

基于蚁群算法的协同过滤推荐技术

宋伟华, 田元

(西安理工大学图书馆, 陕西 西安 710048)

摘要: 协同过滤推荐技术是当前最成功的个性化推荐技术, 并且已经广泛应用于个性化推荐系统中。考虑到用户的推荐时间、推荐人数都是影响推荐准确度的重要因素, 提出了一种基于蚁群算法的动态协同过滤推荐方法。当系统产生推荐项时, 该算法不仅考虑每项的评分, 而且考虑每项上信息素强度。实验结果表明, 该算法可以显著提高传统过滤推荐系统的推荐质量。

关键词: 协同过滤; 信息素强度; 蚁群算法; 个性化推荐

中图分类号: G255

文献标识码: A

文章编号: 1002-1248 (2013) 08-0053-04

Approach to Collaborative Filtering Recommendation Based on Ant Colony Algorithm

SONG Wei-Hua, TIAN Yan

(Xi'an University of Technology Library, Xi'an 710048, China)

Abstract: Collaborative Filtering Recommendation Algorithm (CF) is a well-known method which is applied most widely in the personalized recommendation system. Considering that recommendation time, recommendation times and so on the important factors to influence the accuracy of recommendation system, this paper proposed a dynamic CF based on ant colony algorithm. When getting the recommendation, this algorithm is not only taken account of the each rate but also intensity of pheromone on each item. The experimental results show that this method can efficiently improve the quality of the commendation than traditional collaborative filtering algorithms.

Keywords: Collaborative filtering; intensity of pheromone; Ant colony algorithm; personalized recommendation

随着互联网的迅速发展, 网上信息呈爆炸式的增长。如何为网络用户高效的提供所需信息, 已经成为数据挖掘中的热点问题。目前, 国内外学者提出了很多相关算法。协同过滤 (Collaborative Filtering) 就是在这种需求下提出的一种信息服务的新算法, 目前该算法是使用最成功的技术之一, 已经广泛应用于各大商业站点中的个性化推荐系统当中。Tpestry^[1]是最早提出来的基于协同过滤思想的推荐系统^[2], 该算法的基本思想就是: 根据当前用户的行为或者偏好和其它用户进行比较, 找出与该用户有相似或相同偏好的其它用户, 根据相似用户的评分来预测当前用户对商品的评分, 将预测评分最高的前 N 个商品推荐给当前用户。

实际生活中, 用户对商品的兴趣度是不断变化的, 用户可能开始对某个类商品感兴趣, 但随着时间的变化, 有可能对该类商品失去兴趣, 转而对新的某类商品感兴趣, 这就需要推荐系统不断的跟踪用户群的兴

趣度, 始终给目标用户推荐其感兴趣的商品。然而, 传统的协同过滤系统无法及时跟踪用户群的兴趣度变化, 往往给目标用户推荐出过时的商品信息, 这种过时的商品信息并不是当前用户真正感兴趣的信息, 而是一些无用信息, 这样就大幅降低了推荐系统的推荐质量。因此, 完全有必要对传统协同过滤方法进行改进, 使系统推荐的商品信息能够根据用户的变化而不断更新, 推荐出目标用户真正感兴趣的商品信息。

受蚁群算法的启发, 在用户给商品评分的同时, 留下信息素, 给目标用户做出推荐时不仅考虑预测评分高低, 还要依据信息素的强度做出推荐。这样就给过滤方法增加了信息正反馈, 从而不断调整推荐结果, 帮助用户发现最新出现的感兴趣的商品信息。现分别阐述关于蚁群算法的协同过滤算法基本原理、蚁群算法、基于蚁群算法的协同过滤推荐算法及实验结果。

收稿日期: 2013-04-19

作者简介: 宋伟华 (1980-), 女, 西安理工大学图书馆, 助理馆员。田元 ((1975-), 男, 西安理工大学图书馆, 馆员。

1 协同过滤算法的基本原理

协同过滤算法有 3 个主要步骤^[3]: 构造用户—项目矩阵、查找目标用户的最近邻居、生成推荐项目。

1.1 构造用户—项目矩阵

根据用户对商品的评价或者评分, 建立 $m \times n$ 阶用户—项目矩阵 $A(m, n)$ 。该矩阵如表 1 所示。

表 1 用户—项目矩阵

	Item1	...	Itemk	...	Itemn
User1	R11	...	R1k	...	R1n
...
Userj	Rj1	...	Rjk	...	Rjn
...
Userm	Rm1	...	Rmk	...	Rmn

其中 m 表示 m 个用户, n 表示 n 个商品项目, R_{ij} 表示第 i 个用户对商品 j 的兴趣度或者评分。

1.2 查找目标用户的最近邻居

协同过滤算法的最近邻居就是在用户—项目矩阵中与目标用户评分、兴趣度相似的若干用户。该方法通过得到目标用户和用户 i 评分过的所有项, 然后通过某种相似性度量计算目标用户和用户 i 之间的相似性。然后选取相似度高的 l 个用户作为最近邻居集。目前, 在协同过滤算法中计算相似度 $\text{sim}(i, j)$ 主要使用余弦相似性 (式 1)、相关相似性 (式 2) 以及修正的余弦相似性三种方法 (式 3)^[4]。

$$\text{sim}(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\| \|\vec{j}\|} \quad (\text{式 1})$$

其中 \vec{i} 和 \vec{j} 向量分别表示用户 i 和用户 j 在 n 维项目空间上的评分

$$\text{sim}(i, j) = \frac{\sum_{c \in I_{ij}} (R_{i,c} - \bar{R}_i)(R_{j,c} - \bar{R}_j)}{\sqrt{\sum_{c \in I_{ij}} (R_{i,c} - \bar{R}_i)^2} \sqrt{\sum_{c \in I_{ij}} (R_{j,c} - \bar{R}_j)^2}} \quad (\text{式 2})$$

其中 I_i 和 I_j 分别表示用户 i 和用户 j 评分的项目集, I_{ij} 表示用户 i 和用户 j 共同评分的商品项目集合。 $R_{i,c}$ 表示用户 i 对商品 C 的评价, \bar{R}_i 和 \bar{R}_j 分别表示用户 i 和用户 j 对项目评分的平均值。

$$\text{sim}(i, j) = \frac{\sum_{c \in I_{ij}} (R_{i,c} - \bar{R}_i)(R_{j,c} - \bar{R}_j)}{\sqrt{\sum_{c \in I_{ij}} (R_{i,c} - \bar{R}_i)^2} \sqrt{\sum_{c \in I_{ij}} (R_{j,c} - \bar{R}_j)^2}} \quad (\text{式 3})$$

(式 3) 中的参数于 (式 2) 中的含义一样。

1.3 生成推荐项目

该步骤就是根据上一步得到的用户近似度值, 得到目标用户 P_u 的最近邻居集合 NN_u , 然后通过最近邻

居集合 NN_u 并利用 (式 4) 预测目标用户 P_u 对项目 t 的评分 $P_{u,t}$ 。

$$P_{u,t} = \bar{R}_u + \frac{\sum_{i \in NN_u} \text{sim}(u, i) * (R_{i,t} - \bar{R}_i)}{\sum_{i \in NN_u} |\text{sim}(u, i)|} \quad (\text{式 4})$$

其中 $\text{sim}(u, i)$ 表示为目标用户 U 和最近邻居的相似度, $R_{i,t}$ 表示最近邻居用户 i 对项目 t 的评分。 \bar{R}_u 和 \bar{R}_i 分别表示目标用户 U 和最近邻居用户 i 对项目的平均兴趣度。通过该方法预测目标用户对所有未评分项目进行评分, 最后根据计算出来的评分预测值, 选择预测评分值高的前 n 项作为推荐结果推荐给目标用户。

2 蚁群算法

自然生物中, 蚂蚁的个体智能并不高, 看起来也没有集中的指挥, 但是它们却能够协同工作, 依靠群体的能力发挥超出个体百倍的智能。蚁群算法是由意大利学者 Dorigo M 等^[5]最先提出的, 是模拟蚂蚁群体智能行为的一种仿生算法。由于该算法有较强的鲁棒性、优良的分布式计算、易于与其它算法结合等优点, 已经渗透到多个应用领域。

蚁群算法的原理就是, 蚂蚁没有视觉, 在路径上运动时, 通过其它蚂蚁释放的信息素来选择路径, 蚂蚁走的路径越长, 释放的信息素也越小。当后来的蚂蚁选择路径时, 选择路径上信息量大的概率较大, 这样形成了一个正反馈机制。最优路径上的信息量越来越大, 而其它路径上的信息量随着时间的流逝而逐步减少。同时蚁群还能够适应环境的变化, 根据环境的影响, 发现新的最优路径。可见, 在整个寻优过程中, 虽然单只蚂蚁的选择能力有限, 但是通过信息素的作用使整个蚁群行为具有非常高的自我组织性, 蚂蚁之间交换着路径信息, 最终通过蚁群的集体催化行为找出最优路径。

3 基于蚁群算法的协同过滤推荐算法

通过上面的介绍可以看出, 传统的协同过滤推荐算法通过用户兴趣的大小预测目标用户对项目的喜欢程度, 但是没有考虑到用户兴趣变化的问题, 从而降低了系统推荐结果的准确度。例如: 早期在学习计算机操作系统时, 用户群推荐了不少关于 DOS 操作系统的经典图书, 但随着时代发展, 用户希望协同过滤推荐系统推荐关于 Windows Vista, Windows7 这方面的图书, 而对关于 DOS 方面的图书失去兴趣, 按照传统的协同过滤推荐技术算法推荐出的这类图书, 不仅不会引起读者的兴趣, 反而引起用户的反感。由此可以看

出,随着时间的不断的推移,不能根据用户群已经失去兴趣的商品作为推荐的依据,而应提出某种算法跟踪用户群体兴趣的变化,根据兴趣的变化不断更新系统的推荐结果。

3.1 相关工作

在详细介绍该算法之前,先介绍与该算法有关的一些算法。假设在 t 时刻某个项目的信息素表示为 $\tau_i(t)$, 则 $\Gamma = \{\tau_i(t) | c_i, c_j \in C\}$ 表示为 t 时刻项目集合 C 中项目元素残留信息量的集合。初始时刻各个项目元素上的信息量相等,并设 $\tau_i(t) = \text{const}$ 。

在蚁群算法中为了避免残留信息过多而引起残留信息淹没启发信息,每只蚂蚁走完一步或者完成对所有 n 个城市的遍历后,都要对残留信息进行更新处理。本算法借用该步骤来模拟人类兴趣度的特点,新的商品项不断被用户推荐,新商品的信息素不断的增加,而用户对一些旧的商品项的兴趣度随着知识的不断更新慢慢被淡化,这些商品项的信息素也慢慢减弱,甚至淘汰。因此, $t+k$ 时刻某个商品的信息素按公式(式5)进行调整。

$$\tau_i(t+n) = (1-\rho) * \tau_i(t) + \Delta\tau_i(t) \quad (\text{式 } 5)$$

其中式(5)中, ρ 表示信息素的挥发程度。 $1-\rho$ 表示信息的残留因子,为了避免信息素不断增加 $\rho \in [0,1]$ 。 $\Delta\tau_i(t)$ 表示有用户推荐该图书时的信息增量,初始状态下, $\Delta\tau_i(t) = 0$ 。蚁群算法中,有多种不同的信息素更新策略,此推荐算法中借用 Ant-Density 模型的更新策略,并对其进行修改,即:

$$\Delta\tau_i(t) = \begin{cases} Q, & \text{若有用户推荐 } t \text{ 项商品} \\ -0.5 * Q, & \text{否则} \end{cases}$$

在推荐图书时,系统根据商品项上的信息素和预测评分的高低来计算每个商品的推荐概率,即 $p(t)$ 。

$$p(t) = \frac{[\tau_i]^a [\gamma_i]^\beta}{\sum_{c \in n} [\tau_c]^a [\gamma_c]^\beta} \quad (\text{式 } 6)$$

在(式6)中 r_i 为目标用户对该商品的评分或预测评分。 α 为信息启发因子,表示信息的重要性,反映了系统中一个商品的关注度,当用户对该商品关注度越高,其值越大,该书就有可能被系统推荐给目标用户。 β 为期望启发式因子,表示评分的重要性。根据(式6),如果某个商品的评分高且用户对其评价的次数越多,那么,该商品被推荐的概率越大。反之,某个商品的评分低且用户对其评价的次数越少,则该商品被推荐的概率就小。

3.2 算法实现

下面是基于蚁群算法的协同过滤推荐算法的具体实现:

Step1:初始化,读取用户-项目矩阵并设置 α, β 等参数

Step2:目标用户对感兴趣的商品进行评价,用户每评价一个商品,利用(式5)更新该商品的信息素 $\tau_i(t)$

Step3:用相似度 $\text{sim}(i,j)$ 函数,计算目标用户 Tu 和其它用户的相似度。

Step4:将与目标用户 Tu 相似度最高的 K 个用户存入其最邻近邻居集合 TU 。

Step5:利用(式4)和最邻近邻居集合 TU 预测目标用户对商品项 t 的评分。

Step6:利用(式6)计算每个商品项目的推荐概率。

Step7:根据每个商品的推荐概率,选择 n 个商品作为推荐项目

4 实验

4.1 数据集

为了验证基于蚁群算法的协同过滤推荐算法的有效性,现选择西安理工大学图书馆2005年—2010年图书借阅数据。

首先将2005借阅数据分为80%和20%两部分,其中80%作为训练数据,而另外20%的数据和以后的借阅数据作为测试数据。

在读取训练数据集时,每读取一个训练数据集,都要根据算法更新相应商品项的信息素,在读取测试数据集进行测试时,也要根据目标用户的评价,更新商品项的信息素。

实验中所采取的各项参数如下: $\alpha=3, \beta=5, \rho=0.1$, 系统信息素强度的初始值为0.1。

4.2 评价标准

评价推荐系统推荐质量的度量标准主要包括:统计精度度量方法和决策支持精度度量方法两类^[6]。统计精度度量方法中的平均绝对偏差(Mean Absolute Error, MAE),通过计算预测的评分于实际的评分之间的偏差来衡量预测的准确性,MAE越小,预测越准确。由于该度量方法具有易于理解、相对直观等优点,已被广泛应用于推荐系统的测试当中,故在此也采用平均绝对偏差(MAE)作为系统的评价标准。MAE的定义为^[6]:

$$\text{MAE} = \frac{\sum_{i=1}^N |p_i - q_i|}{N} \quad (\text{式 } 7)$$

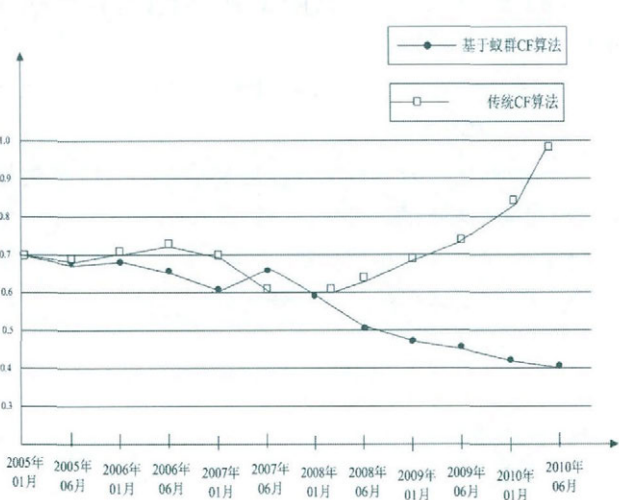
4.3 实验结果

在进行测试的时,测试数据的读取按照时间的顺

表 2 传统 CF 算法和基于蚁群算法 CF 算法的平均 MAE

	2005 年 1 月	2005 年 6 月	2006 年 1 月	2006 年 6 月	2007 年 1 月	2007 年 6 月	2008 年 1 月	2008 年 6 月	2009 年 1 月	2009 年 6 月	2010 年 1 月	2010 年 6 月
AntCF	0.701	0.68	0.679	0.668	0.6	0.665	0.59	0.5	0.473	0.451	0.423	0.402
CF	0.702	0.69	0.71	0.719	0.7	0.6	0.6	0.65	0.68	0.74	0.83	0.99

序进行读入，实验结果见附图。



附图 传统 CF 算法和基于蚁群算法 CF 算法的不同 MAE

从附图可以看出，随着时间不断的推移，传统的 CF（这里记为 CF）的 MAE 值总体上呈上升趋势，而基于蚁群算法的协同过滤推荐算法（这里记为 AntCF）的 MAE 值基本上是下降趋势。表 2 为本次实验的具体数据，表 3 为两种算法的平均 MAE 值。

表 3 传统 CF 算法和基于蚁群算法 CF 算法的平均 MAE

	平均 MAE 值
CF	0.7176
AntCF 算法	0.5693

5 结论

由前面的实验结果可得到如下结论：总体上基于蚁群的协同推荐算法在准确性方面优于传统的协同过滤算法。由于用户的兴趣随着时间逐渐在改变，因此推荐系统中通过信息素的思想而跟踪用户的兴趣的变化，将有助于向用户提供真正感兴趣的商品。

参考文献：

[1] Goldberg D, Nichols D, Oki B M. Using Collaborative Filtering to Weave an Information Tapestry [J]. Communications of the ACM, 1992,35(12): 61-70.

[2] 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,(9)

[3] GOLDBERGD,NICHOLSD,OKIBM, et al. Using collaborative filtering to weave an information tapestry [J]. Communications of the ACM, 1992, 35 (12): 61-70.

[4] Sarwar B, Karypis G, Konstan J, Riedl J. Item-Based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International World Wide Web Conference.2001.285-295.

[5] Colomi A, Dorigo M, Maniezzo V. Distributed optimization by ant colonies. In: Varela F, Bourguine P, eds. Proc. of the ECAL'91 European Conf. of Artificial Life. Paris: Elsevier,1991.134-144.

[6] Sarwar B, Karypis G, Konstan J, Riedl J. Item-Based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International World Wide Web Conference. 2001.285-295.