基于 Logistic 函数的社会化矩阵分解推荐算法

郭云飞， 方耀宁， 扈红超
（国家数字交换系统工程技术研究中心，河南，郑州 450002）

摘 要：持续指数增长的互联网逐渐带来了信息过载问题，使得推荐系统提供的信息过滤服务尤为重要. 协同过滤是推荐系统领域最为成功的技术，但依然存在数据稀疏性等问题. 社会关系信息能够有效提高推荐系统的预测准确性. 为解决数据稀疏性问题，本文提出了一种利用 Logistic 函数的社会化矩阵分解推荐算法. 在 3 组真实数据结合上的实验结果表明，本文提出的算法能够提供更准确的推荐结果，特别是在数据稀疏的情况下，显著缓解了数据稀疏性问题.

关键词：推荐系统；协同过滤；矩阵分解；社会关系；Logistic 函数
中图分类号：TP 393 **文献标志码**：A **文章编号**：1001-0645(2016)01-0070-05
DOI：10.15918/j.tbit1001-0645.2016.01.013

A Social Matrix Factorization Recommender Algorithm Based on Logistic Function

GUO Yun-fei， FANG Yao-ning， HU Hong-chao
(National Digital Switching System Engineering and Technological R&D Center, Zhengzhou, Henan 450002, China)

Abstract: The ongoing exponential growth of the Internet brings an information overload, which greatly increases the necessity of effective recommender systems for information filtering. However, collaborative filtering, which is recognized as the most successful technique in designing recommender systems, still encounters the data sparsity problem. Social relations have been found to be effective to improve the prediction accuracy of recommender systems. In order to handle the data sparsity problem, this paper proposed a new social matrix factorization recommender algorithm by leveraging the Logistic function. Experimental results on three real-world datasets illustrate that the proposed method provides more accurate recommendation results, especially under sparse conditions.

Key words: recommender system; collaborative filtering; matrix factorization; social links; Logistic function

互联网的飞速发展将人类带入了信息爆炸的时代，人们经历了从信息匮乏到泛滥的剧变，发现从海量互联网信息中获取所需内容却成了一件复杂的事情——信息过载(information overload). 传统的搜索引擎只能被动提供无差别的搜索服务，而推荐系统利用数据挖掘、人工智能等技术能够主动帮助人们过滤信息，提供个性化服务[1-2]. 在过去的十多年里，推荐系统在电子商务网站、社交网络中大量涌现，正在引领人类进入一个多元化、个性化的网络新时代. 推荐系统不仅可以提供娱乐、生活方面的服务推荐，如网页、电影、好友等；而且可以提供更加专业化的推荐，如图书、论文、专利技术等[2-3].

收稿日期：2013-09-02
基金项目：国家“九七三”计划项目(2012CB315901)；国家“八六三”计划项目(2011AA01A103)；国家自然科学基金资助项目(61309020)
作者简介：郭云飞(1963—)，男，教授，博士生导师，
通信作者：方耀宁(1987—)，男，硕士，E-mail：fyn07@163.com.

协同过滤推荐算法在学术界和工业界都有了长足的发展,新的算法和研究热点不断涌现[2-3]. 其中,基于矩阵分解的推荐算法在预测准确性和稳定性上显示出较大优势,得到最为广泛的认可[4-6]. 矩阵分解推荐算法把评分矩阵看作是低秩矩阵,并分解成用户和项目两个低维特征矩阵的乘积,从而对未知评分进行预测. 虽然,矩阵分解算法的预测准确性较高,但依然面临着数据稀疏性等问题.

社会关系网络连接着人类社会与网络空间,正在逐渐模糊现实与虚拟的界限. 社交网络中以社会关系为纽带自发形成的群组,与现实社会中"物以类聚,人以群分"的原理一致. 社交网络能够详细记录用户历史行为信息,这就为分析用户行为特征,预测用户行为提供了数据基础. 对社会关系进行挖掘,可以估计出用户间的信任程度,有望解决推荐系统中的冷启动、数据稀疏性等问题[5-6].

本文围绕如何利用社会关系来解决数据稀疏性问题进行研究,提出一种基于 Logistic 函数的社会化矩阵分解推荐算法(logistic social matrix factorization, LSMF). LSMF 算法根据社会关系建立简单高效的好友信任度评估机制,保证用户特征因子与好友特征因子相近,并利用 Logistic 函数对特征因子进行非线性映射. 在 3 种真实数据集合上的实验表明,LSMF 算法能够明显降低预测误差(约 5% RMSE),缓解数据稀疏性问题.

1　LSMF 算法

1.1　LSMF 模型

LSMF(logistic social matrix factorization)从 Logistic 函数的使用和信任度计算两个方面对 Social MF 模型进行了改进[7-8],如图 1 所示.

Logistic($g(x)=1/(1+e^{-x})$)定义域为$(-\infty,+\infty)$,值域为$(0,1)$. Logistic 函数在定义域内呈现出先缓慢增长,然后加速增长,最后逐渐稳定的趋势,能够较好反映生物种群发展、神经元非线性感知、人类认知学习过程等.

如图 1 所示,LSMF 模型中用 $\boldsymbol{P}$ 和 $\boldsymbol{Q}$ 两组系数来表示 Logistic 函数本身的特征,用$(\boldsymbol{P}_i+\boldsymbol{Q}_j)g(x)$对用户/项目组合 $\boldsymbol{U}_i^{\mathrm{T}}\boldsymbol{V}_j$ 进行评分映射. 其中 $\boldsymbol{P}_i$ 和 $\boldsymbol{Q}_j$ 分别与用户 i 和项目 j 相关. 简单起见,假设系数矩阵 $\boldsymbol{P}$ 和 $\boldsymbol{Q}$ 分别服从均值为 m(m 为系统评分均值),方差分别为 σ_P^2 和 σ_Q^2 的高斯分布,如式(1)(2)所示. 此时,用户 i 的特征由 $\boldsymbol{U}_i$ 和 $\boldsymbol{P}_i$ 共同体现,项目 j 的特征由 $\boldsymbol{V}_j$ 和 $\boldsymbol{Q}_j$ 体现. 假设用户 i 对项目 j 的评分服从高斯分布,如式(3)所示.

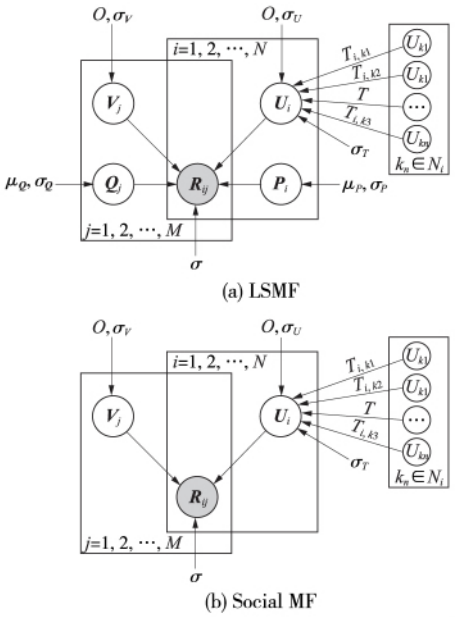


图 1　LSMF 模型与 Social MF 的对比
Fig. 1　Graphical models of LSMF and Social MF

$$p(\boldsymbol{P}\mid m,\sigma_P^2)=\prod_{i=1}^{N}N(\boldsymbol{P}_i\mid m,\sigma_P^2),\qquad(1)$$

$$p(\boldsymbol{Q}\mid m,\sigma_Q^2)=\prod_{j=1}^{M}N(\boldsymbol{Q}_j\mid m,\sigma_Q^2),\qquad(2)$$

$$p(\boldsymbol{R}_{ij}\mid\boldsymbol{U}_i,\boldsymbol{P}_i,\boldsymbol{V}_j,\boldsymbol{Q}_j,\sigma^2)=N(\boldsymbol{R}_{ij}\mid(\boldsymbol{P}_i+\boldsymbol{Q}_j)g(\boldsymbol{U}_i^{\mathrm{T}}\boldsymbol{V}_j),\sigma^2).\qquad(3)$$

Logistic 函数实现对特征因子非线性映射. 与 Social MF 模型相同, LSMF 模型中用户的特征因子与均值为 0 的高斯分布正相关,保证特征因子的取值接近 0 以防止过拟合. 同时,用户的特征因子与好友特征因子的加权平均值正相关,即

$$p(\boldsymbol{U}\mid\boldsymbol{T},\sigma_U^2,\sigma_T^2)\propto p(\boldsymbol{U}\mid\sigma_U^2)p(\boldsymbol{U}\mid\boldsymbol{T},\sigma_T^2)=\prod_{i=1}^{N}N(\boldsymbol{U}_i\mid 0,\sigma_U^2\boldsymbol{I})\prod_{i=1}^{N}N(\boldsymbol{U}_i\mid\sum_{k\in N_i}\boldsymbol{T}_{i,k}\boldsymbol{U}_k,\sigma_T^2\boldsymbol{I}).\qquad(4)$$

于是,已知评分矩阵 $\boldsymbol{R}$ 和社会关系矩阵 $\boldsymbol{T}$,根据贝叶斯公式可以计算出用户和项目特征因子 $\boldsymbol{U}$,$\boldsymbol{V}$,$\boldsymbol{P}$ 和 $\boldsymbol{Q}$ 的后验概率为

$$p(\boldsymbol{U},\boldsymbol{V},\boldsymbol{P},\boldsymbol{Q}\mid\boldsymbol{R},\boldsymbol{T},m,\sigma^2,\sigma_P^2,\sigma_Q^2,\sigma_U^2,\sigma_V^2,\sigma_T^2)\propto p(\boldsymbol{R}\mid\boldsymbol{U},\boldsymbol{P},\boldsymbol{V},\boldsymbol{Q},\sigma^2)p(\boldsymbol{P}\mid m,\sigma_P^2).\qquad(5)$$

$$p(\boldsymbol{Q}\mid m,\sigma_Q^2)p(\boldsymbol{V}\mid\sigma_V^2)p(\boldsymbol{U}\mid\boldsymbol{T},\sigma_U^2,\sigma_T^2)=\prod_{i=1}^{N}\prod_{j=1}^{M}[N(\boldsymbol{R}_{ij}\mid(\boldsymbol{P}_i+\boldsymbol{Q}_j)g(\boldsymbol{U}_i^{\mathrm{T}}\boldsymbol{V}_j),\sigma^2)]^{I_{ij}}\times$$

$$\prod_{i=1}^{N}N(\boldsymbol{P}_i\mid m,\sigma_{\boldsymbol{P}}^2)\prod_{j=1}^{M}N(\boldsymbol{Q}_j\mid m,\sigma_{\boldsymbol{Q}}^2)\times$$
$$\prod_{j=1}^{M}N(\boldsymbol{V}_j\mid 0,\sigma_{\boldsymbol{V}}^2\boldsymbol{I})\times\prod_{i=1}^{N}N(\boldsymbol{U}_i\mid 0,\sigma_{\boldsymbol{U}}^2\boldsymbol{I})\times$$
$$\prod_{i=1}^{N}N(\boldsymbol{U}_i\mid\sum_{k\in N_i}\boldsymbol{T}_{i,k}\boldsymbol{U}_k,\sigma_{\boldsymbol{T}}^2\boldsymbol{I}).$$

对上式取对数，得到

$$\ln p(\boldsymbol{U},\boldsymbol{V},\boldsymbol{P},\boldsymbol{Q}\mid\boldsymbol{R},\boldsymbol{T},m,\sigma^2,\sigma_{\boldsymbol{P}}^2,\sigma_{\boldsymbol{Q}}^2,\sigma_{\boldsymbol{U}}^2,\sigma_{\boldsymbol{V}}^2,\sigma_{\boldsymbol{T}}^2)=$$
$$-\frac{1}{2\sigma^2}\sum_{i=1}^{N}\sum_{j=1}^{M}I_{ij}[\boldsymbol{R}_{ij}-(\boldsymbol{P}_i+\boldsymbol{Q}_j)g(\boldsymbol{U}_i^{\mathrm{T}}\boldsymbol{V}_j)]^2-$$
$$\frac{1}{2\sigma_{\boldsymbol{P}}^2\sigma_{\boldsymbol{U}}^2}\sum_{i=1}^{N}[\sigma_{\boldsymbol{P}}^2\boldsymbol{U}_i^{\mathrm{T}}\boldsymbol{U}_i+\sigma_{\boldsymbol{U}}^2(\boldsymbol{P}_i-m)^2]-$$
$$\frac{1}{2\sigma_{\boldsymbol{Q}}^2\sigma_{\boldsymbol{V}}^2}\sum_{j=1}^{M}[\sigma_{\boldsymbol{Q}}^2\boldsymbol{V}_j^{\mathrm{T}}\boldsymbol{V}_j+\sigma_{\boldsymbol{V}}^2(\boldsymbol{Q}_j-m)^2]-$$
$$\frac{1}{2\sigma_{\boldsymbol{T}}^2}\sum_{i=1}^{N}[(\boldsymbol{U}_i-\sum_{k\in N_i}\boldsymbol{T}_{i,k}\boldsymbol{U}_k)^{\mathrm{T}}(\boldsymbol{U}_i-\sum_{k\in N_i}\boldsymbol{T}_{i,k}\boldsymbol{U}_k)]+C. \quad (6)$$

式中 C 为与 $\boldsymbol{U},\boldsymbol{V},\boldsymbol{P}$ 和 $\boldsymbol{Q}$ 无关的一个常数. 注意到 $\sum_{k\in N_i}\boldsymbol{T}_{ik}=1$，$\sum_{k\in N_i}\boldsymbol{T}_{ik}\boldsymbol{U}_k$ 是用户 i 所有好友特征因子的加权平均. 最大化式(6)等价于最小化式(7)：

$$E=\sum_{i=1}^{N}\sum_{j=1}^{M}[(\boldsymbol{P}_i+\boldsymbol{Q}_j)g(\boldsymbol{U}_i^{\mathrm{T}}\boldsymbol{V}_j)-\boldsymbol{R}_{ij}]^2+$$
$$\beta\sum_{i=1}^{N}\left\|\boldsymbol{U}_i-\sum_{k\in N_i}\boldsymbol{T}_{i,k}\boldsymbol{U}_k\right\|^2+\lambda_{\boldsymbol{U}}\|\boldsymbol{U}\|^2+\lambda_{\boldsymbol{V}}\|\boldsymbol{V}\|^2+$$
$$\lambda_{\boldsymbol{P}}\|\boldsymbol{P}-m\boldsymbol{I}\|^2+\lambda_{\boldsymbol{Q}}\|\boldsymbol{Q}-m\boldsymbol{I}\|^2. \quad (7)$$

式中，$\lambda_{\boldsymbol{U}}=\sigma^2/\sigma_{\boldsymbol{U}}^2$，$\lambda_{\boldsymbol{V}}=\sigma^2/\sigma_{\boldsymbol{V}}^2$，$\lambda_{\boldsymbol{P}}=\sigma^2/\sigma_{\boldsymbol{P}}^2$，$\lambda_{\boldsymbol{Q}}=\sigma^2/\sigma_{\boldsymbol{Q}}^2$，$\beta=\sigma^2/\sigma_{\boldsymbol{T}}^2$.

为了减少正则化系数的个数，通常设定 $\lambda_{\boldsymbol{U}}=\lambda_{\boldsymbol{V}}$，$\lambda_{\boldsymbol{P}}=\lambda_{\boldsymbol{Q}}$. β 为社会化影响因子，β 越大表示好友对目标用户的影响力越大，$\beta=0$ 时，模型退化为利用 Logistic 函数的矩阵分解推荐算法. 与矩阵分解推荐算法一样[5-6]，可以用随机梯度下降法方便地求解式(7).

1.2 信任度计算

“信任”是一个跨学科的概念，在不同领域的具体阐释会有不同. 信任度将“信任”量化，常用来衡量被信任者对目标用户影响程度的大小. 由于网络环境的复杂性、随机性、不确定性以及应用场景的多样性，社交网络好友间信任度的评估并不是一个简单的课题. 信任度的计算是搜索引擎、推荐系统领域的重要研究内容之一. 基于信任度计算的社会化推荐算法能够缓解协同过滤算法中的数据稀疏性问题，不同应用场景下信任度的计算方法也有很大区别[8].

Social MF 中所有好友具有相同的权值，表示不同好友对目标用户的影响力相同. 这种方法显然是不合理的，因为社会关系中包含了不同的社交圈，不同社交圈的兴趣爱好不同；而且，相同社交圈中的不同用户对于其他用户的影响力也是不同的.

一种直观的假设是，目标用户对好友的信任度与他们共同选择过的项目数量正相关. 因为只要用户 i 选择了项目 j，无论评分是高还是低，都表示用户 i 对项目 j 这种类型的事物感兴趣，这是一种含蓄的反馈信息. 与基于评分相似性的信任度计算方法不同，本文采用一种简单高效的方法：

$$\boldsymbol{T}_{ik}=\begin{cases}n_0+n_{ik} & k\in N_i\\ 0 & \text{其他}\end{cases}. \quad (8)$$

式中：n_{ik} 为用户 i 和用户 k 共同选择过的项目个数；n_0 是赋值给所有好友的基础权值. $n_0=0$ 表示只有共同评分过的好友才对目标用户有影响力，且正相关于共同选择过的项目个数；$n_0>0$ 表示所有好友都对目标用户有影响力. 进一步对信任度矩阵 $\boldsymbol{T}$ 进行归一化处理，使得 $\boldsymbol{T}$ 的行和为 1.

1.3 复杂度分析

采用随机梯度下降法进行训练时，LSMF 算法的复杂度主要由计算梯度的过程决定. 用 $\bar{r}$ 表示用户的平均评分数，用 $\bar{l}$ 表示用户的平均社会连接数，用 d 表示特征空间维度. 那么，每次迭代过程中梯度的计算复杂度为 $O(\mathrm{d}N\bar{r}+\mathrm{d}N\bar{l}^2)$，通过降低 $\bar{l}$ 可以有效降低计算复杂度. 需要注意的是，Social MF 每次迭代的复杂度也为 $O(\mathrm{d}N\bar{r}+\mathrm{d}N\bar{l}^2)$，传统矩阵分解算法的复杂度仅为 $O(\mathrm{d}N\bar{r})$.

2 实验设计及结果分析

2.1 数据集合及评价标准

为了测试 LSMF 算法的有效性，本文采用 Epinions、Netflix 和 MovieLens 1M 三种推荐系统中最为常用的真实数据作为测试集合.

Epinions 数据集合包含了 49 290 名用户对 139 783个项目的664 824条评分和487 181条用户间的信任关系. Netflix 数据集合是 Netflix Prize 比赛中使用的标准测试数据集，本文从中随机抽取了包含8 662名用户对3 000部视频的约 30 万条评分信息作为测试集合，不包含社会连接信息. MovieLens 1M 数据集合由 GroupLens 提供，包含了6 039名用户对3 883部电影的一百多万条评分信息.

Netflix 和 MovieLens 1M 数据集合中未提供社会连接信息,可以用于测试 LSMF 算法 $\beta=0$ 的情况,以证明本文提出的 Logistic 映射方法能够有效提高预测准确性. 采用检验推荐算法最常用的预测误差 RMSE 作为评价依据,预测误差越小则表示算法性能越好.

$$\sigma_{\mathrm{RMSE}}=\sqrt{|S_{\mathrm{test}}|^{-1}\sum_{(i,j)\in S_{\mathrm{test}}}\|R_{ij}^{*}-R_{ij}\|^{2}}. \quad (9)$$

式中:S_{test} 为测试集合;$|S_{\mathrm{test}}|$ 为 S_{test} 中的元素个数.

2.2　实验设计

本文设计了 4 组实验从不同方面对 LSMF 算法进行测试. A 组实验较为全面对比 MF、Social MF、LSMF 三种算法的预测准确性;B 组实验测试社会化影响因子 β 的取值对 LSMF 算法性能的影响;C 组实验在 $\beta=0$ 的情况下对比 MF 和 LSMF,验证 Logistic 非线性映射的有效性;D 组实验测试在 $\beta=0$ 时参数 d 对 LSMF 算法性能的影响. 由于每次得到的 RMSE 都已是大量数据的平均,所以每组实验只重复 10 次,取 10 次平均值作为最终实验结果.

A 组实验在 Epinions 数据集合上随机选取 $x\%$ ($x=20,50,80$)的数据作为训练集合,其余作为测试集合. 在不同的特征空间维度 $d=\{5,10,20,50\}$ 的情况下,对比了 MF、Social MF 和 LSMF 算法的预测准确性. MF 算法中正则化系数 $\lambda_U=\lambda_V=0.02$,学习率 $lr=0.005$[12];Social MF 中算法中正则化系数 $\lambda_U=\lambda_V=0.1$,社会化影响因子 $\beta=0.05$,学习率 $lr=0.005$[19];LSMF 分别选择 $n_0=0$ 和 $n_0=1$,其余参数为:$\lambda_U=\lambda_V=0.1$,$\lambda_P=\lambda_Q=0.02$,$\beta=0.05$,$lr=0.005$.

实验结果如表 1 所示,最小预测误差的结果用黑体标注. 在不同训练比例、不同特征维度情况下,LSMF 总能取得最准确的预测结果. 在训练比例为 50%、特征维度为 20 的情况下,LSMF 的预测误差 RMSE 比 MF 减小约 6%,比 Social MF 减小约 4%. 而且,训练比例 $x\%$ 越低、特征维度 d 越小,LSMF 算法的优势越明显,这说明利用社会关系能够明显缓解数据稀疏性问题,LSMF 算法比 Social MF 算法提取潜在特征的能力更强. 随着训练集合密度和特征维度的增大,LSMF 预测误差逐渐降低.

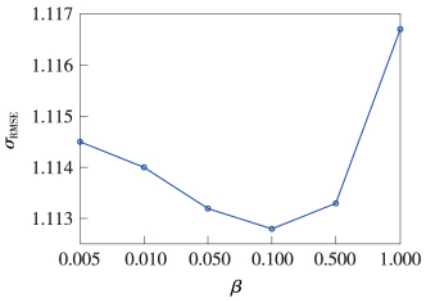
从表中还可以看出 n_0 取值为 0 或者 1 对 LSMF 性能影响不大,n_0 取值为 0 时性能略好,而且此时用户的平均社会连接数 $\bar{l}$ 变小,能够显著降低计算复杂度.

表 1　LSMF 与 Social MF、MF 在 Epinions 上 RMSE 的对比

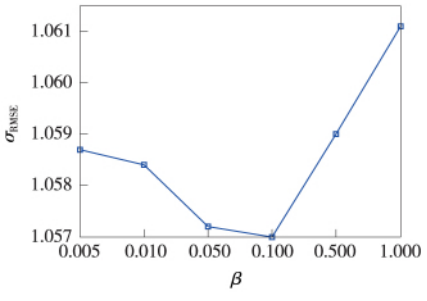
Tab. 1　Comparison of LSMF, Social MF, and MF on Epinions

训练比例/%	维度	MF	Social MF	LSMF($n_0=1$)	LSMF($n_0=0$)
20	5	1.594 3	1.178 8	**1.113 1**	1.113 2
	10	1.327 3	1.175 4	1.109 8	**1.108 8**
	20	1.200 1	1.172 7	1.106 4	**1.105 9**
	50	1.181 7	1.170 7	1.104 0	**1.103 7**
50	5	1.373 1	1.119 7	**1.075 6**	**1.075 6**
	10	1.209 2	1.117 8	1.072 8	**1.072 5**
	20	1.144 0	1.115 5	**1.070 3**	**1.070 3**
	50	1.115 4	1.113 6	1.067 8	**1.067 6**
80	5	1.286 7	1.095 5	1.057 9	**1.057 5**
	10	1.162 5	1.093 0	1.056 0	**1.055 2**
	20	1.119 2	1.091 2	1.053 3	1.053 5
	50	1.090 7	1.089 5	1.050 6	**1.050 5**

B 组分别从 Epinions 数据集中随机选取 20% 和 80%为训练集合,其余作为测试集合. 测试了社会化影响因子 β 的取值对 LSMF 算法性能的影响,实验中 LSMF 的特征维度为 5. 实验结果如图 2 所示,可以看出随着 β 的逐渐增大,社会关系的影响力逐渐增大,LSMF 的预测误差先变小后变大,在 $\beta=0.1$ 左右时 LSMF 算法能够取得较好的性能.



(a) 20%训练比例



(b) 80%训练比例

图 2　社会关系影响因子 β 对 LSMF 性能的影响

Fig. 2　Impacts of social factor β on performances of LSMF

C 组实验在 MovieLens 和 Netflix 数据集合上选取不同比例的训练集合,对比了 LSMF 和 MF 的预测准确性. MF 算法的参数为:$\lambda_U=\lambda_V=0.02$,$lr=0.005$,$d=20$;LSMF 的参数为 $n_0=0$,$d=20$,$\lambda_U=\lambda_V=0.02$,$\lambda_P=\lambda_Q=0.02$,$\beta=0$,$lr=0.005$.

如图3、图4所示，LSMF的预测误差明显低于MF，训练结合越稀疏，优势越明显. 用10%的MovieLens作为训练集合时，LSMF的预测准确性比MF高3%左右；用50%的Netflix作为训练集合时，LSMF能提高约5%预测准确性. 这说明了Logistic函数能够表征潜在因子之间的非线性关系，提高了矩阵分解模型提取潜在特征的能力.

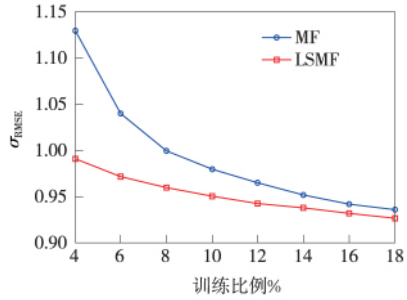


图3 MovieLens上LSMF与MF的对比
Fig. 3 Comparisons of LSMF and MF on MovieLens

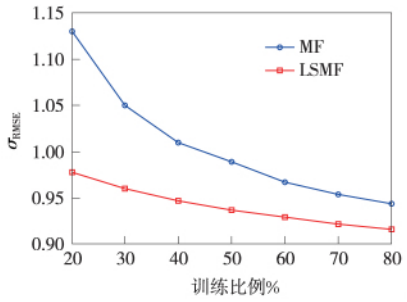


图4 Netflix上LSMF与MF的对比
Fig. 4 Comparisons of LSMF and MF on Netflix

D组实验分别在MovieLens和Netflix数据集合上测试特征维度d的取值对LSMF算法性能的影响. 分别从MovieLens和Netflix数据集合中选择10%、50%作为训练集合，LSMF算法的其他参数同C组实验. 从图5中可以看出，在两种数据集合上LSMF算法的预测误差随着特征维度的增大而逐渐减小，体现了LSMF算法性能的稳定性.

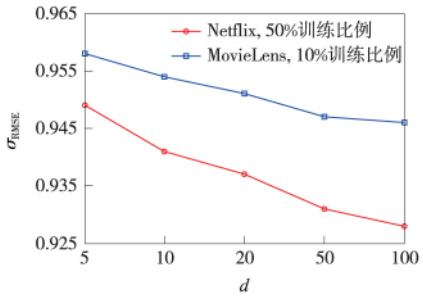


图5 特征维度d对LSMF性能的影响
Fig. 5 Impacts of feature dimension d on LSMF

3 结束语

本文提出了一种利用Logistic函数和社会关系信息，从非线性和社会化两个方面对矩阵分解推荐算法进行了改进. 在3组真实数据集合上的实验表明，本文提出的LSMF算法能够明显提高预测准确性，缓解数据稀疏性问题.

参考文献：

[1] Zhang Z K, Zhou T, Zhang Y C. Tag-aware recommender systems: a state-of-the-art survey[J]. Journal of Computer Science and Technology, 2011, 26(5): 767-777.

[2] LÜ Linyuan, Medo M, Yeung C H, et al. Recommender systems[J]. Physics Reports, 2012, 1(3): 159-172.

[3] 刘建国，周涛，汪秉宏. 个性化推荐系统的研究进展[J]. 自然科学进展，2009，19(1)：1-15.
Liu Jianguo, Zhou Tao, Wang Binghong. Advances in personalized recommender system[J]. Progress in Natural Science, 2009, 19(1): 1-15. (in Chinese)

[4] Cacheda F, Carneiro V, Fernandez D, et al. Comparison of collaborative filtering algorithms: limitations of current techniques and proposals for scalable, high-performance recommender systems[J]. ACM Transactions on the Web (TWEB), 2011, 5(1): 2.

[5] Bellogín A, Cantador I, Díez F, et al. An empirical comparison of social, collaborative filtering, and hybrid recommenders[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2013, 4(1): 14.

[6] Bobadilla J, Ortega F, Hernando A, et al. Recommender systems survey[M]. [S. l.]: Knowledge-Based Systems, 2013.

[7] Jamali M, Ester M. A matrix factorization technique with trust propagation for recommendation in social networks[C]// Proceedings of the fourth ACM Conference on Recommender Systems. [S. l.]: ACM, 2010: 135-142.

[8] Ma H, Zhou D, Liu C, et al. Recommender systems with social regularization[C]// Proceedings of the Fourth ACM International Conference on Web Search and Data Mining. [S. l.]: ACM, 2011: 287-296.

（责任编辑：刘芳）