文章编号:1001 - 9081(2011)05 - 01382 - 05
doi:10.3724/SP.J.1087.2011.01382

基于用户和项目因子分析的混合协同推荐算法

赵宏霞[1],王新海[1],杨皎平[2]
(1.辽宁工程技术大学 营销管理学院,辽宁 葫芦岛 125105;
2.渤海大学 管理学院,辽宁 锦州 121013)
(zhx9878@tom.com)

摘　要:为解决协同过滤推荐(CFR)算法中的数据量过大和数据稀疏性的问题,采用因子分析的方法对数据降维,并使用回归分析方法预测待评估值,既减少了数据量又最大限度保留了信息。该算法首先,采用因子分析的方法将用户和项目降维为若干用户因子和若干项目因子;然后,以目标用户为因变量,以用户因子为自变量建立一个回归模型,并且以待评价项目为因变量,以项目因子为自变量建立另一个回归模型,进而得到目标用户在待评项目上的两个预测值;最后,通过两者的加权得到最终的预测值。实验仿真证实了算法的可行性和有效性。实验结果表明,该算法比基于项目的协同过滤推荐算法在精确度上有所提高。

关键词:推荐系统;协同过滤;因子分析
中图分类号: TP311.13　　**文献标志码**:A

Mixed collaborative recommendation algorithm based on factor analysis of user and item

ZHAO Hong-xia[1], WANG Xin-hai[1], YANG Jiao-ping[2]
(1. *School of Marketing Management, Liaoning Technical University, Huludao Liaoning 125105, China*;
2. *College of Management, Bohai University, Jinzhou Liaoning 121013, China*)

Abstract: In order to solve the problems of data overload and data sparsity in Collaborative Filtering Recommendation (CFR) algorithm, the method of factor analysis was adopted to reduce the dimension of the data, and regression analysis was used to forecast the value that needs to be evaluated. Through these two methods, it not only reduces the amount of data but also maximizes the information retained. The ideas of the algorithm are as follows: first of all, the algorithm reduces the dimensions of user and item vector by use of factor analysis and some representative users and item factors could be got. And then, two regression models were established, with target users and the evaluated items as the dependent variables respectively, and the user factors and item factors as the independent variables respectively, which two predictive values of the evaluated items were achieved. Finally, the final predictive value was achieved weighted by the two. By experimental simulation, the algorithm is demonstrated effective and feasible. Furthermore, the results show that the accuracy of algorithm proposed here has somewhat increased compared with that of the collaborative filtering recommendation algorithm based on item.

Key words: recommendation system; collaborative filtering; factor analysis

0　引言

随着信息技术的飞速发展,开展电子商务的企业为了提高系统的销售能力,重建客户关系,对推荐技术倍加重视和关注,目前应用最广泛的推荐技术之一就是协同过滤推荐(Collaborative Filtering Recommendation, CFR)算法[1]。CFR算法主要包括基于模型的算法和基于记忆的算法,其中基于记忆的算法以其简单性和高推荐质量成为比较成功的一类推荐算法。

基于记忆的算法,又可以分为基于用户的算法[2-3]和基于项目的算法[4-5]。基于用户的算法根据用户的相似性来产生推荐,基于项目的算法根据项目的相似性来产生推荐。两种算法均只考虑了单方面的影响,忽略了用户与项目的联系,为此,汪静等人[6]提出了两种结合的算法,该算法综合利用了两方面信息,大大提高了推荐质量。

但是以上算法都面临如下的挑战[7]:1)Web环境下数据量巨大,需要推荐算法能够在尽可能短的时间内做出响应;2)数据的稀疏性,这看起来与数据量巨大是矛盾的,但是相对于系统中为数众多的用户和待推荐的产品,能够利用的表示用户兴趣的信息实际上是非常稀疏甚至有限的。

针对这两个困难,许多研究者提出了基于聚类的方法来解决,如唐晓波等人[8]和李涛等人[9]提出了基于用户聚类的推荐方法,张海鹏等人[10]提出了基于项目分类的推荐算法。这些聚类算法,主要基于距离的思想对用户或项目进行了分类。本文认为在现实的电子商务中,决定用户(或项目)类别的往往是一些隐式的原因,用户的购买或评价行为只是这种隐式原因的外在表现。如侯翠琴等人[11]认为项目集是若干隐变量集的表现,从而利用隐变量集对项目集进行了降维处理,而张亮等人[12]则认为存在决定用户类别和项目类别的两个隐变量。同时这些算法的分类属于“硬分类”,认为某一用

收稿日期:2010 - 09 - 25;**修回日期**:2010 - 11 - 24。　**基金项目**:国家自然科学基金资助项目(70971059);教育部博士点基金资助项目(200801470004);辽宁省自然科学基金资助项目(20082185)。

作者简介:赵宏霞(1978 -),女(蒙古族),内蒙古赤峰人,副教授,博士,主要研究方向:商务智能、网络营销;　王新海(1972 -),男,山西大同人,副教授,博士,主要研究方向:商务智能;　杨皎平(1980 -),男,山西临汾人,副教授,博士,主要研究方向:系统工程。

户(或项目)只属于某一类,然而在实际当中,一个用户(或项目)往往具有若干类的特征,这就需要进行“软分类”。

本文在以上文献的基础上,提出基于用户和项目因子分析的混合推荐(Mixed Collaborative Recommendation based on Factor Analysis of User and Item, MCR-F)算法。该算法首先采用因子分析的方法将用户和项目降维为若干用户因子(可理解为如文献[13]所述的求实型用户、求美型用户、求名型用户等)和若干项目因子(如时尚型商品、实用性商品、廉价型商品);然后分别用目标用户和待评价项目为因变量,以用户因子和项目因子为自变量做出两个回归模型,进而得到目标用户在待评项目上的两个预测值;最后通过两者的加权得到最终的预测值。

1　相关工作

1.1　问题描述

在基于协同过滤推荐算法的推荐系统中,用户评分数据库中包括 m 个用户的集合 $\boldsymbol{U}=\{\boldsymbol{u}_1,\boldsymbol{u}_2,\cdots,\boldsymbol{u}_m\}$ 和 n 个项目的集合 $\boldsymbol{I}=\{\boldsymbol{I}_1,\boldsymbol{I}_2,\cdots,\boldsymbol{I}_n\}$。用户对项目的评分数据可以采用一个 $m\times n$ 阶的用户-项目评分矩阵 $\boldsymbol{R}=(r_{ij})_{m\times n}$ 来表示:

	$\boldsymbol{I}_1$	…	$\boldsymbol{I}_t$	…	$\boldsymbol{I}_n$
$\boldsymbol{u}_1$	r_{11}	…	r_{1t}	…	r_{1n}
⋮	⋮		⋮		⋮
$\boldsymbol{u}_s$	r_{s1}	…	r_{st}	…	r_{sn}
⋮	⋮		⋮		⋮
$\boldsymbol{u}_m$	r_{m1}	…	r_{mt}	…	r_{mn}

评分表示用户对项目感兴趣的程度,评分的级别越高说明用户越感兴趣。

1.2　基于相似性的邻居选择

1.2.1　用户的相似性

用户之间的相似性表示用户兴趣爱好的相似程度,选择用户 a 和用户 b 的共同评分数据来计算用户 a 和用户 b 之间的相似性 $sim(a,b)$,表示如下:

$$sim(a,b)=\frac{\sum_{i\in\Pi ab}(r_{ai}-\bar{r}_a)(r_{bi}-\bar{r}_b)}{\sqrt{\sum_{i\in\Pi_{ab}}(r_{ai}-\bar{r}_a)^2}\sqrt{\sum_{i\in\Pi_{ab}}(r_{bi}-\bar{r}_b)^2}}\tag{1}$$

其中:$\Pi_{ab}=I(a)\cap I(b)$,$I(a)$ 表示用户 a 评分的项目集合,$I(b)$ 表示用户 b 评分的项目集合;i 属于 $I(a)$ 和 $I(b)$ 的交集,表示用户 a 和用户 b 共同评分的项目集合;$\bar{r}_a$ 和 $\bar{r}_b$ 表示用户的评分均值。

$$\bar{r}_a=\frac{\sum_{i\in\Pi_{ab}}r_{ai}}{|\Pi_{ab}|}$$

$$\bar{r}_b=\frac{\sum_{i\in\Pi_{ab}}r_{bi}}{|\Pi_{ab}|}$$

其中 $|\Pi_{ab}|=|I(a)\cap I(b)|$ 表示用户 a 和用户 b 共同评分的项目个数。

1.2.2　项目的相似性

项目之间的相似性表示用户对若干项目同时感兴趣的程度。选择用户群体对项目 i 和项目 j 的共同评分数据来计算项目 i 和项目 j 之间的相似性 $sim(i,j)$,表示如下:

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

其中:$\Pi_{ij}=U(i)\cap U(j)$,$U(i)$ 表示对项目 i 评分的用户集合,$U(j)$ 表示对项目 j 评分的用户集合;u 属于 $U(i)$ 和 $U(j)$ 的交集,表示对项目 i 和项目 j 共同评分的用户;$\bar{r}_i$ 和 $\bar{r}_j$ 表示项目的评分均值。

$$\bar{r}_i=\frac{\sum_{u\in\Pi_{ij}}r_{ui}}{|\Pi_{ij}|}$$

$$\bar{r}_j=\frac{\sum_{u\in\Pi_{ij}}r_{uj}}{|\Pi_{ij}|}$$

其中 $|\Pi_{ij}|=|U(i)\cap U(j)|$ 表示对项目 i 和项目 j 共同评分的用户个数。

对目标用户 g 的最近邻居集合 $Nei(g)$,以及待评分项目 s 的最近邻居项目集合 $Nei(s)$,既可以选择相似性排在前几位的用户和项目,也可以选择相似性大于某个阈值的用户和项目。

1.3　评分预测

1.3.1　基于用户的评分预测

根据用户 g 的邻居用户集合 $Nei(g)$,预测目标用户 g 对 s 的评分。预测公式如式(3)所示。

$$\hat{r}_{gs}^U=\bar{r}_g+\frac{\sum_{a\in Nei(g)}sim(g,a)\times(r_{as}-\bar{r}_a)}{\sum_{a\in Nei(g)}sim(g,a)}\tag{3}$$

1.3.2　项目的评分预测

根据 s 的邻居项目集合 $Nei(s)$,预测目标用户 g 对 s 的评分。预测公式如式(4)所示。

$$\hat{r}_{gs}^I=\bar{r}_s+\frac{\sum_{i\in Nei(s)}sim(s,i)\times(r_{gi}-\bar{r}_i)}{\sum_{i\in Nei(s)}sim(s,i)}\tag{4}$$

文献[6]在上述基础上,给出了基于用户和项目的加权预测公式:

$$\hat{r}_{gs}=\lambda_U\hat{r}_{gs}^U+\lambda_I\hat{r}_{gs}^I;\lambda_U+\lambda_I=1\tag{5}$$

其中:

$$\lambda_U=\frac{\sum_{a\in Nei(g)}sim^2(g,a)}{\sum_{a\in Nei(g)}sim^2(g,a)+\sum_{i\in Nei(s)}sim^2(s,i)}$$

$$\lambda_I=\frac{\sum_{i\in Nei(s)}sim^2(s,i)}{\sum_{a\in Nei(g)}sim^2(g,a)+\sum_{i\in Nei(s)}sim^2(s,i)}$$

2　因子分析相关理论

2.1　因子分析模型

对于标准化的数据矩阵 $\boldsymbol{X}_{mn}$:

$$\begin{bmatrix}x_{11}&x_{12}&\cdots&x_{1n}\\x_{21}&x_{22}&\cdots&x_{2n}\\\vdots&\vdots&&\vdots\\x_{m1}&x_{m2}&\cdots&x_{mn}\end{bmatrix}$$

其中 $\boldsymbol{X}_n=(x_{1n},x_{2n},\cdots,x_{mn})^{\mathrm{T}}$。对该数据进行因子分析,对应的数学模型为[14]:

$$\begin{cases} \boldsymbol{X}_1 = \boldsymbol{a}_{11}\boldsymbol{F}_1 + \boldsymbol{a}_{12}\boldsymbol{F}_2 + \cdots + \boldsymbol{a}_{1k}\boldsymbol{F}_k + \boldsymbol{e}_1 \\ \vdots \\ \boldsymbol{X}_n = \boldsymbol{a}_{n1}\boldsymbol{F}_1 + \boldsymbol{a}_{n2}\boldsymbol{F}_2 + \cdots + \boldsymbol{a}_{nk}\boldsymbol{F}_k + \boldsymbol{e}_n \end{cases} \tag{6}$$

其中:$k < n$,从而达到降维的目的;$\boldsymbol{F}_j$ 是公共因子,$\boldsymbol{F}_i$ 与 $\boldsymbol{F}_j$ $(i \neq j)$ 之间是两两正交的;$\boldsymbol{e}_i$ 是特殊因子;$\boldsymbol{a}_{ij}$ 是公共因子的负载。

2.2 因子值的求法

在因子分析中,常常需要利用公共因子作进一步的研究,例如用公共因子做回归分析等,因此需要计算因子值。假设第 j 个公共因子的因子值 $\boldsymbol{F}_j$,可以由 $\boldsymbol{X}_1, \boldsymbol{X}_2, \cdots, \boldsymbol{X}_n$ 的样本值计算出来,即:

$$\boldsymbol{F}_j = \begin{bmatrix} f_{1j} \\ f_{2j} \\ \vdots \\ f_{mj} \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \begin{bmatrix} \beta_{1j} \\ \beta_{2j} \\ \vdots \\ \beta_{nj} \end{bmatrix}$$

该式可以表述为 $\boldsymbol{F}_j = \boldsymbol{X}\boldsymbol{\beta}_j$,两边同乘以 $\boldsymbol{X}^{\mathrm{T}}$,即得到 $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{F}_j = \boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{\beta}_j$。容易证明[14] $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{F}_j = \boldsymbol{a}_j = (\boldsymbol{a}_{1j}, \boldsymbol{a}_{2j}, \cdots, \boldsymbol{a}_{nj})$,$\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ 正好是相关系数矩阵 $\boldsymbol{\rho}$。因此有 $\boldsymbol{a}_j = \boldsymbol{\rho}\boldsymbol{\beta}_j$,进而 $\boldsymbol{\beta}_j = \boldsymbol{\rho}^{-1}\boldsymbol{a}_j$,至此得到式(7)。

$$\boldsymbol{F}_j = \boldsymbol{X}\boldsymbol{\rho}^{-1}\boldsymbol{a}_j \tag{7}$$

3 基于因子分析的混合推荐

网络数据库中 m 个用户对 n 个商品的评分矩阵 $\boldsymbol{R} = (r_{ij})_{m\times n}$,以及目标用户 g(不妨设为第 $m+1$ 个用户)对其中 s 件商品(不妨设为前 s 件)的评分数据如图 1 所示。现在希望预测目标用户 g 对项目 $I_{s+1}, \cdots, I_n$ 的评分,图 1 中"?"为需要预测的评价值。

	$\boldsymbol{I}_1$	…	$\boldsymbol{I}_s$	$\boldsymbol{I}_{s+1}$	…	$\boldsymbol{I}_n$	
$\boldsymbol{u}_1$	r_{11}	…	r_{1s}	r_{1s+1}	…	r_{1n}	
⋮	⋮		⋮	⋮		⋮	
$\boldsymbol{u}_m$	r_{m1}	…	r_{ms}	r_{ms+1}	…	r_{mn}	m
$\boldsymbol{u}_g$	r_{g1}	…	r_{gs}	?	…	?	
	s			$n-s$			

图 1 用户-项目评分矩阵

3.1 因子分析

3.1.1 基于用户的因子分析

将图 1 中对应的 m 个用户对 n 个项目的评分矩阵 $\boldsymbol{R} = (r_{ij})_{m\times n}$ 视为 m 个行向量 $\boldsymbol{R}_i = (r_{i1}, r_{i2}, \cdots, r_{in})$,对这 m 个数据向量进行因子分析,得到 K 个用户因子,视为 K 个用户隐分类。

首先,计算每个用户向量 $\boldsymbol{R}_i = (r_{i1}, r_{i2}, \cdots, r_{in})$ 在第 k $(k \leqslant K)$ 个因子 $\boldsymbol{F}_k^U$ 上的负载向量 $\boldsymbol{a}_{ik}$。

然后,计算任意两个用户向量 $\boldsymbol{R}_i$ 和 $\boldsymbol{R}_j$ 的相关系数 ρ_{ij}^U,得到相关系数矩阵 $\boldsymbol{\rho}_U = (\rho_{ij}^U)_{m\times m}$。

最后,根据式(7)得到第 $k(k = 1,2,\cdots,K)$ 个用户因子向量为:

$$\boldsymbol{F}_k^U = \boldsymbol{R}^{\mathrm{T}}\boldsymbol{\rho}_U^{-1}\boldsymbol{a}_k \tag{8}$$

该因子分析可以用图 2 所示的示意图表示。

3.1.2 基于项目的因子分析

对图 1 中前 s 个列向量进行因子分析,得到 H 个项目因子,视为 H 个项目隐分类。

首先,计算前 s 个列向量中每个项目向量 $\underline{\boldsymbol{R}}_j = (r_{1j}, r_{2j}, \cdots, r_{mj}, r_{gj})^{\mathrm{T}}$ 在第 $h(h \leqslant H)$ 个因子 $\boldsymbol{F}_k^I$ 上的负载向量 $\boldsymbol{a}_{jk}$。

然后,计算前 s 个列向量中任意两个项目向量 $\underline{\boldsymbol{R}}_i$ 和 $\underline{\boldsymbol{R}}_j$ 的相关系数 ρ_{ij}^I,得到相关系数矩阵 $\boldsymbol{\rho}_I = (\rho_{ij}^I)_{s\times s}$。

最后,根据式(7)得到第 $h(h = 1,2,\cdots,H)$ 个项目因子向量为(9),该因子分析可以用图 3 所示的示意图表示。

$$\boldsymbol{F}_h^I = \underline{\boldsymbol{R}}^{\mathrm{T}}\boldsymbol{\rho}_I^{-1}\boldsymbol{a}_k \tag{9}$$

$\boldsymbol{u}_1$	r_{11}	…	r_{1s}	r_{1s+1}	…	r_{1n}
$\boldsymbol{u}_2$	r_{21}	…	r_{2s}	r_{2s+1}	…	r_{2n}
⋮	⋮		⋮	⋮		⋮
$\boldsymbol{u}_m$	r_{m1}	…	r_{ms}	r_{ms+1}	…	r_{mn}

⇩ $K<m$

$\boldsymbol{F}_1$	f_{11}	…	f_{1s}	f_{1s+1}	…	f_{1n}
$\boldsymbol{F}_2$	f_{21}	…	f_{2s}	f_{2s+1}	…	f_{2n}
⋮	⋮		⋮	⋮		⋮
$\boldsymbol{F}_K$	f_{K1}	…	f_{Ks}	f_{Ks+1}	…	f_{Kn}

图 2 基于用户的因子分析

$\boldsymbol{I}_1$	$\boldsymbol{I}_1$	…	$\boldsymbol{I}_s$		$\boldsymbol{F}_1$	$\boldsymbol{F}_1$	…	$\boldsymbol{F}_s$
r_{11}	r_{12}	…	r_{1s}	$H<s$ ⇨	f_{11}	f_{12}	…	f_{1s}
⋮	⋮		⋮		⋮	⋮		⋮
r_{m1}	r_{m2}	…	r_{ms}		f_{m1}	f_{m2}	…	f_{ms}
r_{g1}	r_{g2}	…	r_{gs}		f_{g1}	f_{g2}	…	f_{gs}

图 3 基于项目的因子分析

3.2 基于因子的回归分析与预测

3.2.1 基于用户的回归与预测

以用户 g 对前 s 项商品的评分 $\boldsymbol{R}_{g1-s} = (r_{g1}, r_{g2}, \cdots, r_{gs})$ 为因变量(如图 1 所示),以第 $k(k = 1, 2, \cdots, K)$ 个公共因子向量的前 s 项构成的向量 $\boldsymbol{F}_{k,1-s}^U = (f_{k1}, f_{k2}, \cdots, f_{ks})$ 为自变量(如图 2 所示),进行多元线性回归:

$$\boldsymbol{R}_{g1-s} = w_1\boldsymbol{F}_{1,1-s}^U + w_2\boldsymbol{F}_{2,1-s}^U + \cdots + w_K\boldsymbol{F}_{K,1-s}^U + \boldsymbol{C} + \varepsilon \tag{10}$$

首先,采用最小二乘法或其他估计方法,得到回归系数 $\hat{w}_k(k = 1,2,\cdots,K)$,进而得到:

$$\hat{\boldsymbol{R}}_{g1-s} = \hat{w}_1\boldsymbol{F}_{1,1-s}^U + \hat{w}_2\boldsymbol{F}_{2,1-s}^U + \cdots + \hat{w}_K\boldsymbol{F}_{K,1-s}^U + \hat{\boldsymbol{C}} \tag{11}$$

记该回归方程的拟合优度系数(Coefficient of Determination)为 Ψ_g(统计学中习惯记为 R^2,这里为了不引起混淆记为 Ψ)。

然后,将第 $k(k = 1,2,\cdots,K)$ 个公共因子的后 $n-s$ 项构成的向量 $\boldsymbol{F}_{k,s-n}^U = (f_{ks+1}, f_{ks+2}, \cdots, f_{kn})$ 代入式(11),得到用户 g 对后 $n-s$ 件商品评分的预测值:

$$\hat{\boldsymbol{R}}_{gs-n}^U = (\hat{r}_{gs+1}^g, \hat{r}_{gs+2}^g, \cdots, \hat{r}_{gn}^g) \tag{12}$$

3.2.2 基于项目的回归与预测

使用前 m 个用户对第 t 项商品 $(s < t \leqslant n)$ 的评分 $\underline{\boldsymbol{R}}_{1-m,t} = (r_{1t}, r_{2t}, \cdots, r_{mt})^{\mathrm{T}}$ 为因变量(如图1所示),以第 $h(= 1,2,\cdots,H)$ 个公共因子向量的前 m 个元素构成的向量 $\boldsymbol{F}_{1-m,h}^I = (f_{1h}, f_{2h}, \cdots, f_{mh})^{\mathrm{T}}$ 为自变量(如图 3 所示),进行多元线性回归:

$$\underline{\boldsymbol{R}}_{1-m,t} = w_1\boldsymbol{F}_{1-m,1}^I + w_2\boldsymbol{F}_{1-m,2}^I + \cdots + w_K\boldsymbol{F}_{1-m,K}^I + \boldsymbol{C} + \varepsilon \tag{13}$$

首先,采用最小二乘法或其他估计方法,得到回归系数 $\hat{w}_h(h = 1,2,\cdots,H)$,进而得到:

$$\underline{\hat{\boldsymbol{R}}}_{1-m,t} = \hat{w}_1\boldsymbol{F}_{1-m,1}^I + \hat{w}_2\boldsymbol{F}_{1-m,2}^I + \cdots + \hat{w}_H\boldsymbol{F}_{1-m,K}^I + \hat{\boldsymbol{C}} \tag{14}$$

记该回归方程的拟合优度系数(Coefficient of Determination)为 Ψ_t。

然后,将第 $h(h = 1,2,\cdots,H)$ 个公共因子的第 g 项 f_{gh} 代

入式(14)得到用户 g 对第 t 项商品的评分 $\hat{r}_{gt}^{t}$。

类似得到用户 g 对后 $n-s$ 件商品的评分:

$$\hat{\boldsymbol{R}}_{gs-n}^{I}=(\hat{r}_{gs+1}^{s+1},\hat{r}_{gs+2}^{s+2},\cdots,\hat{r}_{gn}^{n})$$

可以看出,在预测用户 g 在后 $n-s$ 件商品的评分时,基于用户因子的方法只需做一次回归分析,而基于项目因子的方法需要做 $n-s$ 次回归分析。

3.3　预测值的加权平均

为结合基于用户的预测和基于项目的预测,本文引入权重 λ_g 和 λ_I,它们都在区间[0,1]取值,λ_g 表示基于用户因子的算法对预测结果的动态影响程度,λ_I 表示基于项目的算法对预测结果的动态影响程度。通过自动选择合适的权重,可以动态结合基于用户因子和基于项目因子的有利因素,使预测结果达到一个良好的平衡,从而使推荐更准确也更稳定。对权重 λ_g 和 λ_I 的动态取值考虑如下:

1)当基于用户因子的回归分析拟合优度越好,即 Ψ_g 越大,说明用户因子对预测结果影响越大,此时 λ_g 应越大,反之若基于项目因子的回归分析拟合优度越好,即 Ψ_t 越大,λ_I 越大,所以 λ_g 和 λ_I 的相对大小,取决于 Ψ_g 和 Ψ_t 的大小。

2)不同用户 Ψ_g 不同,所以不同用户 λ_g 应该不同;同时项目的 Ψ_t 不同,所以不同项目的 λ_I 也应该不同。

为此本文利用 Ψ_g 和 Ψ_t 的比值来动态设定 λ_g 和 λ_I,由于 Ψ_g 和 Ψ_t(统计学中的 R^2)恒为正数,所以也保证了权重的非负性。该权重用式(15)表示:

$$\begin{cases}\lambda_g=\dfrac{\Psi_g}{\Psi_g+\Psi_t}\\ \lambda_t=\dfrac{\Psi_t}{\Psi_g+\Psi_t}\end{cases}\tag{15}$$

其中 $\lambda_g+\lambda_I=1$,经过动态加权后对目标用户 g 在项目 $t(s<t\leqslant n)$ 上的评分预测用式(16)表示。

$$\hat{r}_{gt}=\lambda_g\hat{r}_{gt}^{g}+\lambda_t\times\hat{r}_{gt}^{t}\tag{16}$$

计算得到当前用户对未评分项目的预测评分后,就可以选择预测评分最高的若干项推荐给当前用户,这也是目前的推荐系统中常用的 Top-N 推荐。

4　算例和实验

4.1　算例

为了清晰说明本文的算法,下面提供如表 1 所示的用户对不同电影的评价数据,其中用户对不同电影的评价值为 1~5 分。现在拟将电影 9,10,11 三部电影中部分电影推荐给用户 7,即需要预测表 1 中"?"对应的数值。

表 1　用户-电影评分数据

用户	电影										
	1	2	3	4	5	6	7	8	9	10	11
用户 1	5	2	2	4	3	2	4	2	2	5	4
用户 2	4	2	2	5	3	1	3	1	1	4	4
用户 3	5	1	3	5	3	2	5	2	1	5	5
用户 4	1	4	5	1	4	4	2	4	4	2	2
用户 5	2	5	5	2	4	3	2	5	5	3	1
用户 6	3	4	4	1	5	5	1	4	5	3	3
用户 7	3	3	3	4	4	4	4	3	?	?	?

4.1.1　基于用户因子的预测

对前 6 行数据对应 6 个行向量进行因子分析,得到 $\boldsymbol{F}_1^U$ 和 $\boldsymbol{F}_2^U$ 两个因子,如表 2 所示。其中用户 7 行对应"?"是需要预测的用户 7 对电影 9,10,11 的评价值。

表 2　用户公共因子值

用户因子	电影										
	1	2	3	4	5	6	7	8	9	10	11
$\boldsymbol{F}_1^U$	-1	-0.5	-0.9	0.5	-0.1	0.1	1.1	-1.3	-0.9	1.4	1.4
$\boldsymbol{F}_2^U$	0.1	-0.3	-1.4	-1.8	0.9	1.9	-0.3	0.1	0.5	0.3	-0.1
用户 7	3	3	3	4	4	4	4	3	?	?	?

以用户 7 的数据为因变量,以 $\boldsymbol{F}_1^U$ 和 $\boldsymbol{F}_2^U$ 的前 8 个数据为自变量,进行线性回归得到如下多项式:

$$\hat{\boldsymbol{R}}_7=3.66+0.57\times\boldsymbol{F}_1^U+0.12\times\boldsymbol{F}_2^U;\Psi_g=0.82$$

将 $\boldsymbol{F}_1^U$ 和 $\boldsymbol{F}_1^U$ 的后 3 个数据代入上述多项式得到:

$$\hat{r}_{7,9}^{7}=3.21$$

$$\hat{r}_{7,10}^{7}=4.49$$

$$\hat{r}_{7,11}^{7}=4.45$$

4.1.2　基于项目因子的预测

对前 8 列数据对应 8 个列向量进行因子分析,得到 3 个因子 $\boldsymbol{F}_1^I$、$\boldsymbol{F}_2^I$、$\boldsymbol{F}_3^I$ 如表 3 所示。

表 3　项目公共因子值

用户	电影因子			电影		
	$\boldsymbol{F}_1^I$	$\boldsymbol{F}_2^I$	$\boldsymbol{F}_3^I$	9	10	11
用户 1	-0.84	-0.40	0.23	2	5	4
用户 2	-0.98	-1.43	-0.92	1	4	4
用户 3	-0.25	-0.22	1.59	1	5	5
用户 4	1.25	0.00	-0.31	4	2	2
用户 5	1.52	-0.49	-0.36	5	3	1
用户 6	-0.60	1.63	-1.19	5	3	3
用户 7	-0.10	0.92	0.97	?	?	?

其中电影 9,10,11 列对应的"?"是需要预测的项目评分。

分别以项目 9,10,11 的评分为因变量,以 $\boldsymbol{F}_1^I$、$\boldsymbol{F}_2^I$、$\boldsymbol{F}_3^I$ 为自变量,得到如下 3 个回归方程:

$$\hat{\boldsymbol{R}}_9=3.02+1.07\times\boldsymbol{F}_1^I+1.00\times\boldsymbol{F}_2^I-0.83\times\boldsymbol{F}_3^I;\Psi_9=0.94$$

$$\hat{\boldsymbol{R}}_{10}=3.76-0.76\times\boldsymbol{F}_1^I-0.25\times\boldsymbol{F}_2^I+0.71\times\boldsymbol{F}_3^I;\Psi_{10}=0.89$$

$$\hat{\boldsymbol{R}}_{11}=3.30-1.12\times\boldsymbol{F}_1^I-0.46\times\boldsymbol{F}_2^I+0.76\times\boldsymbol{F}_3^I;\Psi_{11}=0.95$$

将 $\boldsymbol{F}_1^I$、$\boldsymbol{F}_2^I$、$\boldsymbol{F}_3^I$ 的最后一个数据代入上述多项式有:

$$\hat{r}_{7,9}^{9}=3.03$$

$$\hat{r}_{7,10}^{10}=4.29$$

$$\hat{r}_{7,11}^{11}=3.72$$

4.1.3　加权平均

用户 7 对 9,10,11 三部电影的评分预测为:

$$\hat{r}_{7,9}=\frac{0.82}{0.82+0.94}\times3.21+\frac{0.94}{0.82+0.94}\times3.03=3.11$$

$$\hat{r}_{7,10}=\frac{0.82}{0.82+0.89}\times4.49+\frac{0.89}{0.82+0.89}\times4.29=4.39$$

$$\hat{r}_{7,11}=\frac{0.82}{0.82+0.95}\times4.45+\frac{0.95}{0.82+0.95}\times3.72=4.06$$

4.2　实验

为了比较传统的基于项目的协同过滤推荐(item-based CFR)算法[3],基于用户的协同过滤(user-ased CFR)算法[4]和本文提出的基于因子分析的混合协同推荐(MCR-F)算法

的精确度，下面采用 GroupLens 研究项目组搜集的公共数据集 MovieLens(http://www.grouplens.org)进行实验，数据集中包括 943 个用户对 1 682 部电影的 100 000 个评分，评分范围为 1～5 分，每个用户至少评过 20 部电影。选取 430 个用户的 3 万条评分数据，并按照4:1的比例划分训练集和测试集。

采用平均绝对偏差(Mean Absolute Error, MAE)作为度量标准。为了清晰起见，进行 3 组对比。

第 1 组　比较基于项目的协同过滤推荐算法和基于项目因子分析的推荐方法。传统算法将最近项目邻居个数从 5 增加到 40(间隔为 5)，基于因子分析的方法则将项目因子个数从 5 增加到 40(间隔同样为 5)，预测结果如图 4 所示。

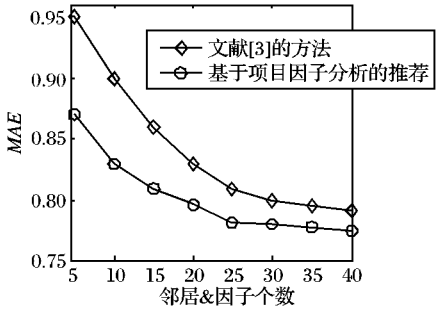


图 4　第 1 组的平均绝对误差 *MAE* 比较 3

第 2 组　比较基于用户的协同过滤推荐算法和基于用户因子分析的推荐方法。传统算法将最近用户邻居个数从 5 增加到 40，基于因子分析的方法则将用户因子个数从 5 增加到 40，预测结果如图 5 所示。

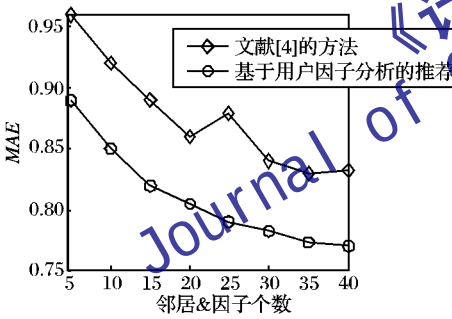


图 5　第 2 组的平均绝对误差 *MAE* 比较

第 3 组　比较只考虑项目因子分析的推荐方法，只考虑用户因子分析的推荐方法和基于用户和项目因子分析的混合推荐算法。均将因子个数从 5 增加到 40，预测结果如图 6 所示。

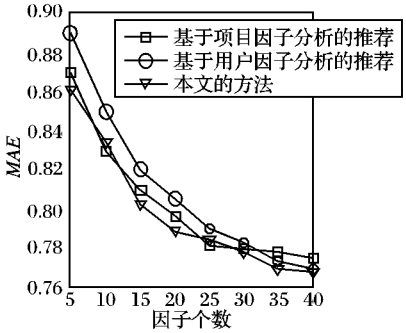


图 6　第 3 组的平均绝对误差 MAE 比较

从图 4～5 可看出，基于因子分析的协同推荐算法比传统的协同过滤推荐算法[3-4]更精确，这是因为，基于因子分析的方法中，这些因子总体代表了几乎所有用户(或已评项目)的信息，而传统方法选择少数几个最近邻居用户(或项目)，必然丢掉了很多用户(或项目)的信息。这与算法的平均绝对偏差 *MAE* 大体随邻居的增加而减少是一致的。

从图 6 可以看出，大体上具有如下规律：通过动态加权将基于用户因子分析的预测方法与基于项目因子分析的预测方法进行综合的混合协同推荐方法比单独只考虑一类因子时更准确。

文献[6]提出了一种混合协同过滤推荐模型(简称 W-模型)，该模型与本文的模型的主要区别在于：W-模型是传统协同过滤推荐模型(文献[3]和文献[4])的混合；本文的模型是基于用户因子分析的推荐模型和基于项目因子分析的推荐模型(这两个模型也由本文作者提出)的混合。

图 4～5 已经通过实验分别说明了基于项目因子分析的推荐模型和基于用户因子分析的推荐模型比文献[3]和文献[4]的模型推荐结果更准确，可以推断本文的混合推荐模型比文献[6]的混合模型推荐效果更准确。

5　结语

本文提出了基于用户和项目因子分析的混合协同推荐算法。该算法首先采用因子分析的方法将用户和项目降维为若干用户因子和若干项目因子；然后分别用目标用户和待评价项目为因变量，以用户因子和项目因子为自变量构造两个回归模型，进而得到目标用户在待评项目上的两个预测值；最后通过两者的加权得到最终的预测值。

本算法的创新之处在于：1)通过采用因子分析的方法一方面达到了数据降维的目的，另一方面最大限度地保留了数据所有信息。传统的算法在所有的数据中寻找邻居，导致计算时间过长，并且采用最近邻居的信息进行推荐，必然损失了非邻居数据的信息。基于聚类的方法，虽然解决了数据量大而导致的计算时间过长的问题，但是仍然存在丢失信息的问题。2)计算因子分析的过程可以采用离线的方式，大大节省了系统开销，传统的算法在对目标用户进行预测时，需要遍历所有用户数据寻找最近邻居，而基于因子分析的方法由于公共因子可以事先计算出来，在线的计算只需要做回归分析即可。3)考虑了用户和项目之间的联系，通过用户因子分析预测值和项目因子分析预测值的动态加权平均同时考虑了用户之间的关联和项目之间的关联。

最后文章通过实验证明了该算法的有效性，为以后研究推荐算法提供了一种新的途径。

参考文献：

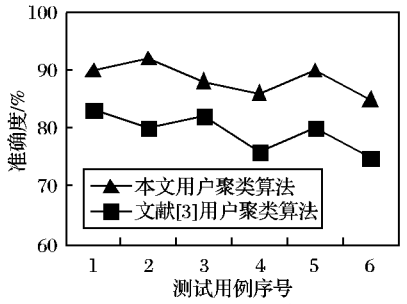
[1] BREESE J S, HECKERMAN D, KAIDIE C. Empirical analysis of predictive algorithm for collaborative filtering [C]// Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence. San Francisco: Morgan Kaufmann Publisher, 1998:43－52.

[2] HERLOCKER L J, KONSTAN A J, RIEDL T J. Empirical analysis of design choices in neighborhood-based collaborative filtering algorithms[J]. Information Retrieval, 2002, 5(4):287－310.

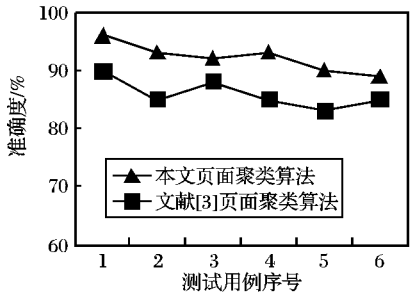
[3] LEE H C, LEE S J, CHUNG Y J. A study on the improved collaborative filtering algorithm for recommender system [C]// The 5th International Conference on Software Engineering Research Management and Applications. Washington, DC: IEEE, 2007:297－304.

(下转第 1390 页)

类,并将聚类结果和用手工方法获得的结果进行对比分析,得到本文用户聚类算法和页面聚类算法的准确度平均值分别约为 88% 和 92%,而文献[3]用户聚类算法和页面聚类算法的准确度平均约为 78% 和 86%,可见本文提出的算法具有更高的准确性,如图 2 所示。



(a) 用户聚类算法比较



(b) 页面聚类算法比较

图 2 算法准确度对比图

为了进一步检验算法的有效性和扩展性,从 Web 服务器(http://c.lstc.edu.cn)不同时段的日志数据分割了 6 组大小分别为 100 KB,256 KB,468 KB,625 KB,872 KB,1000 KB 的数据,先利用现有的 Web 日志挖掘工具对各组日志进行数据预处理,去掉访问日期、时间、等次要信息,只统计保留了每一个日志数据的 URL、UserIP 及相应的 URLLevel 和点击次数,同时生成各组数据的 URL-User 关联矩阵。将该关联矩阵导入 Matlab,用本文算法和文献[3]算法进行用户聚类,记录下对应的 CPU 时间,对比效果见图 3。

从图 3 中可看出本文算法耗时比文献[3]算法略要高些,这主要原因是两种算法基本思想是一致的,只是新算法采用相对 Hamming 距离进行聚类判断,同时对聚类结果的确定更精细,因此新算法肯定会多耗费一些时间,但大大提高了聚类的准确性,因此付出较小的时间代价是值得的。而且随着数据量的增多,本文算法的时间曲线的上升趋势比较缓慢,并呈现近似的线性增长关系,说明本文算法具有较好的有效性和扩展性。

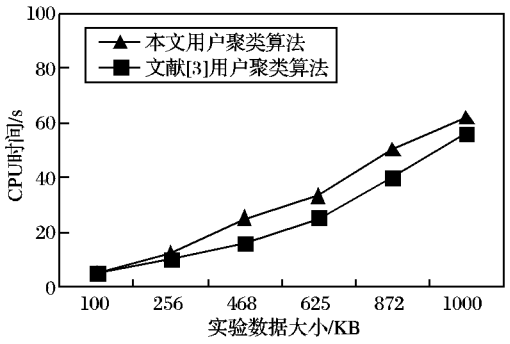


图 3 两种算法耗时对比

5 结语

本文主要针对 Web 日志挖掘中的用户聚类和页面聚类问题,在现有研究基础上提出了一种新的聚类算法。在该算法中,通过计算列向量的两两最大范数和 Hamming 距离构造了相对 Hamming 距离矩阵,并以此为基础通过设定阈值给出了相似用户群体,最后通过定义类不一致度并设计算法对聚类结果进行了确认,获得了最终的用户聚类。类似对行向量计算相对 Hamming 距离及阈值,可给出相关页面聚类。实验表明,本文的算法准确性高于文献[3]中的算法,而且具有良好的扩展性。

参考文献:

[1] 杨怡玲,管旭东,尤晋元. 基于页面内容和站点结构的页面聚类挖掘算法[J]. 软件学报, 2002,13(3):467 - 469.

[2] MOBASHER B, COOLEY R. Creating adaptive Web sites through usage-based clustering of URLs [C]// Proceedings of the 1999 IEEE Knowledge and Data Engineering Exchange Workshop. Washington, DC: IEEE, 1999: 32 - 37.

[3] 宋擒豹,沈钧毅. Web 日志的高效多能挖掘算法[J]. 计算机研究与发展, 2001, 38(3): 328 - 333.

[4] 朱志国,邓贵仕. Web 使用挖掘技术的分析与研究[J]. 计算机应用研究,2008,25(1):29 - 36.

[5] 孙吉贵, 刘杰, 赵连宇. 聚类算法研究[J]. 软件学报, 2008, 19(1): 48 - 61.

[6] BEEFERMAN D , BERGER A . Agglomerative clustering of a search engine query log [C]// Proceedings of the 6th ACM SIGKDD International Conference. Boston: ACM Press, 2000: 407 - 415.

[7] 李新叶, 苑津莎. 一种用于 Web 搜索的高效聚类算法[J]. 计算机工程, 2006, 32(20): 38 - 39.

[8] FU Y, SANDHU K, SHIH M. A generalization-based approach to clustering of Web usage session [C]// Proceedings of WEBKDD '99. Berlin: Springer, 2000: 21 - 38.

[9] KUMAR P, KRISHNA P R, BAPI R S, *et al*. Rough clustering of sequential data [J]. Data & Knowledge Engineering, 2007, 63(2): 183 - 199.

(上接第 1386 页)

[4] DESHPANDE M, KARPIS G. Item-based top-n recommendation algorithms [J]. ACM Transactions on Information Systems, 2004, 22(1): 143 - 177.

[5] 邓爱林,朱扬勇,施伯乐. 基于项目评分预测的协同过滤推荐算法[J]. 软件学报,2003,14(9):1621 - 1628.

[6] 汪静,印鉴,郑利荣,等. 基于共同评分和相似性权重的协同过滤推荐算法[J]. 计算机科学,2010,37(2):99 - 104.

[7] MARLIN B. Collaborative Filtering: A machine learning perspective [D]. Toronto: University of Toronto, 2004.

[8] 唐晓波,樊静. 基于用户聚类的商品推荐[J]. 情报杂志,2009,28(6):143 - 146.

[9] 李涛,王建东,叶飞跃,等. 一种基于用户聚类的协同过滤推荐算法[J]. 系统工程与电子技术,2007,29(7):1178 - 1182.

[10] 张海鹏,李烈彪,李仙,等. 基于项目分类预测的协同过滤推荐算法[J]. 情报学报,2008,27(2):218 - 223.

[11] 侯翠琴,焦李成,张文革. 一种压缩稀疏矩阵用户评分矩阵的协同过滤算法[J]. 西安电子科技大学学报: 自然科学学报,2009,36(4):614 - 618.

[12] 张亮,李敏强. 面向协同过滤的真实偏好高斯混合模型[J]. 系统工程学报,2007,22(6):613 - 619.

[13] 赵宏霞,杨皎平,陈宗娇. 面向用户需求的神经网络挖掘方法[J]. 管理评论,2005(11):53 - 57.

[14] 马庆国. 管理统计:数据获取、统计原理,SPSS 工具与应用研究[M]. 北京:科学出版社,2002:315 - 326.