基于用户兴趣传播的协同过滤方法

高建煌 陈恩红 刘 淇
（中国科学技术大学计算机科学与技术学院）

摘 要：推荐系统帮助用户过滤无用信息并预测其可能感兴趣的产品。在推荐系统中，协同过滤是应用最为广泛的方法之一。然而，传统的协同过滤方法是在产品维度上计算用户相似度，而且在计算相似度时无法考虑邻居用户的影响。因此，该类方法往往受到高维度、数据稀疏等问题的困扰。为此，本文提出一种基于用户兴趣传播的协同过滤方法，在兴趣维度上计算用户相似度，同时考虑了兴趣在不同用户间的传播。该方法不仅可以有效防止冷启动和数据稀疏问题，而且具有较高的预测准确度。在标准数据集MovieLens上的测试结果表明了本文算法的有效性。

关键词：推荐系统；协同过滤；兴趣传播；随机游走

User Interests Transmission Based Collaborative Filtering Approach

Gao Jianhuang Chen Enhong Liu Qi
(School of Computer Science and Technology, University of Science and Technology of China)

Abstract: Recommender systems help users filter useless information and predict the products users may like. Collaborative filtering is one of the most widely used approaches in recommender systems. However, the traditional collaborative filtering methods compute users' similarities in the dimension of products, and they do not take the influence of neighbor users into consideration when computing such similarities. Thus, they often suffer from the problems such as high dimensionality and data sparseness. To that end, we propose a novel collaborative filtering method based on user interests transmission. This method computes users' similarities in the dimension of interests, and takes the interests transmission between different users into consideration. This method not only can cope with "cold start problem" and data sparse problem, but also have higher prediction precision. Experiments on benchmark dataset, MovieLens, show the effectiveness of the proposed method.

Keywords: recommender system; collaborative filtering; interest transmission; random walk

0 引言

推荐系统根据用户与系统的交互历史以及用户的个人信息等构建用户的兴趣模型，预测用户可能感兴趣的产品，不但节省了用户的时间，提高了产品交叉销售的概率，而且通过提供满足用户要求的产品还可以加强用户的忠诚度和购买欲望。目前在推荐系统方面有很多有意义的研究[1]，包括基于内容的推荐[4]、基于协同过滤的推荐[2,3]和混合推荐[5]等。

其中，基于协同过滤(CF)的方法只根据用户与系统的交互历史以及其它用户的喜好进行推荐，不需要去挖掘产品和用户的内容信息(例如，表示产品的关键字或用户的个人信息等等)。与其它两类方法相比，CF受到的限制较少，且具有较好的推荐效果，因而得到了最广泛的应用：Amazon.com[2]利用CF推荐书籍、CD等产品，GroupLens[3]为用户推荐新闻和电影等。

CF又可以分为User-based[7]协同过滤和Item-based[13]协同过滤，前者认为给定用户会喜欢那些与他们有相似喜好的用户所喜爱的产品[3]，而后者则认为给定的产品会被那些喜欢与它们相似的产品的用户所喜欢[2]。

尽管CF方法拥有简单实用等优点，然而传统User-based CF方法一直受到两方面的局限：首先，用户相似度需要在产品维度进行计算，而产品所组成的向量往往维度很高且极为稀疏，所以计算得到的相似度结果与用户的实际相似度会存在一定偏差；其次，在计算用户相似度时，只能考虑两个用户向量的直接相似度，不能考虑用户的间接影响。而在实际生活中，一个用户的兴趣很容易受到周围朋友的影响，如果计算过程不能考虑到这些间接因素，必然会降低预测的准确度。

为了克服以上不足，本文提出一种新的基于用户兴趣传播(User Interests Transmission,UIT)的协同过滤算法。UIT用K维兴趣向量来表示用户，每个用户都是该兴趣向量上的一个概率分布，而且此概率分布受到其它用户概率分布的影响，通过计算概率分布间的相似度就可得到用户的相似度。在算法具体实现过程中，首先通过对用户－产品二部图进行用户维度上的投影得到用户关联图，然后为每个用户随机赋予一个初始兴趣并让用户的兴趣向量在用户关联图上进行随机游走，从而更新用户兴趣分布，最后利用更新后的兴趣向量计算用户相似度，进行User-based CF。

本文通过在一个推荐系统的标准数据集MovieLens[6]上的实验结果验证了UIT算法比传统的CF算法有更高的推荐准确度，而且可以有效防止产品维度过高和数据稀疏等问题。

接下来，本文将首先回顾一下相关工作，在第2节详细地介绍UIT算法模型；在第3节，对UIT算法与传统的协同过滤方法进行对比实验，并分析实验结果；最后总结全文。

1 相关工作

User-based CF推荐算法可以分为两类：基于记忆的方法和基于模型的算法[1]。基于记忆的算法根据系统中所有被打过分的产品信息来对未评分产品进行预测。其中，K-近邻法(KNN)揭示了这类算法的基本流程，它包含以下三个部分：首先，计算其他用户与给定用户的相似度大小，然后选择与给定用户相似度最大的前K个用户，作为该用户的邻居集，最后利用邻居用户对目标产品的

评分信息来预测目标用户对该产品的评分值。

基于记忆的CF算法之间的最大差别在于它们使用不同的方法计算用户的相似度，如夹角余弦法和Pearson相关性方法[3,7]等。近来仍不断有学者在研究如何改进计算用户相似度的方法，如加入时间因素[8]等。然而现存方法往往在产品维度计算用户相似度且不能够考虑邻居用户的影响，所以这些算法的性能仍然有很大的提升空间。

近年来兴起的基于图结构的CF方法是一类基于模型的CF算法，它们可以更准确地计算用户直接相似[11]，或者用图论算法直接描述用户的间接关联[9]。尽管这些方法可以取得良好的推荐效果，然而它们的可解释性不如基于记忆的CF算法，所以很少应用到实际情景中。

基于以上分析，本文提出了基于用户兴趣传播的CF算法UIT，它结合图结构算法[10]和传统基于记忆的CF算法的优点，而且UIT引入兴趣代替产品进行计算，在提高了算法推荐准确性的同时，还加强了算法的可解释性。

2 UIT算法简介

为了有效防止传统协同过滤存在的产品维度过高和数据稀疏性等问题，本文提出了一种基于用户兴趣传播的协同过滤算法UIT。本节将对UIT算法进行详细介绍。2.1节介绍二部图构建及投影；2.2节介绍用户兴趣向量的建立与更新；2.3节介绍相关的基于兴趣的用户相似度计算公式；最后，2.4节介绍如何根据用户相似性计算结果预测用户对产品的评分值并给出UIT算法的总体描述。

2.1 二部图构建及投影

基于图结构的推荐算法指出可以利用二部图(Bipartite Graph)来建立用户－产品的关联关系[10]。

定义用户集合U和产品集合P，构建以集合$U\cup P$为顶点集的用户－产品二部图。其中，如果用户u对产品p打过分，且评分值为r_{up}，就在u和p之间连接一条边且边权重$a_{up}=r_{up}(u\in U,p\in P)$。将该二部图在用户维度上进行投影，得到对应的用户投影图，即用户关联图。在用户投影图中，用户u到v的连边权重$w(u,v)$表示用户u对用户v喜好的影响程度，其计算公式如下：

$$w(u,v)=\frac{1}{k_u}\sum_{p=1}^{|P|}\frac{a_{up}a_{vp}}{k_p} \tag{1}$$

其中，$k_u=\sum_{p=1}^{|P|}a_{up}$，$k_p=\sum_{u=1}^{|U|}a_{up}$。

定义用户关联矩阵$RM_{|U|\times|U|}$(Relation Matrix)，令$RM_{uv}=w(v,u)$，并对RM以行为单位进行归一化，则RM对应于用户关联图的关联矩阵。其中RM_{uv}表示用户u对于用户v的关联系数，即在所有用户对于v喜好的影响之中，用户u的影响所占的比例。

2.2 用户兴趣向量的建立及更新

为每个用户定义一个K维兴趣向量，每个用户都是该兴趣向量上的一个概率分布。该兴趣向量是归一化的，即对任何用户，他属于各个兴趣的概率之和为1。在UIT中，对每个用户随机赋予一个初始兴趣分布。可以用矩阵$IM_{|U|\times|K|}$(Interest Matrix)保存所有用户的K维兴趣向量，称该矩阵为用户兴趣矩阵。

初始的随机兴趣分布显然不能反映用户的真实兴趣。为此，我们模拟现实场景，认为用户兴趣受到周围朋友的影响并通过让用户的兴趣向量在用户关联图上进行随机游走来更新用户兴趣分布，学习用户的真实兴趣。设第i步随机游走后，用户兴趣矩阵更新为IM^i，则有：

$$IM^i=RM\times IM^{i-1} \tag{2}$$

其中，$IM^0=IM$。

经过step步随机游走以后，最终得到用户兴趣矩阵为IM，即IM^{step}。

2.3 基于兴趣的相似度计算

在user-based CF算法中，必须通过计算用户间的相似情况来寻找与给定用户相似的邻居集。传统的协调过滤方法经常采用基于产品维度的Pearson相关性或夹角余弦相关性来计算用户相似度[7]。然而，由于产品所组成的向量往往维度很高且极为稀疏，而且在计算用户相似度时只能考虑两个用户向量之间的直接相似度，不能考虑其他用户的间接影响。因此，此类方法计算得到的相似度结果与用户的实际相似度往往存在较大的偏差。

为了更准确描述用户之间的相似性，UIT采用基于兴趣维度的Pearson相关性或基于兴趣维度的夹角余弦相关性来计算用户之间的相似度。

定义用户u和v之间的基于兴趣维度的Pearson相关性为：

$$\mathrm{sim}(u,v)=\frac{\sum_{i=1}^{K}(IM_{u,i}-\overline{IM_u})(IM_{v,i}-\overline{IM_v})}{\sqrt{\sum_{i=1}^{K}(IM_{u,i}-\overline{IM_u})^2\sum_{i=1}^{K}(IM_{v,i}-\overline{IM_v})^2}} \tag{3}$$

其中，K为用户兴趣维数，$IM_{u,i}$和$IM_{v,i}$分别为用户u和v在第i个兴趣上的概率分布。$\overline{IM_u}=\overline{IM_v}=1/K$，二者分别为用户$u$和$v$在$K$维兴趣向量上的概率均值。

类似地，可以定义用户u和v之间的基于兴趣维度的夹角余弦相关性为：

$$\mathrm{sim}(u,v)=\frac{\sum_{i=1}^{K}IM_{u,i}IM_{v,i}}{\sqrt{\sum_{i=1}^{K}IM_{u,i}^2\sum_{i=1}^{K}IM_{v,i}^2}} \tag{4}$$

2.4 评分值预测及算法总流程

CF算法的最终目的是预测给定用户对未评分产品的评分值，即对给定用户u及产品p，预测用户u对产品p的可能评分值。

为此，在UIT中，首先选取与用户u相似度大于0且已对产品p进行评分的用户组成其邻居集NS_u，然后利用该邻居集对产品p的评分信息来预测用户u对p的评分值$r_{u,p}$。预测公式如下：

$$r_{u,p}=\overline{r_u}+k\sum_{v\in NS_u}\mathrm{sim}(u,v)\cdot(r_{v,p}-\overline{r_v}) \tag{5}$$

其中，k为[illegible]，$\overline{r_u}$为用户u的平均评分值。根据定义有：

$$k=\frac{1}{\sum_{v\in NS_u}\mathrm{sim}(u,v)}\quad，且\quad\overline{r_u}=\frac{\sum_{s\in P_u}r_{u,p}}{|P_u|}$$

其中$\mathrm{sim}(u,v)$表示用户u和v之间的相似度，$P_u=\{p\in P|r_{u,p}\neq 0\}$。

图1给出了UIT算法的总体描述。(参见下页)

3 实验结果及分析

这一节将详细介绍本文的实验设计及相关结果。首先，介绍本文使用的基准数据集；其次，介绍采用的性能评价标准；3.3节给出相关的实验结果；最后，在3.4和3.5节中分别讨论steps参数和K参数对UIT算法性能的影响。

输入：用户评分矩阵 RM，用户兴趣维度个数 K，随机游走趟数 steps
输出：用户评分预测值矩阵 PM

1. 利用矩阵 RM，构建用户-产品二部图 BG；
2. 应用公式(1)对 BG 进行用户维度上的投影，得到用户关联图，进而得到用户关联矩阵 RM；
3. 为每个用户建立一个 K 维兴趣向量，并随机赋予一个初始兴趣分布。将所有用户的兴趣分布存储到用户兴趣矩阵 IM 中；
4. for i: 1->steps
5. IM = RM×IM；
6. 利用公式(3)计算各个用户之间的相似度，将结果存入用户相似度矩阵 SM 中；
7. 对每个用户 u
8. 对每个产品 p
9. {
10. if(用户 u 还未对产品 p 进行评分)
11. 计算得到用户 u 的邻居集 NS_u；
12. 利用公式(5)计算用户 u 对产品 p 的评分预测值，并将结果保存到用户评分预测矩阵 PM 中；
13. }
14. 返回 PM；

图1 UIT算法总体流程描述

3.1 数据集

本文使用Minnesota大学GroupLens项目组提供的基准数据集MovieLens[6]来验证算法的有效性。该数据集包含943个用户对1682部电影的100000条评分记录，其中每个用户至少对20部电影进行了评分，评分值范围从1到5。

在实验中，把每个用户的评分序列划分成训练集和测试集两部分，即以一定的概率随机选取其中一部分评分记录作为训练集，剩余评分记录作为测试集。本文通过调整训练集和测试集的大小比例来验证算法在不同稀疏度的数据上的性能。实验中采用x-(100-x)的方式来表示训练集和测试集的大小比例情况，如"10-90"表示"随机抽取每个用户10%的评分数据加入训练集，其余的90%评分作为测试数据"。在实验中共设定了九组数据，分别为10-90、20-80、30-70、40-60、50-50、60-40、70-30、80-20、90-10。

3.2 评价标准

本文采用平均误差(Mean Absolute Error, MAE)[12]作为算法性能的评价标准。MAE用于度量预测值与实际值之间的平均偏差，偏差越小，预测的精度越高，推荐的质量越高。MAE的定义如下：

首先，计算每个用户u的MAE(u)：

$$\mathrm{MAE}(u)=\frac{\sum_{p\in T(u)}\left|p_{u,i}-r_{u,i}\right|}{\left|T(u)\right|} \tag{6}$$

其中，$T(u)$为测试集中用户u的评分记录。然后，取所有用户MAE(u)的平均值作为总体MAE值：

$$\mathrm{MAE}=\frac{\sum_{u\in U}\mathrm{MAE}(u)}{\left|U\right|} \tag{7}$$

3.3 实验结果与分析

为了验证UIT算法的有效性，在九组不同划分方式的数据集上，本文均选取CF的两个经典算法UserBased[7]和ItemBased[13]作为对比实验。在实验中，K值统一设置为70，steps的初始值设为1，然后逐步增大steps，当得到的MAE值不再发生明显变化的时候，选取该MAE值作为UIT算法的最终MAE值，实验结果如表1所示：

表1 各类算法得到的MAE

Split.\ Alg.	UserBased	ItemBased	UIT
10-90	0.918966	0.930372	**0.844801**
20-80	0.832381	0.811989	**0.793331**
30-70	0.797882	0.781191	**0.774825**
40-60	0.772153	0.766058	**0.764102**
50-50	0.756196	0.758187	**0.754418**
60-40	0.751419	0.752092	**0.750129**
70-30	0.744612	0.750242	**0.743348**
80-20	0.741295	0.748702	**0.740650**
90-10	0.746532	0.755220	**0.745787**

从表1可以看出，在九组不同划分方式的数据集上，UIT算法的MAE值均小于经典的UserBased算法和ItemBased算法。特别在训练集数据比较稀疏的情况下，UIT算法的优势更加明显。这是由于UIT算法在较低的兴趣维度上进行用户相似度的计算，同时它考虑了周围朋友对用户兴趣的影响。因此，在训练数据比较稀疏的情况下，UIT算法能够从用户兴趣传播这种模式中获取更多的信息，使得计算得到的用户相似度比传统的协同过滤更加精确。

3.4 参数steps对UIT算法性能的影响

本小节以10-90数据集为例分析随机游走步数(steps)对UIT算法性能的影响情况。与3.3小节相同，实验中设置兴趣维度K=70，得到的steps-MAE结果如图2所示。从图中可以看出：当0＜steps≤4时，随着steps的增大，MAE值有所降低，但幅度很小，始终保持在0.9以上；当4＜steps≤8时，随着steps的增大，MAE值急剧降低，这说明在这段区间内，每一步随机游走都能带来大量的额外信息，使得计算得到的用户相似度更接近实际，因此大大地提高预测的准确性；当steps＞8时，由于此时用户的兴趣已经趋于稳定，继续增大steps基本上不再影响MAE的变化。

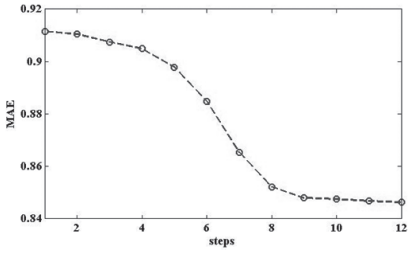

图2 steps对UIT算法性能的影响

3.5 参数K对UIT算法性能的影响

本小节以90-10这组数据集为例来分析用户兴趣维度大小(K)对UIT算法性能的影响，实验中设置steps=1，得到的K-MAE结果如图3所示。从图中可以看出：当0＜K≤100时，随着K的增大，MAE值逐步降低；当K=100时，MAE值降到最低，UIT算法性能达到最佳；当K＞100时，随着K的增大，MAE开始逐步升高。

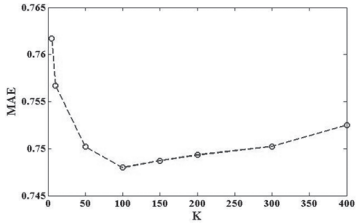

图3 steps对UIT算法性能的影响

4 结束语

传统的CF方法在产品维度计算用户相似度，而且在计算用户相似度时不考虑邻居用户的影响，因此往往受到高维度、数据稀疏等问题的困扰。针对此类问题，本文提出一种基于用户兴趣传播的CF方法，此方法在兴趣维度计算用户相似度，同时考虑了兴趣在不同用户间的传播。通过在MovieLens数据集的九组不同划分上进行与传统CF方法的对比实验证实，此方法不仅可以较有效地防止冷启动和数据稀疏问题，而且具有更高的预测准确度。

参考文献：

[1] Adomavicus G,Tuzhilin A.Toward the next generation of recommender systems:A survey of the state-of-the-art and possible extensions [J].IEEE Trans on Knowledge and Data Engineering,2005, 17(6):734-749.
[2] Linden G,Smith B,York J.Amazon.com Recommendations: Item-to-item collaborative filtering [J].IEEE Internet Computing, 2003(1):76-80.
[3] Resnick P,Iacovou N,Suchak M,et al.Grouplens:an open architecture for collaborative filtering of Netnews [C]// Proceedings of the CSCW conference,Chapel Hill,NC, 1994:175-186.
[4] Debnath S,Ganguly N,Mitra P.Feature weighting in content based recommendation system using social network analysis [C]//Proceedings of WWW,Beijing,China:April, 2008:1041-1042.
[5] Balabanovic M,Shoham Y.Fab:Content-based,collaborative Recommendation [J].Comm.ACM,1997, 40(3):66-72.
[6] MovieLens datasets[DB/OL]. http://www.grouplens.org/node/73#attachments,%202007.
[7] Breese J,Heckerman D,Kadie C.Empirical analysis of predictive algorithms for collaborative filtering[C]// Uncertainty in Artificial Intelligence.Proceedings of the Fourteenth Conference,Morgan Kaufman,1998:43-52.
[8] Gong S,Cheng G.Mining user interest change for improving collaborative filtering[C]//Proceedings of the Workshop on Intelligent Information Technology Application (IITA'08), IEEE Computer Society,2008:24-17.
[9] Liu Q,Chen E H.Collaborative filtering through combining bipartite graph projection and ranking.In Journal of Chinese Computer Systems(In Chinese).(Accepted).
[10] Zhou T,Ren J,Medo M,et al.Bipartite network projection and personal recommendation[J].Physical Review E, 2007, 76(4):046115.
[11] Fouss F,Pirotte A,Renders J -M,et al.Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation.IEEE Trans.Knowl.Data Eng.,2007, 19(3):355-369.
[12] Shardanand U,Maes P.Social information filtering: Algorithms for automating "word of mouth[C]// Proceedings of the ACM CHI Conference on Human Factors in Computing Systems Denver,CO,USA,May 1995,ACM Press:210-217.
[13] SarwarB M,Karypis G,Konstan J.A,et al.Item-based collaborative filtering recommendation algorithms[C]// Proceedings of the 10th International WorldWideWeb Conference(WWW10) Hongkong,May 1-5, 2001:285-295

作者简介：
高建煌，1984年生，硕士研究生，主要研究方向为推荐系统、数据挖掘
电话：0551-3601551；13855126043
电子信箱：gjh@mail.ustc.edu.cn
联系地址：安徽 合肥 中国科技大学西区 10#221 高建煌 收 (230027)
陈恩红，1968年生，教授、博士生导师，主要研究方向为信息检索、语义计算与数据挖掘
电子信箱：cheneh@ustc.edu.cn
刘淇，1986年生，博士研究生，主要研究方向为推荐系统、数据挖掘
电话：0551-3601551
电子信箱：feiniaol@mail.ustc.edu.cn

项目基金：
本文得到国家自然科学基金资助(批准号：60775037)

(上接第6页)

的业务管理研究[J]。北京邮电大学学报, 2004(S2):190-195.
[4] 吴雷. 基于SOA的工作流引擎的研究与实现[D]. 武汉：武汉理工大学. 2007. 5.
[5] 王伟，张磊，韩毅. 基于EAI的项目管理与工程应用系统集成研究[J]. 计算机工程与应用. 2005, 41(35):79-81.

作者简介：
朱尧富(1980－)湖州人，台州职业技术学院计算机工程系助理实验师，2002年至今从事软件设计与网络应用研究。
电话：13606686667
电子信箱：zhuyaofu@163.com