

# 一种属性和评分的协同过滤混合推荐算法

李克潮, 蓝冬梅

(广西民族师范学院图书馆, 广西 崇左 532200)

**摘要:**传统协同过滤推荐算法仅仅根据稀疏的评分矩阵向用户推荐,存在推荐质量不高的问题。提出了一种属性和评分的协同过滤混合推荐算法。该算法由项目的类别属性计算项目之间基于属性的相似性,考虑到用户兴趣随时间的变化,构建评分时间权重的指数函数,并应用到项目之间的 Pearson 相关相似性中。通过权重因子加权项目之间基于属性的相似性和项目之间的 Pearson 相关相似性,然后计算基于项目属性的评分预测。描绘职业分类树,构建职业相似性模型,并与性别加权结合产生用户综合属性的相似性,得到基于用户属性的评分预测。最后,综合两者计算混合评分预测。在 Movielens 实验数据集下,实验结果表明提出的算法具有较好的平均绝对误差。

**关键词:**协同过滤;职业分类树;综合相似性;推荐算法

中图分类号:C931.9

文献标识码:A

文章编号:1673-629X(2013)07-0116-04

doi:10.3969/j.issn.1673-629X.2013.07.029

## A Collaborative Filtering Hybrid Recommendation Algorithm for Attribute and Rating

LI Ke-chao, LAN Dong-mei

(Library of Guangxi Normal University for Nationalities, Chongzuo 532200, China)

**Abstract:** Traditional collaborative filtering algorithm exists poor recommendation quality for recommending to the user based solely on sparse rating matrix. Propose a collaborative filtering hybrid recommendation algorithm for attribute and rating. The algorithm computes similarity based on attribute between item by category attributes of item, takes user's interests change over time into account, builds exponential function based on weight of rating time, and applies to Pearson correlation similarity between item. Weighted similarity based on attribute between item and Pearson correlation similarity between item by weighting factor, then calculated rating prediction based on item attribute. Depict professional classification tree, builds profession similarity model, and gets similarity of user's combined attribute by weighted sex, then obtains rating prediction based on user's attribute. At last, compute combined rating prediction by integrated the two above. Under experimental data set of Movielens, experimental results show that the proposed algorithm has a better mean absolute error.

**Key words:** collaborative filtering; professional classification tree; integrated similarity; recommendation algorithm

### 0 引言

协同过滤推荐被应用于亚马逊、当当网、数字图书馆等,向用户推荐电影<sup>[1,2]</sup>、图书<sup>[3]</sup>、科技文献等项目。传统的协同过滤推荐仅仅根据用户对项目的评分记录,计算用户或项目之间的相似性,向目标用户推荐。然而,评分记录非常稀疏,计算用户或项目之间的相似性不准确。

针对以上问题,Yildirim H, Kermarrec A 等<sup>[4,5]</sup>提出随机游走的方法;曹毅等<sup>[6]</sup>提出用向量空间挖掘用户兴趣特征的算法;杜定宇等<sup>[7]</sup>在考虑用户隐私的同时,挖掘用户的兴趣偏好、项目的特征评价,提出基于

中间代理的智能推荐;孙金刚等<sup>[8]</sup>利用云模型对稀疏评分矩阵进行填充,结合项目属性进行推荐;武优西等<sup>[9]</sup>提出改进粒子群属性权重的聚类算法,迭代自组织数据,进行聚类推荐;姜维等<sup>[10]</sup>改进相似度计算方法,提出综合共同评分用户数量、奇异值分解等的协同过滤推荐;Jianhua Liu<sup>[11]</sup>使用 folk sonomy 来填充没有评分的项目数据;Quan Yuan<sup>[12]</sup>等将成员之间、朋友之间的社会关系融合到基于图框架的协同过滤中;Kai Zhou<sup>[13]</sup>提出综合项目评分相似性和项目分类相似性的方法。虽然研究人员提出很多改进的方法,但推荐系统数据稀疏问题依然有待改善。

收稿日期:2012-09-29

修回日期:2012-12-31

网络出版时间:2013-04-08

基金项目:2012年广西教育科研项目(201204LX481);2011年广西民族师范学院科研项目(XYYB2011030)

作者简介:李克潮(1982-),男,硕士,研究方向为个性化推荐系统。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20130408.1715.058.html>

在前人研究的基础上,提出一种属性和评分协同过滤混合推荐算法:

(1)由项目属性矩阵,计算项目之间基于属性的相似性。针对用户兴趣随时间变化的问题,提出基于评分时间加权的项目 Pearson 相关相似性。再根据前面的两种相似性项目综合相似性,计算项目属性的评分预测。

(2)在用户属性中考虑用户职业、性别,产生用户属性的评分预测。

(3)综合前面的两种评分预测,得到加权评分预测。

## 1 属性和评分的协同过滤混合推荐算法

### 1.1 项目属性的评分预测

传统的协同过滤推荐算法,仅依赖 1% ~ 2% 的评分来计算项目之间的相似性,不考虑项目的类别属性对推荐的影响,导致推荐质量较低<sup>[1]</sup>。项目有电影、玩具、服饰等类别属性,而电影又具有动作片、喜剧片、爱情片、武侠片、战争片等属性。这些属性可由如下的矩阵表示:

$$PM = \begin{bmatrix} I_1(1) & I_1(2) & \cdots & I_1(t) & \cdots & I_1(m) \\ I_2(1) & I_2(2) & \cdots & I_2(t) & \cdots & I_2(m) \\ \vdots & \vdots & & \vdots & & \vdots \\ I_n(1) & I_n(2) & \cdots & I_n(t) & \cdots & I_n(m) \end{bmatrix} \quad (1)$$

其中,  $n$  为项目的总个数,  $m$  为项目的属性总个数,  $I_n(t)$  为项目  $I_n$  的第  $t$  个属性。项目之间基于属性的相似性  $\text{sim}_{PM}(I_i, I_j)$  为:

$$\text{sim}_{PM}(I_i, I_j) = \sum_{t=1}^m WE_{I_i, I_j}(t) \quad (2)$$

(2)式中:

$$WE_{I_i, I_j}(t) = \begin{cases} 1 & I_i(t) = I_j(t) \\ 0 & \text{else} \end{cases} \quad (3)$$

传统算法在寻找项目的最近邻居时,也没有考虑到不同用户评分的具体时间对预测未评分项目的影响,忽略用户兴趣随时间的变化<sup>[14]</sup>。

设  $D_{u, I_i}$  为任意目标用户  $u$  对任意目标项目  $I_i$  的评分时间。 $D_{\text{begin}} = \text{Min}(D_{u, I_i})$  表示在所有用户集合评分的项目集合中,用户集合评分第一个项目的时间。 $D_{\text{latest}} = \text{Max}(D_{u, I_i})$  表示在所有用户集合评分的项目集合中,用户集合评分最近一个项目的时间。 $D(u, I_i) = D_{u, I_i} - D_{\text{begin}}$  表示用户  $u$  评分某个项目  $I_i$  的时间与在所有用户集合评分的项目集合中用户  $u$  评分第一个项目

的时间间隔。 $RL$  表示所有用户集合中,最近评分项目的时间与用户集合评分第一个项目的时间之差,即  $RL = D_{\text{latest}} - D_{\text{begin}}$ 。

构建基于用户  $u$  对项目  $I_i$  评分时间权重的指数函数:

$$WD(u, I_i) = e^{-\frac{D(u, I_i)}{RL}} \quad (4)$$

设  $U_{\text{gr}}(u)$  为在最近  $T$  (文中取 10 天) 时间段,已经对项目  $I_i$  和  $I_j$  共同评分过的用户的集合,并且  $U_{\text{gr}}(u) = \{v \mid \text{simps}(u, v) > \delta, u \neq v\}$ 。文中选取  $\delta = 0.5$ ,即只选择那些职业、性别综合属性的相似性大于 0.5 的用户。其中,  $\text{simps}(u, v)$  由下文的(10)式计算。基于评分时间加权的  $I_i$  和  $I_j$  之间的 Pearson 相关相似性  $\text{sim}_D(I_i, I_j)$  如式(5)。

式(5)中,  $w_{u, I_i}$  和  $w_{u, I_j}$  分别为用户  $u$  对项目  $I_i$  和  $I_j$  的平均评分;  $WD(u, I_i) = e^{-\frac{D(u, I_i)}{RL}}$  和  $WD(u, I_j) = e^{-\frac{D(u, I_j)}{RL}}$  分别为用户  $u$  对项目  $I_i$  和  $I_j$  评分时间权重的数据函数;  $\bar{w}_{I_i}$  和  $\bar{w}_{I_j}$  分别为所有用户对项目  $I_i$  和  $I_j$  的平均评分。

根据(2)和(5)式,得到项目的综合相似性:

$$\text{sim}_p(I_i, I_j) = \alpha \times \text{sim}_{PM}(I_i, I_j) + (1 - \alpha) \times \text{sim}_D(I_i, I_j) \quad (6)$$

其中,  $\alpha \in [0, 1]$  为权重因子。

根据  $\text{sim}_p(I_i, I_j)$  大小排序,获得目标项目  $I_i$  的  $p$  个邻居项目集合  $NN_i = \{NN_1, NN_2, \dots, NN_p\}$ ,则用户  $u$  对项目  $I_i$  的基于项目属性的评分预测  $IP_{u, I_i}$  为:

$$IP_{u, I_i} = \bar{w}_{I_i} + \frac{\sum_{I_j \in NN_i} \text{sim}_p(I_i, I_j) * (w_{u, I_j} \times WD(u, I_j) - \bar{w}_{I_j})}{\sum_{I_j \in NN_i} (|\text{sim}_p(I_i, I_j)|)} \quad (7)$$

### 1.2 用户属性的评分预测

不同属性的用户,对项目的喜欢往往不一样。因职业工作的需要,用户购买与职业相关的项目可能性较大。由于性别的差异,女士与男士购买的项目也会有差别。从用户的评分、购买、浏览历史记录,可以看出用户对哪类项目较感兴趣。未来一段时间内,用户很可能再评分、购买、浏览与历史记录相似的项目。

根据智联招聘网提供的职业,可建立图 1 所示的职业分类树。职业可分为机关事业单位、IT、销售等,IT 可分为计算机硬件、计算机软件、互联网等子职业,计算机软件又可进一步细分为软件工程师、软件测试师、系统工程师等子职业。在职业树中,不同子职业的最近

$$\text{sim}_D(I_i, I_j) = \frac{\sum_{u \in U_{\text{gr}}(u)} (w_{u, I_i} \times WD(u, I_i) - \bar{w}_{I_i}) (w_{u, I_j} \times WD(u, I_j) - \bar{w}_{I_j})}{\sqrt{\sum_{u \in U_{\text{gr}}(u)} (w_{u, I_i} \times WD(u, I_i) - \bar{w}_{I_i})^2} \sqrt{\sum_{u \in U_{\text{gr}}(u)} (w_{u, I_j} \times WD(u, I_j) - \bar{w}_{I_j})^2}} \quad (5)$$

父职业所处的层越高,说明这两个子职业的相似性越大。处于职业树最低层的子职业,它们之间的相似性最大。

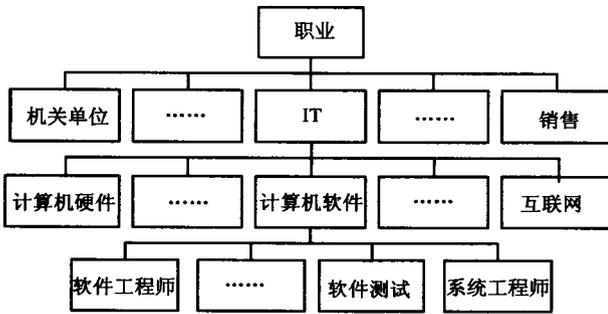


图 1 职业分类树

构建用户  $u$  与  $v$  基于职业属性的相似性  $\text{simp}(u, v)$  :

$$\text{simp}(u, v) = \begin{cases} \frac{\text{Layerp}(u, v) - 1}{\text{Layerp}(\text{all})} & p(u) \neq p(v) \\ 1 & p(u) = p(v) \end{cases} \quad (8)$$

其中,  $\text{Layerp}(u, v)$  为用户  $u$  与  $v$  的职业  $p(u)$  和  $p(v)$  在职业树中最近的共同父节点所在的层,  $\text{Layerp}(\text{all})$  为职业树的总层数。

设用户  $u$  与  $v$  的性别分别为  $s(u)$  和  $s(v)$ , 则  $u$  与  $v$  基于性别属性的相似性  $\text{sims}(u, v)$  为:

$$\text{sims}(u, v) = \begin{cases} 1 & s(u) = s(v) \\ 0 & s(u) \neq s(v) \end{cases} \quad (9)$$

则用户  $u$  与  $v$  基于职业、性别综合属性的相似性  $\text{simps}(u, v)$  为:

$$\text{simps}(u, v) = \beta \times \text{simp}(u, v) + (1 - \beta) \times \text{sims}(u, v) \quad (10)$$

其中,  $\beta \in [0, 1]$  为权重因子。

根据  $\text{simps}(u, v)$  大小排序, 获得目标用户  $u$  的  $p$  个邻居用户集合  $NN_u = \{NN_1, NN_2, \dots, NN_p\}$ , 则用户  $u$  对项目  $I_i$  的基于用户属性的评分预测  $IU_{u, I_i}$  为:

$$IU_{u, I_i} = \bar{w}_{u_i} + \frac{\sum_{I_i \in NN_u} \text{simps}(u, v) * (\bar{w}_{u_i, I_i} \times WD(u, I_i) - \bar{w}_{u_i})}{\sum_{I_i \in NN_u} (|\text{simps}(u, v)|)} \quad (11)$$

其中,  $\text{simps}(r, r')$  由(10)式得到。

### 1.3 属性和评分的混合预测

根据(7)式和(11)式, 计算得到用户  $u$  对项目  $I_i$  的混合评分预测  $P(u, I_i)$  :

$$P(u, I_i) = \chi \times IP_{u, I_i} + (1 - \chi) \times IU_{u, I_i} \quad (12)$$

其中,  $\chi \in [0, 1]$  为权重因子。

### 1.4 推荐过程

为解决常见算法仅仅从评分做推荐, 导致推荐效果不理想的问题, 文章引入项目的属性及用户评分的时间来计算项目的相似性, 引入用户的职业及性别计算用户的相似性。并且, 把评分时间作为基于项目评

分预测的权重, 把目标项目与用户历史项目的相似性作为基于用户评分预测的权重。最后, 根据两者的综合评分预测得到对用户的综合评分预测。

属性和评分的协同过滤混合推荐算法:

输入: 项目的项目属性矩阵  $PM$ , 用户对项目的评分时间, 评分矩阵, 每位用户的职业、性别和项目历史记录集合。

输出: 项目推荐的集合  $\text{top} - N$ 。

过程:

Step1 根据公式(2), 计算项目之间基于属性的相似性  $\text{sim}_{PM}(I_i, I_j)$ 。

Step2 根据公式(5), 计算项目之间基于评分时间加权的 Pearson 相关相似性  $\text{sim}_D(I_i, I_j)$ 。

Step3 根据 Step1 和 Step2 的结果, 计算项目之间的综合相似性  $\text{sim}_P(I_i, I_j)$ 。

Step4 根据图 1 的职业分类树, 由公式(8), 计算用户  $u$  与  $v$  基于职业属性的相似性  $\text{simp}(u, v)$ 。

Step5 根据公式(9), 计算用户  $u$  与  $v$  基于性别属性的相似性  $\text{sims}(u, v)$ 。

Step6 根据公式(10), 计算用户  $u$  与  $v$  基于职业、性别综合属性的相似性  $\text{simps}(u, v)$ 。

Step7 根据公式(12), 对公式(7) 基于项目属性的评分预测  $IP_{u, I_i}$  和公式(11) 基于用户属性的评分预测  $IU_{u, I_i}$  进行加权, 得到用户  $u$  对项目  $I_i$  的混合评分预测。

Step8 由 Step7 的  $P(u, I_i)$  大小对项目进行降序排序, 将最前面的  $N$  个项目集合  $\text{top} - N$  推荐给用户。

## 2 实验结果及分析

### 2.1 实验数据集及度量

文中采用美国 Minnesota 大学 GroupLens 的 MovieLens 数据集 (<http://movielens.umn.edu>)。该数据集包含项目的属性、用户的职业和性别、用户访问项目的时间、943 位用户对 1682 部电影的 100,000 条评分, 并且每位用户至少对 30 部电影进行评分, 评分的值为 1 到 5。将 80% 的数据作为训练集, 余下的 20% 的数据作为测试集。

文中采用平均绝对误差<sup>[1]</sup>, 作为比较常见基于用户(U)的协同过滤算法、基于项目(I)的协同过滤算法与提出的项目属性的评分预测(PF)、用户属性的评分预测(UF)、综合用户和项目的协同过滤推荐算法(UP)推荐质量的标准。

### 2.2 实验结果及分析

实验一 取项目、用户的推荐数和邻居数都为 20, (6)式和(10)式中  $\alpha, \beta$  分别取 0.3、0.4、0.5、0.6、0.7, 基于项目属性(PF)、基于用户属性(UF)的评分

预测平均绝对误差(MAE)对比如图2所示。

从图2可以看出, $\alpha$ 值取过大或过小对算法的效果都不好。当 $\alpha$ 取0.4、 $\beta$ 取0.5时,平均绝对误差取得较好的效果。

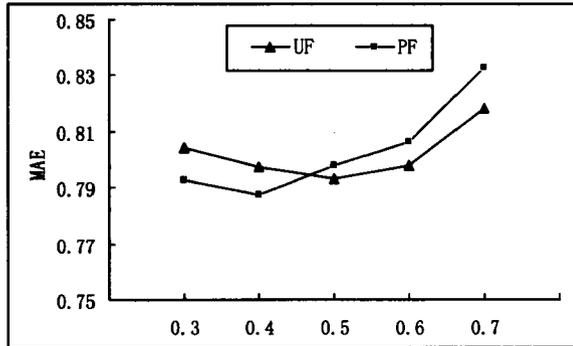


图2 不同 $\alpha$ 、 $\beta$ 值的MAE比较

实验二 项目、用户的推荐数和邻居数都取20,  $\alpha$ 取0.4,  $\beta$ 取0.5, (12)式中 $\chi$ 分别取0.3、0.4、0.5、0.6、0.7, 属性和评分的混合预测推荐算法平均绝对误差(MAE)如图3所示。

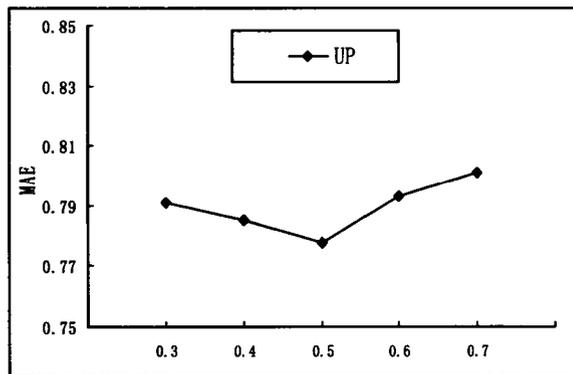


图3 不同 $\chi$ 值的MAE比较

同样,从图3可以看出, $\chi$ 取太大或太小时,算法的推荐效果也不是最理想的。当 $\chi$ 位于0.5时,推荐质量最好。取(12)式中 $\chi=0.5$ ,项目、用户的推荐数和邻居数依次都为10、15、20、25、30,属性和评分的混合预测推荐算法(UP)与常见基于用户(U)、常见基于项目(I)的协同过滤推荐算法在平均绝对误差(MAE)的对比如图4所示。

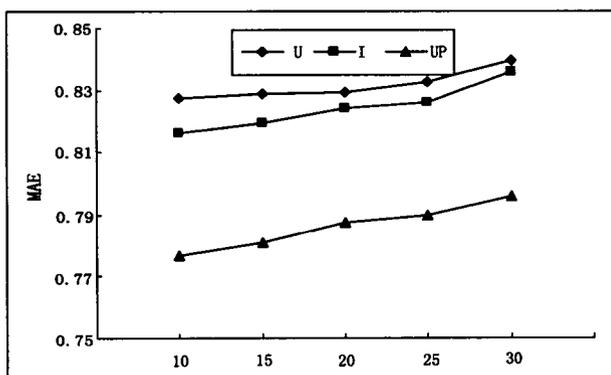


图4 不同推荐数和邻居数的MAE比较

由图4可知,当项目、用户的推荐数和邻居数取上

面的值时,属性和评分的混合推荐效果都比常见的算法有很大的提高。由此可见,文中不仅仅根据用户稀疏的评分矩阵进行推荐,而且还从项目的属性、用户兴趣随时间的变化、用户的职业和性别属性进行推荐,在实验数据集下,得到很好的推荐效果。

### 3 结束语

随着网络的发展,网络资源以指数级的速度在增长,导致用户在信息中出现“迷失”现象。个性化推荐技术能解决用户信息“迷失”、带领大家进入崭新的信息时代,是电子商务等商业的未来<sup>[15]</sup>。因此,个性化推荐技术,将是未来学术界、商业界热门的研究对象。

#### 参考文献:

- [1] Ricci F, Rokach L, Shapira B, et al. Recommender Systems Handbook [M]. New York: Springer Science+Business Media, 2011.
- [2] Jannach D, Zanker M, Felfernig A, et al. Recommender Systems: An Introduction [M]. New York: Cambridge University, 2011.
- [3] Kuroiwa T, Bhalla S. Dynamic personalization for book recommendation system using web services and virtual library enhancements [C]//Proc of the 7th IEEE International Conference on Computer and Information Technology. [s. l.]: [s. n.], 2007.
- [4] Yildirim H, Krishnamoorthy M S. A random walk for alleviating the sparsity problem in collaborative filtering [C]//Proceedings of the 2008 ACM Conference on Recommender Systems. Lausanne: [s. n.], 2008: 131-138.
- [5] Kermarrec A, Leroy V, Moin A, et al. Addressing sparsity in decentralized recommendation systems through random walks [R]. [s. l.]: INRIA, 2010.
- [6] 曹毅, 贺卫红. 基于内容过滤的电子商务推荐系统研究 [J]. 计算机技术与发展, 2009, 19(6): 182-185.
- [7] 杜定宇, 王茜. 一种基于中间代理的个性化推荐系统 [J]. 计算机技术与发展, 2011, 21(9): 66-69.
- [8] 孙金刚, 艾丽蓉. 基于项目属性和云填充的协同过滤推荐算法 [J]. 计算机应用, 2012, 32(3): 658-660.
- [9] 武优西, 侯丹丹, 李建满, 等. 属性权重聚类算法的研究 [J]. 小型微型计算机系统, 2012, 33(3): 651-654.
- [10] 姜维, 庞秀丽. 面向数据稀疏问题的个性化组合推荐研究 [J]. 计算机工程与应用, 2012, 48(12): 1-6.
- [11] Liu Jianhua. Improving the quality of the personalized collaborative filtering recommendation approach employing folksonomy method [M]//Advances in Electronic Commerce, Web Application and Communication Advances in Intelligent and Soft Computing. [s. l.]: Springer, 2012: 597-602.
- [12] Yuan Quan, Chen Li, Zhao Shiwan. Augmenting collaborative recommenders by fusing social relationships: membership and

(下转第123页)

```

If(p->visited == false)
{
If(p 是 temp 的兄弟节点)
{
生成 p 和 temp 的父节点 parent;
把 parent 节点插入到 nodes[i+1]表示的链表中;
把节点 p 和节点 temp 从 nodes[i]中删除;
}
}
p=p->next;
}
Temp->visited=true;
temp=temp->next;
}
}

```

Anycast 地址拆分算法:讲述本算法时举例使用的数据见图 5,本算法为当 Anycast 路由器发现某个 Anycast 地址失效时要采取的动作,即把路由表中对应的路由项删除,当路由表中存在等于这个 Anycast 地址时可以简单地删除此地址即可,如果路由表中存在包含此地址的子网时,需要拆分这个子网,如要删除 G 节点这个时候路由表中存在 B 节点,需要把 B 节点拆分成 C 和 F 节点,然后把 F 节点拆分成 G 和 H 节点,接着把 G 节点删除即可。实现本算法的伪代码如下:

```

node * nodes[32]; //其中包含表示当前路由表的叶节点和网段的数据结构
node * left, * right, * parent; //三个临时的节点指针
node current; //包含要删除的节点
Int i=0; //nodes 数组的索引
for(int j=0; j<32; j++)
{
if((nodes[j]包含 current 节点) || (nodes[j]包含 current 节点的父节点))
{
I=j;
Break;
}
}
* parent = nodes[j];
for(int j=I; j>0; j--)
{
拆分 * parent 为 * left 节点和 * right 节点;
}
}

```

```

if( * left 是 current 节点的父节点)
{
把 * right 插入到 nodes[j-1]链表中;
* parent = * left;
} else
{
把 * left 插入到 nodes[j-1]链表中
* parent = * right;
}
}
把 current 节点从 nodes[0]链表中删除

```

## 5 结束语

文中给出了 IPv6 中新的服务 Anycast,讲解了当前 Anycast 技术实现过程中的一些问题,并对 Anycast 遇到的关键问题给出了相应的解决办法,如对于 Anycast 的规模性问题,文中给出三种方法联合起来解决。文中给出的划分单独的地址空间、单独 Anycast 路由表、地址归并三种方法的联合使用能够大大地解决 Anycast 路由的性能问题,同时又不给 Unicast 和 Multicast 路由的性能带来负面影响。

### 参考文献:

- [1] Partridge C, Mendez T, Milliken W. Host Anycasting Service[S]. RFC 1546, 1993.
- [2] Hinden R M, Deering S E. IP Version6 Addressing Architecture[S]. RFC 2374, 1998.
- [3] 张华健,唐家益. Anycast 通信模型的研究[J]. 计算机应用与软件, 2004, 21(3): 105-107.
- [4] 陈涛,陈启买. 分布式计算机系统负载均衡研究[J]. 计算机技术与发展, 2006, 16(5): 33-35.
- [5] Johnson D. Reserved IPv6 Subnet Anycast Addresses[S]. RFC 2526, 1999.
- [6] Hinden R, Deering N S. IP Version 6 Addressing Architecture[S]. RFC 4291, 2006.
- [7] 伍孝金. IPv6 技术与应用[M]. 北京:清华大学出版社, 2010.
- [8] Stevens W R. TCP/IP Illustrated Volume1: The Protocols[M]. Beijing: China Machine Press, 2002.
- [9] 严蔚敏,吴伟民. 数据结构[M]. 北京:清华大学出版社, 1997.

(上接第 119 页)

- friendship[M]//Recommender Systems for the Social Web Intelligent Systems Reference Library. [s. l.]: Springer, 2012: 159-175.
- [13] Kai Zhou. Combining Item Rating Similarity and Item Classification Similarity for Better Recommendation Quality[C]//Advanced Materials Research. Switzerland: [s. n.], 2012: 289-292.
  - [14] 李克潮,梁正友. 适应用户兴趣变化的指数遗忘协同过滤算法[J]. 计算机工程与应用, 2011, 37(6): 226-243.
  - [15] 苏萌,柏林森,周涛. 个性化:商业的未来[M]. 北京:机械工业出版社, 2012.

## 一种属性和评分的协同过滤混合推荐算法

作者: [李克潮](#), [蓝冬梅](#), [LI Ke-chao](#), [LAN Dong-mei](#)  
作者单位: [广西民族师范学院图书馆, 广西崇左, 532200](#)  
刊名: [计算机技术与发展](#)   
英文刊名: [Computer Technology and Development](#)  
年, 卷(期): 2013, 23(7)

本文链接: [http://d.wanfangdata.com.cn/Periodical\\_wjfz201307029.aspx](http://d.wanfangdata.com.cn/Periodical_wjfz201307029.aspx)