一种个性化协同过滤混合推荐算法

蒋宗礼，汪瑜彬

（北京工业大学 计算机学院，北京 100124）

摘 要：传统协同过滤算法主要根据稀疏的评分矩阵向用户作出推荐，存在推荐质量较差的问题。为此，提出一种基于信息熵的综合项目相似度度量方法。考虑到用户的兴趣会随时间发生变化，而且不同用户群体的兴趣变化不同，受艾宾浩斯记忆遗忘规律启发，提出适应于不同用户群体兴趣变化的数据权重。基于 movielens 数据集的实验结果表明，改进后算法不仅能缓解评分数据稀疏问题，而且能提高算法的准确率。

关键词：推荐系统；协同过滤；项目属性；信息熵

DOI：10.11907/rjdk.1511423

中图分类号：TP312 **文献标识码**：A **文章编号**：1672-7800(2016)003-0052-05

0 引言

随着互联网飞速发展，特别是 Web2.0 技术逐步走向成熟，推荐系统在电子商务中的作用越来越大，一方面帮助用户找到和发现自己感兴趣的物品；另一方面，能够将一些冷门的商品呈现在感兴趣的用户面前，帮助商家挖掘潜在用户，体现其长尾效应[1]，实现消费者和商家共赢，使得推荐系统得到越来越多学者与专家的关注。

1 相关工作

推荐系统的核心是推荐算法，目前，基于项目的协同过滤算法[2]（以下简称 Item_CF）应用最为广泛。许多大型网站，如 Amazon，Netflix，Hulu，YouTuBe 所用推荐系统均以该算法为基础[1]。Item_CF 算法描述如下：

算法 1：Item_CF 算法

输入：用户 u、与之对应的已访问资源集 I_u

输出：用户 u 的 top－N 推荐

步骤 1：由历史评分数值出发计算项目之间的相似度，并按照相似度的大小从高到低排序构造近邻模型 M。

步骤 2：对于每一个项目 $i \in I_u$，从 M 中读取其 k 个最近邻集合 $Neb_i = \{i_1, i_2, \cdots i_k\}$，合并所有最近邻集合 Neb_i 得到集合 C。

步骤 3：删除候选集 C 中用户已经访问过的项目，得到推荐候选集合 C'。

步骤 4：对候选集 C' 中的每一个项目，分别计算该项目对用户 u 的加权推荐度。

步骤 5：对候选集中的项目按推荐度的大小从高到低进行排序，选取排名靠前的 N 个项目推荐给用户 u。

Item_CF 虽然在性能上不错，但是存在数据稀疏[4]和无法捕捉用户兴趣变化问题。

所谓数据稀疏是指随着用户和项目的不断增长，用户/项目评分矩阵变得稀疏，导致计算项目相似度时不够准确[1]。为了解决这一问题，一些改进方法采用了预填充项目评分矩阵思想。文献[5]提出了一种基于项目和基于用户相结合的预填充模型算法；文献[6]则介绍了一种基于概率知识的评分填充算法。基于模型的降维技术是解决数据稀疏问题的另外一种有效途径。文献[7]提出采用 SVD 方法对评分矩阵进行降维，再使用降维后的稠密评分矩阵进行推荐；文献[8]指出可以通过主成分分析（PCA）的降维方法来缓解数据稀疏问题。这些改进使得 Item－CF 算法在性能上有所提高，但是都主要是依赖于用户对项目的评分，而没有考虑项目属性，依靠单一的数据源，特别是在数据极为稀疏情况下，对推荐算法性能的提高是有限的。

传统的 Item_CF 算法在实现推荐过程中，历史评分值是实现推荐的主要依据，无论历史评分的产生时间存在何种差别，其在推荐中所起的作用是相等的[3]，忽视了用户兴趣会随时间发生变化，从而导致算法准确度提高受

基金项目：北京市重点学科基金项目（007000541215042）

作者简介：蒋宗礼（1956－），男，河南南阳人，北京工业大学计算机学院教授，研究方向为网络信息搜索与处理；汪瑜彬（1990－），男，安徽黄山人，北京工业大学计算机学院硕士研究生，研究方向为网络信息搜索与处理。

限。为此，文献[9]～[11]借鉴心理学遗忘理论，在算法中考虑用户访问项目的时间，以刻画用户的兴趣变化。文献[9]提出了基于用户兴趣变化的时间权重与资源权重，他们认为人的兴趣是随时间线性衰减的。文献[10]、[11]进一步提出了与人遗忘规律较为接近的艾宾浩斯遗忘曲线。实验结果表明这些改进提高了 Item_CF 算法的推荐质量。但是，他们在改进中为所有用户设置了相同的兴趣衰减速率，相当于认为所有用户群体的兴趣变化是一样的，这与现实存在差异，因而未能更加准确地反映用户的兴趣变化。

针对以上两点，本文提出对 Item－CF 算法进行改进：①在步骤 1 构造近邻模型 M，计算项目相似度时，考虑项目属性相似度的计算，得到结合项目属性和项目评分的综合相似度；②在步骤 4 计算候选集中项目对目标用户的推荐度时，考虑用户兴趣的变化，添加适应于不同用户群体兴趣变化的数据权重。

2　基于信息熵的综合相似度

在协同过滤系统中，近邻模型构造是否准确，直接影响推荐效果。而近邻模型的构造需要通过计算项目相似度得到。传统 Item_CF 算法在计算项目相似度时主要依靠用户对项目的评分，当评分数据量比较稀疏时，相似度计算不够准确，最终将导致推荐算法的精度下降。为此，本文提出一种融合项目属性的综合相似度计算方法，以解决评分数据稀疏问题。

2.1　计算项目评分相似度

考虑到不同用户评价标准存在着差异[13－15]，本研究采用修正后的余弦相似度度量项目评分相似度，公式如下：

$$sim_{\cos}(i,j)=\frac{\sum_{u\in U}(R_{i,u}-\bar{R}_u)*(R_{j,u}-\bar{R}_u)}{\sqrt{\sum_{u\in U}(R_{i,u}-\bar{R}_u)^2}*\sqrt{\sum_{u\in U}(R_{j,u}-\bar{R}_u)^2}} \tag{1}$$

2.2　计算项目属性相似度

假设需要计算的项目共有 n 个：I_1 、I_2 … I_n 。项目当中共有 s 种属性：a_1 、a_2 … a_s 。可以得到如下项目属性表。

表 1　项目属性

项目\属性	a_1	a_2	…	a_s
I_1	1	0	…	1
I_2	0	1	…	1
…	…	…	…	…
I_i	1	1	…	1
…	…	…	…	
I_i	0	1	…	1
…	…	…	…	…
I_n	0	1	…	1

计算属性相似度最简单的方法是 Jaccard 系数度量[12]法，公式如下：

$$sim_J(i,j)=\frac{|A(i)\cap A(j)|}{|A(i)\cup A(j)|} \tag{2}$$

其中，|A(i)UA(j)|表示项目 i 与项目 j 所具有的属性总数，|A(i)∩A(j)|表示项目 i 与项目 j 同时具有的属性个数。但是，这种方法并没有考虑到项目当中不同属性的影响力是不一样的，如系统中大多数电影都具有喜剧这个属性，那么喜剧属性对于项目相似性的区分度就比较低，从而影响力就比较小。为了考虑不同属性间影响力的差异，提出以属性信息熵为加权，构造加权的 Jaccard 系数相似度度量方法。

信息是抽象概念，人们常常说信息量很大，或者信息较少，但却很难说清楚信息到底有多少。直到 1948 年，香农借助热力学理论，提出了“信息熵”的概念，才解决了对信息的量化度量问题。信息熵计算公式如下：

$$H(X)=\sum_{i=1}^{n}-p_i\times log_2(p_i) \tag{3}$$

其中，X 表示某随机变量，n 表示随机变量 X 的 n 种不同取值，p_i 表示随机变量 X 第 i 种取值出现的概率。利用上述信息熵公式可以度量某种信息不确定性的大小。

经典的 Jaccard 系数度量方式没有考虑不同属性之间对项目区分的贡献度是有差异的，而这种差异体现了属性间不确定性的大小，因此可以用属性的信息熵来描述每个属性对项目的区分度，形成加权 Jaccard 系数相似度。

对于一个属性 a_m ，由表 1 可知其取值只能是 1 或者 0，表示某个项目是否有该属性。若假定属性 a_m 在系统所有项目中出现的概率为：

$$p(a_m)=\frac{k}{|I|} \tag{4}$$

其中，k 表示属性 a_m 出现的次数，|I|表示项目的总数，则根据式(3)可知，属性 a_m 的信息熵如下：

$$H(a_m)=-p(a_m)\times log_2(p(a_m))-(1-p(a_m))\times log_2(1-p(a_m)) \tag{5}$$

信息熵大的属性对项目有更好的区分度，应赋予更大的权重。从而可以构造加权的 Jaccard 系数相似度：

$$sim_H(i,j)=\frac{\sum_{a_n\in A(i)\cap A(j)}H(a_n)}{\sum_{a_m\in A(i)UA(j)}H(a_m)} \tag{6}$$

式(6)中，A(i) 表示某个项目 i 所具有的属性集合，A(i)∩A(j)表示项目 i 与项目 j 属性集合的交集，A(i)UA(j)表示项目 i 与项目 j 属性集合的并集。

2.3　融合评分相似度和属性相似度

融合上述两种不同方式计算出来的相似度，通常所用方法是取两者的线性组合[12,15,16]。公式如下：

$$sim(i,j)=\alpha\times sim_{\cos}(i,j)+(1-\alpha)\times sim_H(i,j) \tag{7}$$

2.4　近邻模型 M 生成算法

上述综合相似度，可以生成近邻模型 M。

算法 2：近邻模型 M 的生成算法。

输入：用户—项目评分矩阵 $R[u,I]_{n\times m}$，项目—属性矩阵 $A[I,a]_{m\times s}$，权重因子 α

输出：项目近邻模型 M

步骤 1：根据式(1)，由用户—项目评分矩阵 $R[u,I]_{n\times m}$ 求出项目与项目之间评分相似性矩阵 $sim_{\cos}(i,j)]_{m\times m}$；

步骤 2：根据公式(5)、(6)，由项目—属性矩阵计算项目与项目之间属性相似度矩阵 $sim_H(i,j)]_{m\times m}$；

步骤 3：采用公式(7)计算最终项目与项目之间相似度矩阵 $sim(i,j)]_{m\times m}$；

步骤 4：对 $sim(i,j)]_{m\times m}$ 每行按照相似度的大小从高到底排列。

步骤 5：输出项目近邻模型 M。

3 适应不同用户群体兴趣变化的权重

传统的 Item_CF 在计算候选集中项目对目标用户的推荐度时，存在一个问题：它等同地对待目标用户在不同时间段内访问的项目，忽略了用户兴趣会随时间发生变化，从而导致推荐系统推荐的项目在很大程度上偏离了用户的需求。

为了解决上述问题，本文提出两种权重：基于指数衰减的时间权重和基于项目相似度的数据权重。并采用乘积的方式将两者融合在一起，得到一种适应不同用户群体兴趣变化的数据权重，以解决传统的 Item_CF 不能及时反映用户兴趣变化这一弊端。

3.1 基于指数衰减的时间权重

德国心理学家艾宾浩斯认为人的遗忘规律是：开始阶段遗忘的速度很快，到后来遗忘速度会越来越缓慢。这一观点被大量心理学实验所证实。基于此，文献[12]将该观点引入到推荐算法中用于表征用户兴趣的衰减，并给出如下时间权重度量公式：

$$WD(u,i)=e^{-a\times\frac{Date_{interval}(i)}{L_u}} \tag{8}$$

其中，L_u 表示用户 u 使用推荐系统的时间跨度，$Date_{interval}$ 表示用户对项目 i 的评分时间与用户对最近一个项目的评分时间的间隔。a∈(0,1)，可以通过实验找到一个最佳值。通过引入适应用户兴趣变化的指数衰减权重，提高算法的准确率。但是，这一改进并没有考虑不同用户群体随时间变化是不一样的，通常情况下，青少年的兴趣变化波动较大，而成年人的兴趣变化相对比较稳定。因此，可以将不同的用户群体根据年龄划分为不同的年龄段，为年龄大的用户群体设置较为缓慢的兴趣衰减速率。经过以上分析，可以对式(8)进行如下改进：

$$WD\_A(u,i)=e^{-\frac{1}{1+\beta\times ageGroup}\times\frac{Date_{interval}(i)}{L_u}} \tag{9}$$

ageGroup 表示目标用户所处的年龄阶段数。显然 ageGroup∈[1,ranges]，其中 ranges 表示推荐系统将所有用户依照年龄所划分的年龄阶段个数。本文实验所用 Movielens 数据集已将年龄层次划分为如下几个阶段：18岁以下，18～24岁，25～34岁，35～44岁，45～49岁，50～55岁，56岁以后。因此 ranges=7。$\beta>0$，是一个随时间变化的衰减因子，β 越大，说明用户兴趣随时间衰减越缓慢，可以根据不同数据集通过实验设定一个合适的值。

3.2 基于项目相似度的数据权重

式(9)在于根据不同用户群体设置不同速率的指数衰减曲线，以获得削弱旧行为数据，突出新行为数据的效果。但现实中，用户兴趣存在一定的反复性，早期行为有时对于生成推荐同样很重要，单纯采用上述时间权重会削弱所有早期数据，对算法准确度提高不利，为此引入第二种权重：基于项目相似度的数据权重。

设用户 u 已访问的项目集合为 I_u，用户最近 T 时间段内访问的项目集合为 I_{uT}，显然，I_{uT} 代表了最近 T 时间段的兴趣，对于任意项目 i∈ I_u，如果项目 i 与 I_{uT} 中很多项目相似度很高，则说明项目 i 与用户最近感兴趣的项目比较相关，所以可以认为用户 u 在未来一段时间内很可能对与项目 i 相似的其它项目感兴趣，也就是说，项目 i 对生成最终的推荐同样起着比较重要的作用。因此，定义基于项目相似度的数据权重 WS(u,i)以反映上述项目 i 和用户近期兴趣的关系，公式如下：

$$WS(u,i)=\overline{sim(u,i)}=\frac{\sum_{j\in I_{uT}}sim(i,j)}{size(I_{uT})} \tag{10}$$

其中，$size(I_{uT})$ 表示用户 u 最近 T 时间段内访问的项目个数。T 的大小可以根据不同的数据集设置不同的值。

3.3 权重融合

基于指数衰减的时间权重能够为不同用户群体设置不同的兴趣衰减速率，目的是突出用户当前兴趣的重要性，解决不同用户群体兴趣变化比较频繁的问题；而基于项目相似度的数据权重能够避免有价值的早期兴趣被忽略，解决用户兴趣的反复性问题。两种权重各有千秋，为了更好地和文献[10]对比，本文仍采用乘积的方式将两者融合在一起，得到一种适应于不同用户群体兴趣变化的数据权重：

$$WTS(u,i)=WD\_A(u,i)*WS(u,i) \tag{11}$$

通过上述改进后，候选集中项目 j 对目标用户 u 的推荐度计算方式如下：

$$rec\_w(u,j)=\sum_{i\in I_u\&\&j\in Nebli,k)}WTS(u,i)\times r(u,i)\times sim(i,j)\times sim(i,j) \tag{12}$$

式(12)中，$r(u,i)$ 表示用户 u 对项目 i 的评分，$i\in I_u\&\&j\in Neb(i,k)$ 表示对于要进行累加权值的项目 i 不仅要为用户 u 历史访问过的项目，而且其 k 个最近邻里面含有项目 j。

4　改进 Item_CF 算法描述

算法 3：改进的 Item_CF 算法

输入：用户 u，包含用户 u 访问的项目及时间的集合为 I_u、用户所处年龄段 ageGroup、时间窗大小 T、参数 β 值、最近邻个数 k

输出：用户 u 的 top－N 推荐

步骤 1：按照算法 2 构造近邻模型 M；

步骤 2：对于每一个项目 $i \in I_u$ 读取它的 k 个最近邻集合，$Neb_i = \{i_1, i_2, \cdots I_k\}$ 合并所有最近邻集合 Neb_i 得到集合 C；

步骤 3：删除候选集 C 中用户已经访问过的项目，得到候选推荐集合 $C' = C - I_u$；

步骤 4：对于每一个项目 $i \in I_u$，根据式(11)计算采用乘积方式融合的综合权重 $WTS(u,i)$。得到用户 u 的综合考虑时间因素的权重向量 VT；

步骤 5：对于推荐候选集中的每一个项目 $j \in C'$，结合上述权重向量 VT 和公式(12)，计算最终的加权推荐度 $rec\_w(u,j)$；

步骤 6：按加权推荐度大小排列 C' 中的项目，将排名最为靠前的 N 个项目推荐给用户 u。

5　实验与结果分析

本文实验数据来源于 movielens 网站提供的 1M 数据集，它收录了 6 040 个用户对 3 952 部电影的约 100 万次评分记录，每个用户至少有 20 次评分记录。数据集主要有 3 个文件，其中 rating. dat 包含用户 ID、电影 ID、评分分值(分值为 1～5)以及评分时间等信息；users. dat 内含有用户所处的年龄段等信息；movies. dat 含有电影类型等信息，这些都是实现本文算法所必需的信息。

5.1　评价指标

本文所使用的评价指标是推荐准确率(precision)，该指标是评价推荐系统优劣的重要指标。将整个数据集随机划分为训练集与测试集(按照常规 4∶1 划分)，根据训练集中的数据为每个用户作 Top－N 推荐，若系统为用户预测的项目落入该用户所在的测试集当中，则称之为一次命中，统计命中的次数并用它除以推荐项目总数得到准确率。公式如下：

$$precision = \frac{Hits}{N} \tag{13}$$

5.2　实验策略与结果分析

在实验描述当中，将本文提出的推荐算法命名为 HAttrTime_CF；采用本文方法将添加项目属性的算法命名为 HAttr_CF；文献[10]提出的适应于用户兴趣变化的指数衰减推荐算法命名为文献[10]；传统 Item_CF 命名不变。

5.2.1　HAttrTime_CF 算法的参数确定

首先，确定最近邻的个数 k，以及为目标用户推荐的项目数量 N，为了更好地进行实验对比，直接设置常见的 k＝30，N＝3。其次，需要确定式(7)中的权重因子 α，本文分别令 α 取 0，0.1，…1，在完整数据集下，通过 HAttr_CF 算法寻找合适的值。图 1 反映的是不同权重因子 α 下 HAttr_CF 算法的准确率，可以看出，当 α＝0.2 时准确率最高，因此式(7)中的权重因子 α 取值为 0.2。

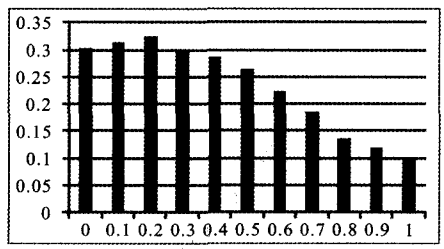

图 1　不同 α 值对 HAttr_CF 算法的影响

最后，确定式(9)中的动态调整因子 β 以及公式(10)时间窗 T 的大小。本实验采用 HAttrTime_CF 算法的准确率寻找这两个待定参数的合适值。经实验验证，当 β＞0.5，时间窗 T＜30 天，或者时间窗 T＞60 天时对推荐的准确度影响不大，因此重点对比了 T＝30，T＝40，T＝50，T＝60 四种情况下，β 分别取 0.1，0.2，…，0.5 时，HAttrTime_CF 算法的准确率。由图 2 可知，当 T＝40，β＝0.4 时算法 HAttrTime_CF 的准确率达到峰值，因此本文算法当中令 T＝40，β＝0.4。

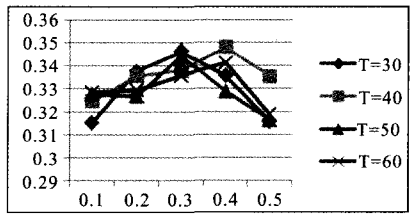

图 2　T 对 HAttrTime_CF 算法准确率的影响

5.2.2　对比试验

为了验证不同数据稀疏度对算法准确率的影响，本实验从数据集 100 万条评分记录中分别随机抽取 10 万，20 万，…，90 万，以及完整数据集，并分别按照 4∶1 的比例随机划分为训练集与测试集，计算不同稀疏度情况下，HAttrTime_CF、文献[10]以及 Item_CF 三种算法的推荐准确率，如图 3 所示。

对比本文提出的 HAttrTime_CF 算法与文献[10]算法及传统的 Item_CF 可以发现，当评分数据量小于约 60 万条时，HAttrTime_CF 算法的准确率明显要高于其它两种算法，说明在算法中融合项目属性能够增强算法对不同数据稀疏度的适应性，缓解数据稀疏问题；当评分数据量在约 60 到 70 万条时，HAttrTime_CF

的准确率与文献[10]比较接近，这主要是因为随着评分数据量的增加，完全依赖于评分相似度作出推荐决策的准确度要高于完全依赖于属性相似度；而当评分数据量超过 70 万条时，HAttrTime_CF 的准确率比文献[10]算法有了一定幅度的提高，这表明采用本文方法考虑不同用户群体兴趣变化的权重，需要在评分数据超过一定量时，才能达到一定效果。最后，从整体上看，HAttrTime_CF 算法和文献[10]算法的性能都要明显优于传统的 Item_CF，这说明在算法中考虑用户兴趣变化的时间因素尤为必要。

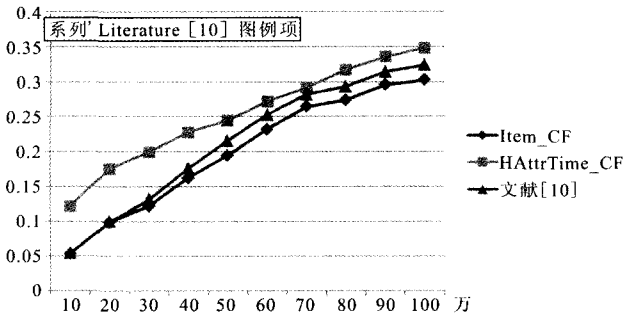

图 3 不同稀疏度下三种算法性能的对比

6 结语

本文对业界应用最为广泛的基于项目的协同过滤算法提出了两点改进。实验结果表明，这两点改进不仅能够缓解评分数据稀疏问题，而且还能够提高算法的准确率。由于准确率并不是衡量推荐算法优劣的唯一指标，下一将进一步考虑对其它指标，如覆盖率、惊喜度等的影响，从而提高推荐算法的综合性能。

参考文献：

[1] 项亮. 推荐系统实践[M]. 北京：人民邮电出版社，2012.

[2] SARWAR B, KARYPIS G, KONSTAN J, et al. Item—based collaborative filtering recommendation algorithms[C]. 2001:285-295.

[3] 任磊. 一种结合评分时间特性的协同推荐算法[J]. 计算机应用与软件，2015，32(5)：112-115.

[4] BILLINGS S A, LEE K L. Nonlinear fisher discriminant analysis using a minimum squared error cost function and the borthogonal least squares algorithm[J]. Neural Networks, 2002, 15(2): 263-270.

[5] MA H, KING I, LYU M R. Effective missing data predicition collaborative filtering[C]. New York: ACM Press, 2007: 39-46.

[6] XU JIANHUA, ZHANG XUEGONG, LI YANDA. Kernel MSE algorithm: a Unified framework for KFD, LS-SVM and KRR[C]. Washington D. C: IEEE Press, 2001: 1486-1491.

[7] SARWAR B, KARYPIS G, KONSTAN J, et al. Application of dimensionality reduction in recommender system-a case study[R]. 2000.

[8] GOLDERG K, RODER T, GUPTA D, et al. A constant time collaborative filtering algorithm[J]. Information Retrival, 2001, 4(2): 133-151.

[9] 邢春晓，高凤荣，战思南，等. 适应于用户的兴趣变化的协同过滤推荐算办法[J]. 计算机研究与发展，2007，44(2)：296-301.

[10] 李克潮，梁振友. 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，2011，47(33)：154-156.

[11] 于洪，李转运. 基于遗忘曲线的推荐算法[J]. 南京大学学报：自然科学版，2010，46(5)：520-527.

[12] 彭石，周志彬，王国军. 基于评分矩阵预填充的协同过滤算法[J]. 计算机工程，2013，39(1)：175-182.

[13] 黄莹，宋伟伟，邓春玲，等. 协同过滤算法在电影推荐系统中的应用[J]. 软件导刊，2015，14 (8)：92-93.

[14] 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，2003，14(9)：1621-1628.

[15] 王全民，刘 鑫，朱蓉，等. 一种新型的混合个性化推荐算法[J]. 计算机与现代化，2013(8)：63-67.

[16] 李克潮，蓝冬梅. 一种属性和评分的协同过滤混合推荐算法[J]. 计算机技术与发展，2013，23(7) ： 116-119.

（责任编辑：陈福时）

A Kind of Personalized Collaborative Filtering Hybrid Recommendation Algorithm

Abstract: Traditional collaborative filtering algorithm exists poor recommendation quality for recommending to the user based solely on sparse rating matrix. A new comprehensive item similarity measurement algorithm based on information entropy is proposed in this paper to dispose the data sparse problem. Meanwhile, taking into account the user's interest will change over time, and the change is not same in different user groups, this paper put forward the weight of adapt to different user interest changes, which is inspired by Ebbinghaus's memory rule. The performed experiment based on the dataset of movielens shows that the modified algorithm can not only alleviate the problem of rating data sparse, but also can improve the accuracy of the algorithm.

Key Words: Recommend System; Collaborative Filtering; Item Attribute; Interest Change; Information Entropy