Journal of Computer Applications
计算机应用,2015,35(S2):122 - 124,127
ISSN 1001-9081
CODEN JYIIDU
2015-12-15
http://www.joca.cn

文章编号:1001-9081(2015)S2-0122-03

基于主题模型的矩阵分解推荐算法

林晓勇*,代苓苓,史晟辉,李　芳
(北京化工大学 信息科学与技术学院,北京 100029)
(*通信作者电子邮箱 linxy@mail.buct.edu.cn)

摘　要:针对协同过滤算法存在的数据稀疏和忽视用户喜好多主题的问题,提出了基于主题模型的矩阵分解推荐算法,将标签、主题模型融合到了矩阵分解模型当中。该方法首先根据物品的标签提取物品的主题特征,用物品主题特征向量表达该物品,然后通过相似度计算方法得到每个物品的最近邻,最后用基于最近邻的正则化项来改进矩阵分解模型。在实验分析中,选择了不同的主题数进行比较,并且在潜在因子数不同的情况下,对比了该算法和潜在因子模型、正则化奇异值分解推荐算法。实验结果表明,改进算法能够降低预测评分的均方根误差,提高评分预测的准确度。

关键词:推荐系统;标签;主题模型;矩阵分解;正则化项
中图分类号:TP181　　**文献标志码**:A

Matrix factorization recommendation based on topic model

LIN Xiaoyong*, DAI Lingling, SHI Shenghui, LI Fang
(*College of Information Science and Technology, Beijing University of Chemical Technology, Beijing* 100029, *China*)

Abstract: In order to solve the problems of data sparseness and neglecting the multi-themed of user preferences existed in collaborative filtering algorithm, a kind of matrix factorization recommendation algorithm based on the topic model was proposed. This method integrated label and topic model into matrix factorization model. First, tags of item were classified as topics. Then the similarity among items was calculated according to the topics of items. Last the nearest neighbors of each movie was found using particular similarity calculation method. The information of nearest neighbors was used as regularization term to improve the matrix factorization model. Different number of topics was used to compare this method with other recommendation algorithms using different latent factor numbers. The experimental result shows that the proposed method is able to reduce the root mean squared error between predicted ratings and true ones, and improve the accuracy of predicted rating.

Key words: recommendation system; label; topic model; matrix factorization; regularization term

0　引言

随着互联网日益发展成熟,网络上的信息量越来越庞大,人们步入了大数据时代。在这个时代,用户从大量数据中找到自己感兴趣的信息是一件非常困难的事情。为了解决这个问题,推荐系统应运而生。推荐系统不仅能够帮助用户找到感兴趣的信息,而且能够将信息展现在对它感兴趣的用户面前,从而实现了信息使用者和信息提供者的双赢。

当前存在的两种主要的推荐算法分别是基于内容的推荐和协同过滤推荐。基于内容的推荐算法根据待推荐物品的信息元数据,发现物品之间的相关性,然后基于用户的历史记录,将相似的物品推荐给用户。Park 等[1]对基于内容的推荐算法在计算最近邻方面进行了改进,在不失准确率的前提下提高了时间效率,但该算法仍存在数据稀疏的问题。随着 Web2.0 技术的发展,Web 站点更加提倡用户参与,在这种情况下基于协同过滤的推荐算法被广泛应用,该算法根据用户对物品的偏好发现物品的相关性。近年来在推荐领域非常流行的矩阵分解模型是协同过滤算法的一种。该模型具有良好的可扩展性,能够融合大量的对推荐结果有帮助的特征信息。例如,Krasaoshchok 等[2]通过内容上下文来扩展模型,Manzato 等[3]利用用户的潜在反馈信息进行扩展,Ma 等[4]提出了带有社交网络特征的矩阵分解模型。这些方法对矩阵分解模型的正则项进行了扩展,均取得了很好的效果。矩阵分解的扩展模型不仅能够在一定程度上弥补用户-物品评分矩阵稀疏的问题,而且可以从多个方面兼顾用户的喜好。

标签在推荐系统中的作用日益显现。Kim 等[5]提出了结合标签系统和推荐系统的混合框架。该框架首先利用标签系统帮助用户标注高质量的标签,然后再利用推荐系统将与标签相关度高的物品推荐给用户。Zhang 等[6]总结了基于标签的推荐系统的最新进展,指出基于主题的方法能够产生更加便于解释的推荐结果。有学者将标签系统和主题模型相结合[7],对标签按照主题进行分类,不仅兼顾了用户喜好的多主题性,而且解决了标签词汇表数量巨大、不完整、不一致的问题。

基于以上矩阵分解模型和标签主题模型的优点,本文利用潜在狄利克分配(Latent Dirichlet Allocation, LDA)模型对标签按照主题分类,提出了基于主题模型的矩阵分解推荐算法,该推荐方法不仅考虑到了用户喜好的多主题性,而且能够

收稿日期:2015-03-26;**修回日期**:2015-05-07。　**基金项目**:中央高校基本科研业务费资助项目(JD1413)。
作者简介:林晓勇(1979 -),男,福建浦城人,副教授,博士研究生,主要研究方向:基于 Web2.0 的 SNS、数据挖掘;　代苓苓(1986 -),女,山东泰安人,硕士研究生,主要研究方向:人工智能、数据挖掘;　史晟辉(1974 -),女,河北河间人,副教授,博士研究生,主要研究方向:大数据分析、编译技术、生物信息、自然语言处理;　李芳(1977 -),河北安国人,高级工程师,博士研究生,主要研究方向:智能信息处理。

解决稀疏的评分矩阵面临的问题。

1　相关研究

1.1　基于物品的矩阵分解模型

矩阵分解是协同过滤算法中一种十分有效的方法,与传统的算法相比它可以带来更好的推荐结果。在 2006 年 Netflix Prize 比赛开始后,Simon Funk 公布了潜在因子模型(Latent Factor Model, LFM)算法[9]。由于其较高的准确度和易扩展性,后来被 Koren 和其他一些研究者加以改进[10-11]。矩阵分解模型是一种潜在因子模型,其基本原理是将表示用户-物品评分的矩阵 $\boldsymbol{R}$ 分解成两个矩阵,即用户因素矩阵 $\boldsymbol{P}=[\boldsymbol{p}_1,\boldsymbol{p}_2,\cdots,\boldsymbol{p}_n]$ 和物品因素矩阵 $\boldsymbol{Q}=[\boldsymbol{q}_1,\boldsymbol{q}_2,\cdots,\boldsymbol{q}_m]$,其中 $\boldsymbol{p}_u$ 和 $\boldsymbol{q}_i$ 是 $K*1$ 的向量,而 K 是因素个数。那么,预测评分可以通过用户特征向量 $\boldsymbol{p}_u$ 和物品特征向量 $\boldsymbol{q}_i$ 的点乘近似得到,如式(1)所示:

$$r_{ui}\approx \boldsymbol{p}_u^{\mathrm{T}}\boldsymbol{q}_i=\sum_{k=1}^{K}p_{ku}\times q_{ki} \tag{1}$$

1.2　LDA 主题模型

LDA 是一种生成主题模型,在 2003 年由 Blei 等提出[8]。它可以将文档集中的每篇文档的主题以概率分布的形式给出,从而得到文档的主题(分布),根据主题进行主题聚类或文本分类。在文本分析中,LDA 模型除了兼顾了文本的多主题性之外,还具有降维的作用,可以实现高质量的文本特征提取,完整准确地描述文本的内容。对于文档数据集,经过 LDA 生成主题,就可以利用主题向量来表示每篇文档,主题概率作为每个主题的权重。

1.3　相似度计算方法

相似度的计算是近邻模型的关键,可以根据具体情况选择不同的相似度计算方法,常用的方法包括皮尔逊相关系数、Jaccard 系数和余弦相似度。在本文中采用余弦相似度算法来计算电影之间的相似度,它利用多维空间两点与所设定的点形成夹角的余弦值来表示相似度,它的取值范围为 -1 ~ 1,值越大说明夹角越大,两点相距就越远,相似度就越小。公式如下所示:

$$sim(u_1,u_2)=\cos(n_1,n_2)=\frac{n_1\times n_2}{|n_1|\times|n_2|} \tag{2}$$

2　基于标签主题模型的矩阵分解推荐算法

本文采用的方法将 LDA 主题模型融合进了矩阵分解模型。该算法首先提取与物品相关度较高的标签,将每个物品的标签集合看作一篇文档,这样所有的物品组成一个文档数据集。然后通过 LDA 得到每篇文档的主题分布,也就是每个物品的主题分布。最后根据物品的标签主题求得物品的最近邻,并将其融合到矩阵分解模型中。

2.1　寻找物品基于标签的主题特征

虽然目前很多网站的推荐系统均提供了标签系统。但是随着网站上信息的巨量增长,标签词汇表也存在词汇数量大、不完整、不一致的问题。针对这个问题,我们可以人工对物品与标签的相关度进行评分,保留相关度高的标签。这样一个物品就可以用一个关联度向量来表示,用余弦相似度来求物品间的相似度,从而将相似度高的物品推荐给用户。但是,对于一个网站来说,标签的数量往往是很大的,如果一个电影推荐系统有十万个电影和一万个标签,计算相似度时需要处理的数据量就为十亿,计算量是相当大的。为了解决该问题,本文利用 LDA 模型将物品的标签集合按照主题分类,因为主题个数远远小于标签个数,这样把物品的向量从高维映射到低维,然后基于该低维向量进行相似度计算,可以大幅度降低运算量。

2.2　计算物品的相似度求最近邻

本文采用余弦相似度计算最近邻。在 2.1 节中通过 LDA 模型得到了每个物品的主题特征,用主题向量 $\boldsymbol{d}_i$ 来表示物品 i,$w_k^{(i)}$ 表示物品 i 属于主题 k 的概率,则基于主题求物品的最近邻如式(3)所示:

$$sim(\boldsymbol{d}_i,\boldsymbol{d}_j)=\frac{\sum_{k=1}^{N}\left(w_k^{(i)}\times w_k^{(j)}\right)}{\sqrt{\sum_{k=1}^{N}\left(w_k^{(i)^2}\right)\times\sum_{k=1}^{N}\left(w_k^{(j)^2}\right)}} \tag{3}$$

2.3　基于最近邻的正则化项

矩阵分解模型是一种协同过滤算法模型,它从用户喜好的数据集中训练出一个推荐模型,然后根据实时的用户喜好信息进行预测。该模型不仅能带来很好的推荐结果而且有非常好的扩展性,能融合各种特征。融合之前的首要工作是对各类信息进行充分分析,挑选出对提升推荐结果有帮助的信息作为特征,然后将这些特征融合到矩阵分解模型中。

标签是推荐的一种先验知识,对于具有相似标签特征的物品,用户评分也更加相似。因此,可以利用求出的物品最近邻来正则化模型,通过分析得出的正则化项如式(4)所示:

$$\frac{\partial}{2}\sum_{u\in U}\sum_{j\in R^k(i)}W_{ij}\|\boldsymbol{q}_i-\boldsymbol{q}_j\|_F^2 \tag{4}$$

其中:W_{ij} 为物品 i 和物品 j 基于主题模型计算出的的相似度(即式(3)求出的结果)。用这个正则化项用来惩罚两个相似物品的潜在特征向量之间的距离。$R^k(i)$ 表示基于 LDA 主题模型得到的物品 i 的最近 k 个邻居的集合。这个正则化项能够提升评分预测的准确度,并且不会给矩阵参数的优化过程增加额外的计算时间。

本文利用式(5)来计算模型的评分预测值。其中:μ 表示训练集中所有记录的全局平均分;b_i 为物品偏置项,表示物品接受的评分中和用户没有什么关系的部分;b_u 为用户偏置项,表示了用户的评分习惯中和物品没有关系的部分。

$$\tilde{r}_{ui}=\boldsymbol{p}_u\cdot\boldsymbol{q}_i+\mu+b_u+b_i \tag{5}$$

公式中参数的求解可以通过优化公式(6)所示的损失函数来实现,目的是使得分解之后的两个矩阵的乘积与原始的评分矩阵的均方根误差最小。

$$\min_{\boldsymbol{P},\boldsymbol{Q}}L=\sum_{(u,i)\in\kappa}(r_{ui}-\tilde{r}_{ui})^2+\lambda_1\sum_{j\in R^k(i)}W_{ij}\|\boldsymbol{q}_i-\boldsymbol{q}_j\|_F^2+\lambda_2(\|\boldsymbol{P}\|_F^2+\|\boldsymbol{Q}\|_F^2+\|b_u\|_F^2+\|b_i\|_F^2) \tag{6}$$

采用随机梯度下降算法对该损失函数的参数进行学习,根据随机梯度下降算法,该模型的参数更新规则如公式(7)所示,利用该模型训练得到正确参数,将参数代入预测公式即可预测评分。

$$\begin{cases}\boldsymbol{p}_u\leftarrow\boldsymbol{p}_u+\eta\cdot(e_{ui}\boldsymbol{q}_i-\lambda\boldsymbol{p}_u)\\ \boldsymbol{q}_i\leftarrow\boldsymbol{q}_i+\eta\cdot(e_{ui}\boldsymbol{p}_u-\lambda\boldsymbol{q}_i-\partial\sum_{j\in R^k(i)}w_{ij}(\boldsymbol{q}_i-\boldsymbol{q}_j))\\ \boldsymbol{q}_j\leftarrow\boldsymbol{q}_j+\eta\cdot\partial\cdot w_{ij}(\boldsymbol{q}_i-\boldsymbol{q}_j)\\ b_u\leftarrow b_u+\eta(e_{ui}-\lambda b_u)\\ b_i\leftarrow b_i+\eta(e_{ui}-\lambda b_i)\end{cases} \tag{7}$$

3 实验及结果分析

3.1 数据集

本文使用 MovieLens 站点提供的数据集对算法进行评估,该站点由美国 Minneresota 大学的 GroupLens 项目研究小组创建并维护。在实验中用到的数据集包括标签关联度 tag_relevance 数据集和用户评分 rating 数据集。Movielens 自支持标签开始,已经拥有了 31 325 个标签。

tag_relevance 数据集的数据格式为 < MovieID > < TagID > < Relevance >。其中:Relevance 为标签与电影的相关度,其值在 0 和 1 之间,值越大表示 tag 与 movie 的关联性越强。在进行实验前,对该数据集进行预处理,找出关联度在 0.5 以上的记录,即找出最能代表电影属性的标签词语。预处理之后得到的记录每一行即为一个“文本”的概念,整个数据集作为一个“文本集合”。利用 LDA 模型对该数据集进行建模,得到文本的主题概率,进而计算得到得基于主题模型的电影相似度。Ratings 数据集的格式为 < userID > < MovieID > < Rating > < Timestamp >,其中 Rating 评分的范围为 1 ~ 5,评分越高表示用户越满意。在实验中,按照 86% 和 14% 的比例将该数据集划分为训练集和测试集。

3.2 度量标准

推荐算法要通过评测指标才能评估算法的推荐效率。推荐系统评测指标的问题吸引了很多研究人员的研究[12-14]。评分预测的目的是希望能够给用户一个分数,表明我们认为用户是否会喜欢这部电影,而这个分数也可以帮助用户决策是否要看这部电影,如果用户看了这部电影,那么就产生了推荐的效果。在评分预测系统中常采用均方根误差(Root Mean Squared Error, RMSE)作为度量标准,RMSE 越小,则推荐质量就越高。RMSE 的求解公式如下:

$$RMSE = \sqrt{\frac{\sum_{(u,i)\in T}(r_{ui} - \hat{r}_{ui})^2}{|Test|}} \tag{8}$$

其中:T 为测试集合,r_{ui} 为实际评分,$|Test|$ 为测试记录的数目。

3.3 实验结果

本文通过实验验证改进算法在评分预测问题中的效果。算法主要的参数有主题模型的主题个数 t、潜在因子个数 f、学习速率 *alpha* 和正则化参数 *lambda*。学习速率 *alpha* 如果选择太小收敛太慢,选择太大则可能跳过收敛的最优解,正则化项的惩罚因子 *lambda* 如果选择合适,则能较好地防止过拟合。这两个参数对算法性能的影响相对较小。为了更好地说明测试结果与 LDA 主题个数、矩阵分解的潜在因子个数和均方根误差 *RMSE* 的关系,进行了多组实验的比较。在实验中参数的值固定为:*alpha* = 0.005,*lambda*1 = 0.001,*lambda*2 = 0.003,在训练集上迭代次数为 15。

3.3.1 标签主题个数比较

因为 LDA 模型的性能受到主题数目取值的影响,所以要事先确定好主题数目 t。在实验中将改进算法的 t 分别取 20, 40, 60, 80, 100, 120, 140, 160, 180, 200 时检测 *RMSE* 的变化。由于随着 f 的增大,计算量也随之变大,为了节省计算时间,在这里将 f 的值固定为 300。实验结果如图 1 所示。

由图 1 可以看出,当主题数目 t 为 120 时,*RMSE* 最小,对电影推荐产生最佳效果,所以在下面的实验中选择主题数 t 值为 120。

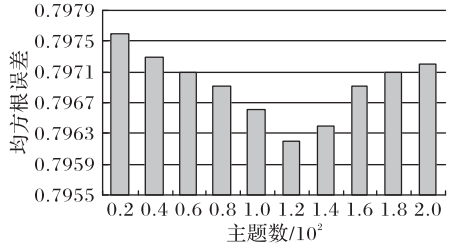

图 1 主题个数与 *RMSE* 的关系

3.3.2 矩阵分解的潜在因子比较

当主题个数为 120 时,比较评测结果与矩阵分解的潜在因子个数 f 的关系。检测 *RMSE* 在 f 取 50, 100, 150, 200, 250, 300, 350, 400, 450, 500 时的变化。实验表明 f 值越大,实验精度越高,产生的推荐效果越好。实验结果如图 2 所示。

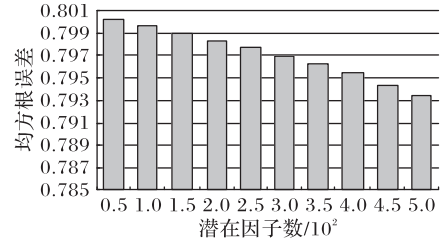

图 2 潜在因子个数与 *RMSE* 的关系

3.3.3 推荐算法比较

改进模型由基本的 LFM 模型扩展而来,与基本的 LFM 模型对比,更能够直接地体现出改进模型在推荐准确度上的提高。其次,正则化奇异值分解模型(Regularized Singular Value Decomposition, RSVD)[15]是一种经典的基于 LFM 改进的模型。而针对于 LFM 改进的模型很多,在这里不能一一列举,所以在本文的实验中就选取了这两种比较有代表性的模型作为实验的比较对象。

当主题个数为 120 时,比较了改进算法和 LFM 模型、RSVD 模型在 f 取 50, 100, 150, 200, 250, 300, 350, 400, 450, 500 时 *RMSE* 的变化,对比结果如图 3 所示。

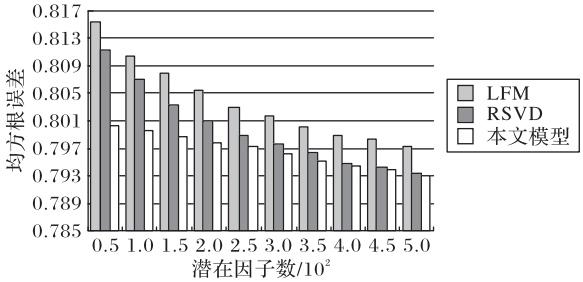

图 3 各种算法的比较

从图 3 可以看出矩阵分解模型的 *RMSE* 随着 f 值的增加而单调下降。由实验数据可以得出,基于 LFM 和 RSVD 的算法得到的误差要高于本文中改进的模型,从而说明了改进算法在融合了基于 LDA 的主题模型后,得到推荐准确度要优于其他两种模型。

4 结语

本文通过融合基于标签主题特征的物品最近邻,改进了矩阵分解模型。将物品的标签按主题进行分类,解决了目前推荐系统中存在的忽视用户喜好多主题性的问题。利用物品的最近邻信息作为正则化项优化了矩阵分解模型,使得预测评分的准确度得到了一定程度的提升。但是该算法目前仍旧

(下转第 127 页)

及基于权值情感词分别进行实验,准确率分别为 61.39%、72.18% 和 81.06%。

结果表明,虽然情感词权值由情感词在训练样本中正、负样本占比,以及基础情感字典共同决定,但是,通过引入权值,其总体性能远远高于单独使用。此结果也说明,这种权值计算方法,符合预期。

3　结语

通过对情感词权值研究,提出了一种新的情感权值计算方法,并基于情感权值进行文本情感极性分析。由实验结果可知,这种方法能有效提升情感极性分析的准度。目前,只针对于情感词的权值进行研究分析,在后续的工作中,将在更大的粒度上进行权值研究,如在情感词基础上加入褒贬指向或者情感短语。

参考文献:

[1] 魏韡, 向阳, 陈千. 中文文本情感分析综述[J]. 计算机应用, 2011, 31(12):3321 – 3323.

[2] 朱嫣岚, 闵锦, 周雅倩, 等. 基于 HowNet 的词汇语义倾向计算[J]. 中文信息学报, 2006, 20(1): 14 – 20.

[3] 赵鹏, 赵志伟, 卓景文. 一种情感词语义加权的句子倾向性识别方法[J]. 计算机工程与应用, 2011, 47(35):161 – 163.

[4] LIU B. Sentiment analysis and opinion mining[J]. Synthesis Lectures on Human Language Technologies, 2012, 5(1): 1 – 167.

[5] TABOADA M, BROOKE J, TOFILOSKI M, *et al.* Lexicon-based methods for sentiment analysis[J]. Computational Linguistics, 2011, 37(2): 267 – 307.

[6] PANG B, LEE L, VAITHYANATHAN S. Thumbs up?: sentiment classification using machine learning techniques[C]// Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing. Stroudsburg: Association for Computational Linguistics, 2002: 79 – 86.

[7] ABBASI A, CHEN H, SALEM A. Sentiment analysis in multiple languages: feature selection for opinion classification in Web forums[J]. ACM Transactions on Information Systems, 2008, 26(3): 12.

[8] LIU J, SENEFF S. Review sentiment scoring via a parse-and-paraphrase paradigm[C]// Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. Stroudsburg: Association for Computational Linguistics, 2009,1 : 161 – 169.

[9] JIA L, YU C, MENG W. The effect of negation on sentiment analysis and retrieval effectiveness[C]// Proceedings of the 18th ACM Conference on Information and Knowledge Management. New York: ACM, 2009: 1827 – 1830.

[10] TITOV I, McDONALD R. Modeling online reviews with multi-grain topic models[C]// Proceedings of the 17th International Conference on World Wide Web. New York: ACM, 2008: 111 – 120.

[11] LI J, SUN M. Experimental study on sentiment classification of Chinese review using machine learning techniques[C]// NLP-KE 2007: International Conference on Natural Language Processing and Knowledge Engineering. Piscataway: IEEE, 2007: 393 – 400.

[12] 冯时, 付永陈, 阳锋, 等. 基于依存句法的博文情感倾向分析研究[J]. 计算机研究与发展, 2012, 49(11): 2395 – 2406.

(上接第 124 页)

还存在着一些问题和不足:首先该方法基于历史数据,对于新用户和新物品存在“冷启动”的问题;其次,推荐结果依赖于用户历史偏好数据的规模和准确度,如果用户偏好数据过于稀疏或者准确度不高,那么推荐结果也会出现一定的偏差。针对上述问题,还需要进一步的研究。

参考文献:

[1] PARK Y, PARK S, JUNG W, *et al.* Reversed CF: a fast collaborative filtering algorithm using a *k*- nearest neighbor graph[J]. Expert Systems with Applications, 2015, 42(8):4022 – 4028.

[2] KRASNOSHCHOK O, LAMO Y. Extended content-boosted matrix factorization algorithm for recommender systems[J]. Procedia Computer Science, 2014, 35: 417 – 426.

[3] MANZATO M G. gSVD ++: supporting implicit feedback on recommender systems with metadata awareness[C]// SAC 2013: Proceedings of the 28th Annual ACM Symposium on Applied Computing. New York: ACM, 2013: 908 – 913.

[4] MA H, ZHOU D, LIU C, *et al.* Recommender systems with social regularization[C]// WSDM 2011: Proceedings of the 4th ACM International Conference on Web Search and Data Mining. New York: ACM, 2011: 287 – 296.

[5] KIM H, KIM H J. A framework for tag-aware recommender systems[J]. Expert Systems with Applications, 2014, 41(8): 4000 – 4009.

[6] 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.

[7] BOGÁRDI-MÉS, RÖVID A, ISHIKAWA H, *et al.* Tag and topic recommendation systems[J]. Acta Polytechnica Hungarica, 2013, 10(6): 171 – 191.

[8] BLEI D M, NG A Y, JORDAN M I. Latent Dirichlet allocation[J]. Journal of Machine Learning Research, 2003, 3: 993 – 1022.

[9] KOREN Y, BELL R, VOLINSKY C. Matrix factorization techniques for recommender systems[J]. Computer, 2009, 42(8): 30 – 37.

[10] FUNK S. Netflix Update: Try this at home [EB/OL]. [2006-12-11]. http://sifter.org/~simon/journal/20061211.html.

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

[12] CELMA Ò, HERRERA P. A new approach to evaluating novel recommendations[C]// RecSys 2008: Proceedings of the 2008 ACM Conference on Recommender Systems. New York: ACM, 2008: 179 – 186.

[13] GOEL S, BRODER A, GABRILOVICH E, *et al.* Anatomy of the long tail: ordinary people with extraordinary tastes[C]// WSDM 2010: Proceedings of the 3rd ACM International Conference on Web Search and Data Mining. New York: ACM, 2010: 201 – 210.

[14] HERLOCKER J L, KONSTAN J A, TERVEEN L G, *et al.* Evaluating collaborative filtering recommender systems[J]. ACM Transactions on Information Systems, 2004, 22(1): 5 – 53.

[15] PATEREK A. Improving regularized singular value decomposition for collaborative filtering[C]// Proceedings of KDDCup 2007. New York: ACM, 2007: 5 – 8.