# 基于改进最近邻的协同过滤推荐算法

硕良勋[1],柴变芳[2],张新东[3]
SHUO Liangxun[1], CHAI Bianfang[2], ZHANG Xindong[3]

1.石家庄经济学院 网络信息安全实验室,石家庄 050031
2.北京交通大学 计算机与信息技术学院,北京 100044
3.河北新冀网络传媒有限公司,石家庄 050031
1.Network Security Laboratory, Shijiazhuang University of Economics, Shijiazhuang 050031, China
2.Institute of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China
3.Hebei Xinji Network Media Co., LTD, Shijiazhuang 050031, China

**SHUO Liangxun, CHAI Bianfang, ZHANG Xindong. Collaborative filtering algorithm based on improved nearest neighbors. Computer Engineering and Applications, 2015, 51(5):137-141.**

**Abstract:** Aiming to the problems that the quality and precision are caused by the sparse user scorings and cold-start, a novel collaborative filtering algorithm based on improved nearest neighbors is proposed in this paper. User-item matrix is established, and similarity between items and users is measured, the nearest neighbor of items and users is acquired, in which the particle swarm optimization algorithm is used to select the optimal value of the parameter *k*, the simulation experiments are carried out on MovieLens and Book-Crossing dataset. The results show that the proposed algorithm can achieve lower MAE and efficiently improve recommendation precision, and it can enhance the quality of recommendations.
**Key words:** collaborative filtering; improved nearest neighbor; particle swarm optimization algorithm; selecting parameters

**摘 要:**针对当前协同过滤推荐算法易受数据稀疏性与冷启动的问题,提出了一种改进最近邻的协同过滤推荐算法。建立用户-项目评分矩阵,并度量项目之间、用户之间的相似性,获取项目和用户的最近邻居,其中最近邻居的最优参数*k*值采用粒子群算法选择,在MovieLens和Book-Crossing数据集上进行了仿真对比实验。结果表明,相对于其他协同过滤推荐算法,该算法降低了平均绝对误差值,提升了推荐准确度,达到提高推荐质量效果的目的。
**关键词:**协同过滤;改进最近邻;粒子群优化算法;参数选择
**文献标志码:**A **中图分类号:**TP391 doi:10.3778/j.issn.1002-8331.1405-0236

随着电子商务的发展,服务商在保证产品质量的前提下,需要掌握用户的需求与偏好变化趋势,从而可以对用户提供按需服务,因此如何建立优秀的推荐模型是电子商务应用中的关键,成为研究者的关注热点[1-2]。

协同过滤推荐算法的基本思想是通过和目标用户具有相似兴趣偏好的其他用户对目标的项目进行评价,从而判断评价结果是否对用户有价值,从而决定是否将项目推荐给用户。目前有许多协同过滤推荐算法,其中最近邻协同过滤推荐应用最为广泛的技术之一,其通过评分相似多个最近邻居的评分向用户产生推荐[3]。然而,随着实际应用的开展,最近邻推荐技术存在许多不足,主要包括:(1)推荐结果易受数据稀疏性的影响;(2)冷启动问题[4]。为了解决该难题,许多学者从不同角度,不同方向对近邻推荐算法进行了相应的完善和改进,其中文献[5]通过用户间共同评分数据来增强其间相似度,缓解了评分数量过少的问题;文献[6]提出了将内容过滤与经典协同过滤加权融合的推荐算法,缓解了数据稀疏性的问题;文献[7]采用传播的思想从用户与项目两个角度为用户生成推荐,提高了推荐算法的精确度;文献[8]通过将联合聚类分析与传统协同过滤算法

**基金项目:**国家自然科学基金(No.61370129);河北省科技计划项目(No.13210702D)。
**作者简介:**硕良勋(1971—),男,博士,副教授,主要研究领域为网络信息安全;柴变芳(1979—),女,博士生,讲师,主要研究领域为社区发现、概率图模型;张新东(1983—),男,工程师,主要研究领域为新闻信息管理。
**收稿日期:**2014-05-20 **修回日期:**2014-07-02 **文章编号:**1002-8331(2015)05-0137-05
CNKI**网络优先出版:**2014-09-15, http://www.cnki.net/kcms/doi/10.3778/j.issn.1002-8331.1405-0236.html

结合,并采用聚类分析相关技术来获取用户的偏好,在一定程度上提高了推荐的精确度,然而这些算法均没有讨论最近邻参数 $k$ 对推荐结果的影响,均采用经验法确定 $k$ 的值,难以获得十分理想的推荐结果。

为了解决协同过滤算法易受数据稀疏性与冷启动的问题,针对最近邻参数 $k$ 优化问题,提出了一种基于改进最近邻协同过滤推荐算法,该算法通过粒子群算法对参数进行在线优化和选择,并通过在MovieLens和Book-Crossing数据集上进行仿真对比实验,以测试本文算法的优越性。结果表明,本文算法降低了平均绝对误差值,提升了推荐准确度,获得更加理想的推荐结果。

## 1　传统协同过滤推荐算法

### 1.1　建立用户-项目评分矩阵

在协同过滤推荐系统中,存在大量的用户-项目评分数据,数据结构中至少包含三个字段,分别表示项目编号 ($Item_i$) 、用户编号 ($User_j$) 和用户-项目评分 ($S_{ij}$) 。表1给出了用户-项目评价矩阵的数据示意,其中存在大量的未评价信息,体现了用户-项目评价矩阵的稀疏性[9]。

**表1　用户-项目评价矩阵数据示例**

| | $User_1$ | $User_2$ | … | $User_k$ | … | $User_M$ |
|---|---|---|---|---|---|---|
| $Item_1$ | $S_{11}$ | / | … | $S_{1k}$ | … | / |
| $Item_2$ | / | $S_{22}$ | … | / | … | $S_{2M}$ |
| … | … | … | … | … | … | … |
| $Item_p$ | — | $S_{p2}$ | …. | … | … | … |
| … | … | … | … | … | .. | … |
| $Item_N$ | $S_{N1}$ | … | … | . | .. | / |

### 1.2　计算相似度

为了获取项目和用户的最近邻居,需度量项目-项目相似性和用户-用户相似性,因此,相似性度量方法是最近邻协同过滤推荐过程中的关键技术之一。常有相似性度量方法具体如下:

(1)余弦相似度(Cosine Similarity)。用户 $a$ 与用户 $b$ 间余弦相似度、项目 $\alpha$ 与项目 $\beta$ 间余弦相似度的计算公式分别为:

$$sim(a,b)=\frac{\bar{\boldsymbol{a}}\cdot\bar{\boldsymbol{b}}}{\|\bar{\boldsymbol{a}}\|*\|\bar{\boldsymbol{b}}\|} \tag{1}$$

$$sim(\alpha,\beta)=\frac{\bar{\boldsymbol{\alpha}}\cdot\bar{\boldsymbol{\beta}}}{\|\bar{\boldsymbol{\alpha}}\|*\|\bar{\boldsymbol{\beta}}\|} \tag{2}$$

式中,$\bar{\boldsymbol{a}}$ 和 $\bar{\boldsymbol{b}}$ 分别表示用户 $a$ 和 $b$ 在 $m$ 维项目空间上所做出的评分值向量;$\bar{\boldsymbol{\alpha}}$ 和 $\bar{\boldsymbol{\beta}}$ 是项目 $\alpha$ 和 $\beta$ 在 $n$ 维用户空间上的评分值向量。

(2)Pearson相关系数(Pearson Correlation Coefficient)。用户 $a$ 与用户 $b$ 间余弦相似度、项目 $\alpha$ 与项目 $\beta$ 间余弦相似度的计算公式分别为:

$$sim(a,b)=\frac{\sum_{k=1}^{m}(v_{ak}-\bar{v}_a)(v_{bk}-\bar{v}_b)}{\sqrt{\sum_{k=1}^{m}(v_{ak}-\bar{v}_a)^2}\sqrt{\sum_{k=1}^{m}(v_{bk}-\bar{v}_b)^2}} \tag{3}$$

$$sim(a,b)=\frac{\sum_{k=1}^{n}(v_{ak}-\bar{v}_a)(v_{\beta k}-\bar{v}_\beta)}{\sqrt{\sum_{k=1}^{n}(v_{ak}-\overline{v_\alpha})^2}\sqrt{\sum_{k=1}^{n}(v_{\beta k}-\bar{v}_\beta)^2}} \tag{4}$$

式中,$\bar{v}_a=\frac{1}{m}\sum_{k=1}^{m}v_{ak}$ 表示用户 $a$ 对项目群进行打分的平均评分;$\bar{v}_\alpha=\frac{1}{n}\sum_{k=1}^{n}v_{ak}$ 表示项目 $\alpha$ 被用户群进行打分的平均评分。

### 1.3　选择$k$近邻

推荐系统的核心是为需要推荐服务的目标用户寻找“最相似用户”或“最近邻居”集(Nearest-neighbor),然后根据“邻居”的信息进行推荐。在协同过滤算法中,$k$ 最近邻确定主要通过特定的相似度算法进行,即选择与当前用户相似度最高的前 $k$ 个用户作为邻居。对于一个活动用户 $a$,要产生一个依相似度由大到小排列的邻居集合 $U=\{u_1,u_2,\cdots,u_k\}, a\notin U$ 。

### 1.4　产生推荐列表

设 $I_{ab}$ 为用户 $a$ 与用户 $b$ 的共同评分项目集合,用户 $a$ 与用户 $b$ 的任一最近邻用户,首先获取用户 $b$ 对于项目 $I_i(I_i\in I_{ab})$ 的预测评分如下:

$$P_{bi}=\bar{v}_a+\frac{\sum_{a\in kNN(b)}(S(a,i)-\bar{v}_a)\cdot sim(b,a)}{\sum_{u\in kNN(a)}|sim(b,a)|} \tag{5}$$

其中,$S(a,i)$ 表示用户 $a$ 对于项目 $I_i$ 的评分值,$sim(b,a)$ 表示用户 $b$ 与用户 $a$ 的相似度[10]。

## 2　改进近邻的协同过滤算法

### 2.1　问题分析

当前协同过滤推荐算法大多通过 $k$ 最近邻算法(KNN)确定当前用户感兴趣的项目,但是最近邻数 $k$ 到底取多少适合,当前主要通过经验选择,具有一定的盲目性,而不合理的 $k$ 值设置往往会大幅降低算法性能。为了解决该难题,本文选择粒子群优化算法选择最优 $k$ 值。

### 2.2　粒子群优化算法

粒子群优化算法模拟鸟类的群体飞行觅食行为,在 $D$ 维搜索空间中,第 $i$ 个粒子 $(i=1,2,\cdots,m)$ 的位置为 $Z_i=(z_{i1},z_{i2},\cdots,z_{iD})$, $V_i=(v_{i1},v_{i2},\cdots,v_{iD})$ 为粒子 $i$ 的位置移动距离, $P_i=(p_{i1},p_{i2},\cdots,p_{iD})$ 表示第 $i$ 个粒子“飞行”历史中最优位置, $P_g=(p_{g1},p_{g2},\cdots,p_{gD})$ 表示种群历史最优位置,粒子根据以下公式更新速度和位置:

$$v_{id}(t+1)=\omega\times v_{id}(t)+c_1\times rand()\times(p_{id}(t)-z_{id}(t))+ c_2\times rand()\times(p_{gd}(t)-z_{id}(t)) \quad (6)$$

$$z_{id}(t+1)=z_{id}(t)+v_{id}(t+1) \quad (7)$$

式中，$t$ 表示迭代次数；$c_1, c_2$ 为学习因子；$rand()$ 是 [0, 1] 之间的随机数；$\omega$ 是惯性权重。

### 2.3 粒子群优化算法选择最近邻 $k$ 值的步骤

**步骤** 1 建立用户-项目评分矩阵。

**步骤** 2 度量项目-项目相似性和用户-用户相似性。

**步骤** 3 随机产生 $m$ 个粒子，每一个粒子为最近邻的参数 $k$ 的值。

**步骤** 4 根据 $k$ 值产生推荐列表，计算算法的推荐精度，并将其作为粒子的适应度值。

**步骤** 5 更新每一个粒子的历史最优位置 ($P_{best}$) 和粒子群的历史最优位置 ($g_{best}$)。

**步骤** 6 根据式(6)和(7)对粒子的速度和位置进行更新。

**步骤** 7 如果满足算法结束条件，得到最优参数 $k$。

**步骤** 8 根据最优参数 $k$ 建立最优协同过滤推荐模型。

## 3 仿真实验

### 3.1 仿真环境

在Intel® Core™ 2 Duo CPU E7400，主频为2.8 GHz，2 GB内存，320 GB硬盘，操作系统为Windows professional sp3的计算机上，所用Visual C++语言实现仿真实验。采用常用度量推荐算法精度的公开数据集MovieLens作为仿真对象，选取其公开的部分数据集合，共包括943个用户的对于1 682部电影的100 000多条真实评分记录[11]。

### 3.2 对比算法及评价标准

为了使本文算法结果具有可比性，选择文献[12]、文献[13]算法进行对比实验。当前推荐系统的评测指标有很多种，其中最常用的有平均绝对误差(MAE)[14]和均方根误差(RMSE)[15]两类。$MAE(u)$ 等于目标用户 $u$ 的评分预测值与测试集中真实评分值偏差绝对值和的平均，整个推荐算法的MAE是所有用户MAE的平均，具体如下：

$$MAE(u)=\sum_{i=1}^{n}\left|r_{ui}-\hat{r}_{ui}\right|/n \quad (8)$$

$$MAE=\sum_{u=1}^{N}MAE(u)/N \quad (9)$$

式中，$n$ 为目标用户在测试集中已有的评分个数。

RMSE对大误差更为敏感，$RMSE(u)$ 等于目标用户 $u$ 的评分预测值与测试集中真实评分值偏差平方和的均方差，整个推荐算法的RMSE是所有用户RMSE的平均，具体如下：

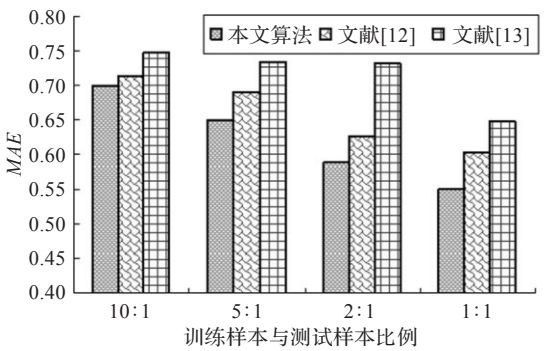
$$RMSE(u)=\sqrt{\sum_{i=1}^{n}\left(r_{ui}-\hat{r}_{ui}\right)^2}/n \quad (10)$$

$$RMSE=\sum_{u=1}^{N}RMSE(u)/N \quad (11)$$

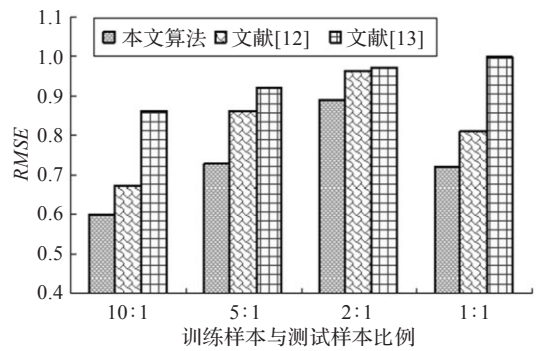
### 3.3 结果与分析

#### 3.3.1 推荐精度比较

通过粒子群优化算法找到最优参数 $k=20$。不同测试集和训练集条件下，不同算法实验结果如图1所示。从图1可知，相对于对比算法，本文算法的推荐精确度相对较高，获得了更好的推荐结果。对比结果表明，本文算法通过粒子群算法选择最优 $k$ 值，建立了基于最优近邻的协同过滤推荐模型，有效降低推荐误差，可以获得满足用户需要的推荐结果。




(a)MAE性能对比




(b)RMSE性能对比

**图1　不同算法实验对比结果**

#### 3.3.2 冷启动条件下的性能对比

为了测试本文算法的冷启动性能，从数据集中随机抽取20个用户，删除其相关的评分信息，模拟冷启动情景，本文算法与对比算法的MAE变化曲线如图2所示。从图2可知，相对于对比算法，本文算法较好地解决了冷启动的难题，获得更小的MAE值，提高了推荐精度，具有一定的优越性。

#### 3.3.3 不同稀疏度条件下的性能对比

从数据集中随机去除部分评分数据，提高数据稀疏度，本文算法与对算法的MAE的变化曲线如图3所示。从图3可知，当数据极端稀疏时，全部算法的平均绝对误差(MAE)上升，然而在同等条件下，本文算法的

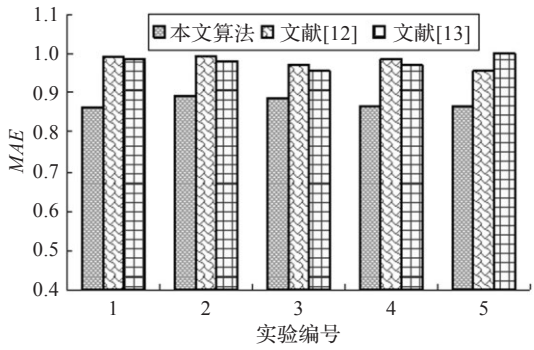



**图2　冷启动条件下的算法性能对比**

MAE值最小,对比结果表明,本文算法可以适应不同稀疏度的数据,鲁棒性更强。

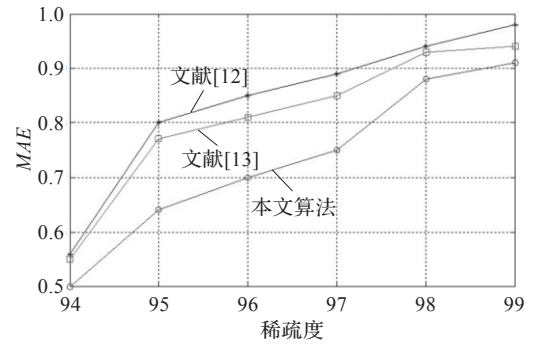



**图3　不同稀疏度下的算法性能对比**

### 3.3.4　迭代步数对算法计算效率的影响

为了测试本文协同过滤推荐算法计算效率,分析迭代步数对算法性能的影响,实验结果如图4和5所示。从图4可以看出,当迭代步数处于1~7时,MAE不断减少,推荐精度得以提高,大迭代步数 $n>7$ 后,算法MAE变化比较平稳,变化幅度比较小。从图5可以知道,RMSE的整体变化趋势与MAE十分相似,以较少的迭代步数可以获得较高的计算效率。

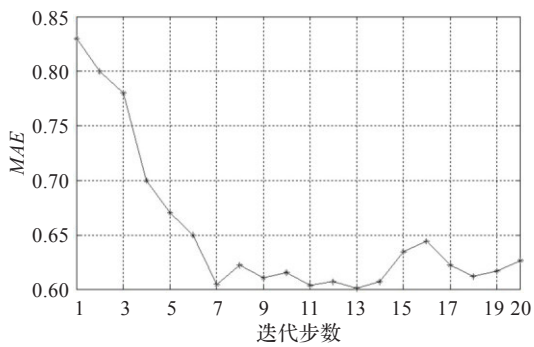



**图4　不同迭代步数的MAE变化曲线**

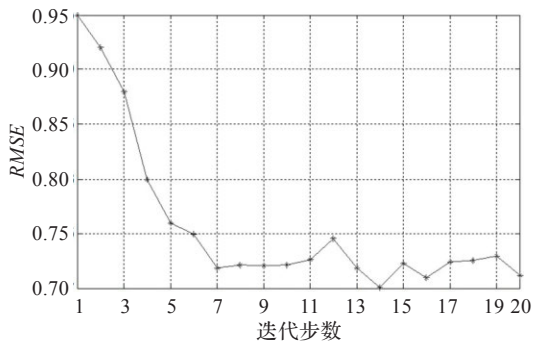
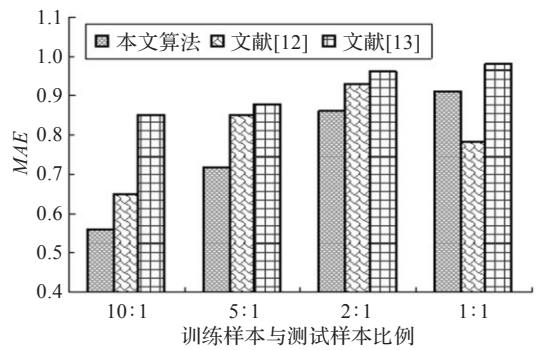


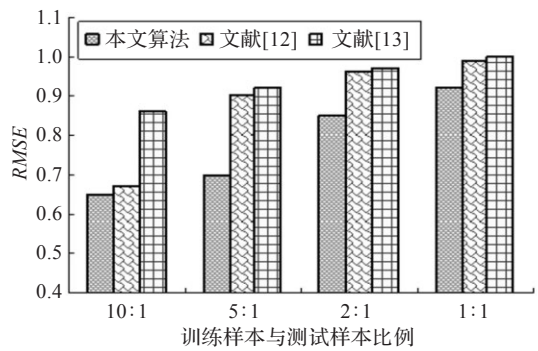

**图5　不同迭代步数的RMSE变化曲线**

### 3.3.5　与其他数据集上的实验结果

为了进一步测试本文算法的性能,采用Book-Crossing数据集进行仿真实验,该数据集由Cai-Nicolas Ziegler通过Book-Crossing网站(http://www.bookcrossing.corn)收集,包含了278 858个用户对271 379本书的1 149 780条评分信息,所有的评分值分布在[0,10]区间内,越高的评分值代表越强的用户兴趣。为了便于实验,对评分值进行重新标定,将评分值为9、10的标定为+1,将评分值为0~8的标定为−1,在Book-Crossing数据集中前1 000个项目的被评分数据上进行仿实验。该子数据集总共包含约3.5万名用户在这1 000个项目上超过14万条的评分数据。不同算法的实验结果如图6所示。从图6可知,本文算法在MAE和RMSE性能指标上都明显优于对比算法,具有较高的推荐准确度,再一次证明了本文算法的有效性和优越性。




(a)MAE性能对比




(b)RMSE性能对比

**图6　不同算法对Book-Crossing数据集的性能对比**

## 4　结束语

为了解决当前协同过滤算法受数据稀疏度的影响较大,且面临冷启动问题,针对最近邻参数选择问题,提出一种改进最近邻协同过滤推荐算法。仿真实验结果表明。本文算法提高推荐精确度与推荐质量,较好地解决了传统最近邻协同过滤推荐算法存在的缺陷,大幅度提高算法的效率,可以应用于电子商务推荐系统中。

## 参考文献:


[1] 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.

[2] Candillier L,Meyer F,Fessant F.Designing specific weighted similarity measures to improve collaborative filtering systems[C]//Proc of the Industrial Conference on Data Mining,2008,50(77):242-255.

[3] Bell R,Koren Y,Volinsky C.Modeling relationships at multiple scales to improve accuracy of large recommender systems[C]//Proc of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,2007:95-104.

[4] Kim D,Yum B J.Collaborative filtering based on iterative principal component analysis[J].Expert Systems with Applications,2005,28(4):823-830.

[5] 熊忠阳,刘芹,张玉芳.结合项目分类和云模型的协同过滤算法[J].计算机应用研究,2012,29(10):3660-3664.

[6] 陈健,印鉴.基于影响集的协作过滤推荐算法[J].软件学报,2007,18(7):1685-1694.

[7] 冯永,陈显勇.基于评分信息量的协同过滤算法研究[J].计算机工程与应用,2013,49(20):198-201.

[8] 赵琴琴,鲁凯,王斌.SPCF:一种基于内存的传播式协同过滤推荐算法[J].计算机学报,2013,36(3):671-676.

[9] George T,Merugu S.A scalable collaborative filtering framework based on co-clustering[C]//5th IEEE International Conference on Data Mining.Texas,USA:IEEE,2005,4:10-17.

[10] 杨兴耀,于炯,吐尔根,等.融合奇异性和扩散过程的协同过滤模型[J].软件学报,2013,24(8):1868-1884.

[11] 方耀宁,郭云飞,扈红超,等.一种基于Sigmoid函数的改进协同过滤推荐算法[J].计算机应用研究,2013,30(6):1688-1691.

[12] Chen Zhimin,Jiang Yi,Zhao Yao.A collaborative filtering recommendation algorithm based on user interest change and trust evaluation[J].International Journal of Digital Content Technology and Its Applications,2010,4(9):106-113.

[13] Michael J,Andreas T,Robert L.Combining predictions for accurate recommender systems[C]//Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.New York:ACM Press,2010:693-702.

[14] Zhou T,Ren J,Medo M,et al.Bipartite network projection and personal recommendation[J].Physical Review E,2007,76(4).

[15] 郑翠翠,李林.协同过滤算法中的相似性度量方法研究[J].计算机工程与应用,2014,50(8):147-149.


(上接120页)


[3] Zhang R,Yasuda K,Sumita E.Improved statistical machine translation by multiple Chinese word segmentation[C]//Proceedings of the 3rd Workshop on Statistical Machine Translation.[S.l.]:Association for Computational Linguistics,2008:216-223.

[4] Xu J,Zens R,Ney H.Do we need Chinese word segmentation for statistical machine translation[C]//Proceedings of the 3rd SIGHAN Workshop on Chinese Language Learning,2004:122-128.

[5] 奚宁,李博渊,黄书剑,等.一种适用于机器翻译的汉语分词方法[J].中文信息学报,2012,26(3):54-58.

[6] Wang Y,Uchimoto K,Kazama J,et al.Adapting Chinese word segmentation for machine translation based on short units[C]//LREC 2010:Proceedings of the 7th International Conference on Language Resources and Evaluation,La Valetta,Malta,2010:1758-1764.

[7] Wang Y,Kazama J,Tsuruoka Y,et al.Improving Chinese word segmentation and POS tagging with semi-supervised methods using large auto-analyzed data[C]//Proceedings of 5th International Joint Conference on Natural Language Processing,2011:309-317.

[8] Ma Y,Way A.Bilingually motivated domain-adapted word segmentation for statistical machine translation[C]//Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics.[S.l.]:Association for Computational Linguistics,2009:549-557.

[9] Bai M H,Chen K J,Chang J S.Improving word alignment by adjusting Chinese word segmentation[C]//Proceedings of the 3rd International Joint Conference on Natural Language Processing,2008:249-256.

[10] Dyer C,Muresan S,Resnik P.Generalizing word lattice translation[R].[S.l.]:Inst for Advanced Computer Studies,College Park,Maryland Univ,2008.

[11] Chu C,Nakazawa T,Kurohashi S.Japanese-Chinese phrase alignment using common Chinese characters information[C]//Proceedings of MT Summit,2011,13:475-482.

[12] Chu C,Nakazawa T,Kurohashi S.Chinese characters mapping table of Japanese,traditional Chinese and simplified Chinese[C]//Proceedings of the 8th Conference on International Language Resources and Evaluation (LREC'12),2012.

[13] Goh C L,Asahara M,Matsumoto Y.Chinese word segmentation by classification of characters[J].Computational Linguistics and Chinese Language Processing,2005,10(3):381-396.