在给出合理结构和适当推理算法的情况下，这种扩充最可能在基于特征抽取的识别与判断、需表示事件属性与属性间推理和动作间关系的领域得到应用。此外，扩展模糊 Petri 网中的知识表示和推理方法由于易于表达模糊产生式规则，权值可由定性基准来确定，且定性基准可不事先确定，而由学习获得，因此对其他知识系统的信息处理具有很好的借鉴意义。文中给出的网模型的扩充是初步的，相关网络结构、活性分析、学习方法和学习能力等问题将是下一步需要认真进行研究的内容。

参 考 文 献

[1] Cao Z C, Zhao H D, Wang Y J. Releasing control policy for semiconductor wafer fabrication based on fuzzy Petri nets-reasoning [J]. Acta Electronica Sinica, 2011, 39(7): 1545-1550 (in Chinese)
曹政才，赵会丹，王永吉. 基于模糊 Petri 网推理的半导体生产线投料控制策略[J]. 电子学报，2011，39(7)：1545-1550

[2] Meng X G, Yan H S. Products demand forecasting in knowledgeable manufacturing systems based on attributes fuzzy Petri nets[J]. Systems Engineering-theory & Practice, 2012, 32(4): 790-798(in Chinese)
孟宪刚，严洪森. 基于多属性模糊 Petri 网的知识化制造系统产品需求预测[J]. 系统工程理论与实践，2012，32(4)：790-798

[3] Feng J L. Attribute network computing based on qualitative mapping and its applications in pattern recognition[J]. Journal of Intelligent & Fuzzy Systems, 2008, 19(2): 1-16

[4] Chen S Y. Philosophical foundation of variable fuzzy sets theory [J]. Journal of Dalian University of Technology (Social Sciences), 2005, 26(1): 53-57(in Chinese)
陈守煜. 可变模糊集理论的哲学基础[J]. 大连理工大学学报(社会科学)，2005，26(1)：53-57

[5] Li D Y, Liu C Y, Du Y, et al. Artificial intelligence with uncertainty[J]. Journal of Software, 2004, 15(11): 1583-1594(in Chinese)
李德毅，刘常昱，杜鹢，等. 不确定性人工智能[J]. 软件学报，2004，15(11)：1583-1594

[6] Cai W, Shi Y. Extenics: its significance in science and prospects in application[J]. Journal of Harbin Institute of Technology, 2006, 38(7): 1079-1083(in Chinese)
蔡文，石勇. 可拓学的科学意义与未来发展[J]. 哈尔滨工业大学学报，2006，38(7)：1079-1083

[7] Feng J L. Thought, intelligence and attribute theory method[J]. Journal of Guangxi Normal University (Natural Science Edition), 1997, 15(3): 1-6(in Chinese)
冯嘉礼. 思维、智能与属性论方法[J]. 广西师范大学学报(自然科学版)，1997，15(3)：1-6

[8] Smarandache F. A Unifying field in logics: Neutrosophic logic, Neutrosophic Probability and Statistics[M]. America: Xiquan Publishing Hours, 2003

[9] Zadeh L A. Fuzzy Sets[J]. Information and Control, 1965, 8(3): 338-353

[10] Acampora G, Loia V. On the temporal granularity in fuzzy cognitive maps[J]. IEEE Transactions on Fuzzy Systems, 2011, 19(6): 1040-1057

[11] Gao M M, Wu Z M. Fuzzy reasoning Petri net and its application to fault diagnosis[J]. Acta Automatica Sinica, 2000, 26(5): 677-680(in Chinese)
高梅梅，吴智铭. 模糊推理 Petri 网及其在故障诊断中的应用[J]. 自然化学报，2000，26(5)：677-680

[12] Shenvictor R L, Chang Yue-shan, Juang T T-Y, et al. Supervised and Unsupervised Learning by Using Petri Nets [J]. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 2010, 40(2): 363-375

[13] Feng Liang-bing, Obayashi M, Kuremoto T, et al. A Learning Fuzzy Petri Net Model[J]. IEEE Transactions on Electrical and Electronic Engineering, 2012, 7(3): 274-282

[14] Liu Hu-chen, Lin Qing-lian, Mao Ling-xiang, et al. Dynamic Adaptive Fuzzy Petri Nets for Knowledge Representation and Reasoning[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2013, 43(6): 1399-1410

(上接第 61 页)

[8] Koren Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model[C]//Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2008. New York: ACM, 2008: 426-434

[9] Salton G, Wong A, Yang C S. A Vector Space Model for Automatic Indexing[J]. Communications of the ACM, 1975, 18(10): 613-620

[10] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. Journal of Machine Learning Research, 2003, 3: 601-608

[11] Riedl J, Konstan J. Movielens dataset[EB/OL]. (1998-10-19) [2008-07]. http://www.grouplens.org/data