一种结合用户评分信息的改进好友推荐算法

汤 颖 钟南江 范 菁

(浙江工业大学计算机科学与技术学院 杭州 310023)

摘 要 传统的好友推荐算法在计算好友相似度时通常仅仅考虑用户在社交网络的拓扑结构的相似性,而对用户的兴趣相似性考虑较少,因此推荐的结果往往不够精准。现有的很多社交网站(如豆瓣网)提供了用户评分功能,用户可以对某类物品(如电影)给出自己的评分。为了在推荐时计算用户的兴趣相似度,提出基于用户给出的对某类物品的评分来计算用户的兴趣相似度,从而在拓扑相似度的基础上结合兴趣相似度得到更精准的推荐结果。首先使用余弦相似度计算出用户间拓扑相似度;其次在计算基于评分的用户兴趣相似度时,通过建立概率模型得到用户聚类评分相似度矩阵,从该评分矩阵推导出用户间基于评分的兴趣相似度;最后,结合拓扑相似度和评分相似度得到最终的改进好友推荐算法,计算出相似度值最高的 N 个人推荐给当前用户。为了验证所提方法的有效性,用提出的方法对豆瓣网抓取的用户数据进行好友推荐,实验结果证明所提方法与传统的基于拓扑的好友推荐算法相比可以有效提高好友推荐的准确性。

关键词 社交网络,推荐,拓扑结构,评分,聚类,相似度

中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.9.021

Improved Friends Recommendation Algorithm Combining with User Rating Information

TANG Ying ZHONG Nan-jiang FAN Jing

(College of Computer Science & Technology, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract Traditional friends recommendation algorithms only consider the topological similarity in calculating the similarity of friends. The similarity of users' interests is seldomly taken into account, and the recommendation results are often not precise enough. Many existing social networking sites (such as Douban.com) provide the functions of user ratings, i.e. users can give ratings for certain types of items (such as movies). In order to calculate user's interests similarity, we proposed a method which computes the interests similarity based on the ratings given by users and got more accurate result of recommendation by incorporating the interests similarity into the topological similarity. Firstly, we used cosine similarity to calculate the topological similarity between users. In calculating the interests similarity based on user ratings, we got the users cluster rating similarity matrix through the establishment of a probabilistic model, and derived users' interests similarity from the rating similarity matrix. Finally, users' interests similarity and topological similarity were combined to get the final improved friends recommendation algorithm. In order to verify the effectiveness of our method, we applied our method to the crawled user data from Douban website. The experimental results show that our method can effectively improve the accuracy of recommendation results compared with the traditional recommendation algorithm based on topology similarity.

Keywords Social network, Recommendation, Topology, Rating, Cluster, Similarity

1 引言

在社交网络服务中,好友推荐是一项非常热门的功能,它能帮助用户认识新的朋友,扩展以他们为中心的社交圈子。通常我们可以根据用户在社交网络中的拓扑结构进行好友推荐,但随着社交网络的发展,越来越多的信息能够用来帮助提高推荐算法的准确性以及支持新的推荐任务,例如可以利用用户的个人资料做出好友推荐,比如用户的教育以及职业背景。但是有些社交网络为了保护用户隐私,并不会将用户的个人资料公开,这时就需要利用社交网络中的其他个人数据来提升推荐系统的性能,例如,利用标签信息结合用户拓扑结构[1],利用用户的兴趣相似度来提升推荐性能[2]。

目前有许多用户兴趣的相似度计算方法,但是它们大多基于用户标签[1],或者基于用户个人资料[3],而对于类似豆瓣网带有评分功能的社交网络没有用户标签,也不公开的用户个人资料,因此需要寻找其他方法。本文针对这一情况提出了基于评分的用户兴趣相似度计算方法。许多社交网络中都有评分功能,用户可以对电影、音乐或者图书等物品进行评

到稿日期:2015-07-07 返修日期:2015-08-19 本文受国家自然科学基金(61003265),浙江省自然科学基金(LY14F020021),国家科技支撑计划(2014BAH23F03)资助。

汤 颖(1977—),女,博士,副教授,CCF会员,主要研究方向为信息可视化、虚拟现实,E-mail:tangying@gmail.com;**钟南江**(1991—),男,硕士生,主要研究方向为数据分析,E-mail:znj8059143@163.com;**范 菁**(1969—),博士,教授,CCF会员,主要研究方向为虚拟现实、软件工程,E-mail:fanjing@zjut.edu.cn(通信作者)。

分，评分高低代表用户对物品的喜好程度。用户的评分越相似，表示用户在此领域的兴趣也越相似，因此可以利用用户的评分数据计算出用户之间的兴趣相似度。更加具体地来说，使用概率模型对用户的评分数据进行建模，同时对用户和物品聚类，计算出每个用户聚类对物品聚类的评分，通过计算用户属于每个聚类的概率计算出两两用户间基于评分的兴趣相似度（本文将用“评分相似度”作为“基于评分的兴趣相似度”的简称）。

传统好友推荐方法计算用户相似度时多是基于拓扑结构，它主要向用户推荐同属于一个好友圈子（同一所学校、同一个工作单位等）的潜在好友。这种方法很有效，因为用户更有可能和同一个圈子的人成为好友，但也有局限性，因为有时用户之间能成为好友是因为趣味相投，有相同的兴趣爱好。本文通过对传统的基于拓扑结构的方法进行改进，使其不仅能基于拓扑结构为用户推荐同一个圈子内的潜在好友，也能基于评分信息为用户推荐相同兴趣爱好的人，使其在带评分功能的社交网络中能进行更加精准的好友推荐。

本文第2节将简单介绍社交网络推荐的相关工作；第3节阐述本文提出的改进的用户相似度的计算方法；第4节给出实验设计来验证本文方法的有效性；最后总结全文并对未来工作进行展望。

2 相关工作

近年来社交网络推荐成为学术界的热门话题。本节主要回顾基于用户信息的推荐和基于网络拓扑结构的推荐，其中在基于用户信息的推荐中既有针对好友推荐的方法，也有针对物品推荐的方法。

2.1 基于用户信息的推荐

基于用户个人信息的推荐系统可以个性化地为用户推荐好友或者一些有价值的信息。这些系统通常使用用户的个人信息文件的集合，然后根据他们的个人信息计算与其他人之间的相似性。个人信息文件或偏好由明确要求用户添加的属性信息（如年龄、性别）或追踪用户的行为（如网页访问、采购的历史）得到。目前主要有两种基于用户信息的推荐方法：基于内容的过滤和协同过滤。

基于内容的过滤方法从用户的属性信息中匹配用户间的相同偏好或者属性，例如他们的自我描述和人口统计学的数据[3]。用户的个人信息可以用由一些关键项组成的向量来表示，用户信息的相似度可以用一些机器学习的方法来计算，如朴素贝叶斯分类器[4]。传统的基于内容的过滤在某些情况下难以定义一些合适的描述符，比如在电子商务中，用户可能难以察觉到或者至少是疏忽了自己的爱好。在这种情况下，预测用户参数以及向用户推荐商品或者好友就不能使用用户明确给出的信息了。

协同过滤通过收集用户的口味信息（对物品的评分）来对用户的兴趣爱好进行动态的推荐。例如，书籍的协同过滤可以根据用户的书籍口味以及书中其他用户的口味来预测用户可能喜欢看哪些书籍。协同过滤的方法也可以分为基于记忆的协同过滤和基于模型的协同过滤两种。

基于记忆的协同过滤使用成对的用户间的相似度[5]或者物品间的相似度[6]来做推荐（多是物品推荐）。传统的使用用户间相似度做推荐的方法的基本步骤为：1）识别出和当前用户相似的用户；2）通过这些相似用户的购买行为做出预测和推荐。这种算法生成推荐时基于一小部分与当前用户最相似的用户。基于记忆的方法很受欢迎，因为它在概念上比较简单，所以避免了在复杂的模型建立阶段的种种潜在的问题。同时，它被认为能解决现实世界中的很多问题。但是现在它存在着下面几个方面的缺点：1）基于记忆的方法在最终的正确率方面还没有达到最理想；2）基于记忆的方法效率不高，占用内存多，计算时间长；3）很难系统地调整基于记忆的算法来完成一个特定的任务，即灵活性不好。

以模型为基础的协同过滤（Model-based Collaborative Filtering）先用历史资料，使用数据挖掘或者机器学习的方法得到一个概率模型。文献[8]将贝叶斯概率模型应用到社交网络的电影推荐中。潜在语义模型[7]是统计方法——潜在语义分析（pLSA）[9]的泛化。它和贝叶斯模型的不同主要在于后者直接使用观察到的数据来建立模型，而潜在语义模型是基于一个潜在的原因来构造模型。文献[10]是潜在语义模型的扩展，主要提出了一个基于社会影响力的方法来进行推荐，即不仅仅对用户的偏好建模，更考虑到了用户朋友的偏好对其的影响。FMM [15]引入了两个潜在变量，可以根据用户的评分信息同时对用户、物品聚类，以此来做用户未知评分预测。

2.2 基于拓扑结构的推荐

基于拓扑结构的推荐方法使用网络结构中的内部属性来定义节点之间的相似度。经常使用的基于拓扑结构的推荐方法包括 Jaccard 相似度[11]和余弦相似度[12]，它们都是建立在同一个假设上，即两个节点相同的邻居节点越多就越相似。文献[13]提出了一种改进算法，为不同的节点分配不同的权重（例如，节点的度越低权重越高）。

基于拓扑结构的推荐也可以利用拓扑的附加属性[16]，比如拓扑结构中新的链接更有可能和度更高的节点连接。文献[17]使用局部的拓扑结构计算用户的相似度；文献[18]使用全局的拓扑信息计算用户间的拓扑相似度，即使用网络中所有的节点之间的链接，并给所有链接设置一个权重，离当前用户越远的链接权重越低。一些使用全局拓扑的方法更为复杂，包括 SimRank[19]，LHN[20]，P-Rank[21]。

尽管基于全局拓扑结构的方法能通过复杂的计算方法得到更加准确的用户相似度，但是它的计算量却很大。于是，有许多研究者在这方面进行努力，旨在减少计算的复杂度。文献[22,23] 通过分层次的节点聚类简化了 LHN 方法；文献[24]使用增量更新的方法近似地估计了 SimRank 方法得到的值；文献[25]通过找到系统核心用户来降低计算复杂度。

3 改进的用户相似度计算

推荐系统中最关键的部分就是计算当前用户与其他用户之间的相似度值，然后再为当前用户推荐相似度值最高的用户——Top-N，即与当前用户最相似的 N 个用户。

本节主要介绍改进的用户相似度的计算方法。首先介绍用户间拓扑结构相似度的计算方法，然后重点介绍聚类评分矩阵的定义和训练过程、基于聚类评分矩阵的用户评分相似度的计算以及最终结合拓扑和评分的相似度的计算。

3.1 拓扑结构相似度

在社交网络中，有一个重要的假设“物以类聚，人以群分”[14]，即朋友之间总是在一些地方是相似的，比如在同一所学校念书或者在同一家公司上班，也就是说有一个特定的圈子。在这种假设下，使用社交网络中的拓扑结构来度量用户间的相似度是非常有效的。

基于拓扑结构相似性的方法使用余弦相似度计算用户在

社交网络中的相似度。这种方法的意义是：如果两个用户在社交网络中拥有的共同好友在他们的好友总数中占比越大，那么他们就越相似。用户 i 与用户 j 在社交网络中的拓扑结构相似度因此被定义为：

$$TS(u_i,u_j)=\cos(u_i,u_j)=\frac{u_i \cdot u_j}{|u_i| \cdot |u_j|}$$

其中，u_i，u_j 是用户 i 与用户 j 的好友关系向量，向量中某个元素值为 1 时表示其与当前用户是好友关系，为 0 则表示没有联系；$u_i \cdot u_j$ 为两个向量的点积。因此，分子为两个用户共同好友的数量，分母为两个向量欧几里得长度的乘积。

已有研究表明根据拓扑结构相似度来进行好友推荐可以得到具有一定准确性的推荐结果，但是我们发现仅仅利用拓扑结构相似性是不够的。有些用户也许并不在同一个社交圈子，但是他们有着相同的兴趣，不同的圈子有时并不能阻止他们成为好友。比如金庸笔下的曲洋和刘正风，两人分属敌对的阵营（也就是说没有共同好友），但是二人因为共同爱好音乐，趣味相投而成为忘年之交。因此除了拓扑结构外，还应当考虑用户的兴趣来提升推荐的准确率。

3.2 基于评分的用户兴趣相似度

由于豆瓣网没有用户标签信息，而且对用户个人资料保密，但豆瓣网有许多公开的用户评分信息（比如对电影、音乐、图书等的评分），因此设计了一种基于评分的用户兴趣相似度计算方法。

(3,4,2,?,3,5,?)表示某个用户的评分向量，其中数字表示用户对相应电影的评分，问号表示空值。现在有许多方法可以预测出用户对电影评分的值，但即使知道每个评分向量完整的值，直接利用评分向量计算余弦相似度的计算量也是巨大的。本节将介绍所提的计算方法，不需要求出向量中未知的值，就能计算出用户间的基于评分的兴趣相似度。

首先通过图 1 的例子来简要介绍用户评分相似度的大致计算思路，然后再介绍具体的计算过程，即评分概率模型的建立以及如何通过概率模型计算得到用户的评分相似度。

	a	b	c	d	e	f
1	?	3	?	3	2	3
2	3	1	2	2	?	1
3	3	?	2	?	3	1
4	1	?	1	1	1	2
5	2	3	3	?	2	?
6	1	2	?	1	?	2

(a)矩阵(A)

	a	e	b	f	c	d
2	3	?	1	1	2	2
3	3	3	1	1	2	?
1	?	2	3	3	?	3
5	2	2	3	?	3	?
4	1	1	?	2	1	1
6	1	?	2	2	?	1

(b)矩阵(B)

注：矩阵(A)表示原评分矩阵；矩阵(B)表示矩阵(A)经过行、列变换得到的矩阵。

图 1 评分矩阵的行列变换

图 1 中的两个矩阵都表示用户对物品的评分矩阵，其中行为用户，列为物品，矩阵中的值为用户对物品的评分，? 表示未知的评分。从矩阵(A)到矩阵(B)的过程是矩阵的行列变换过程，相当于是一个用户和物品同时聚类的过程，可以看到把评分一样的项聚在了一起。矩阵(B)可以写成如图 2 所示的形式。

	A	B	C
I	3	1	2
II	2	3	3
III	1	2	1

图 2 用户聚类评分矩阵

在图 2 中行 I、II、III 表示 3 个用户聚类，列 A、B、C 表示 3 个物品聚类。矩阵中每一行为其对应用户聚类的评分向量，对评分进行标准化操作后，通过计算用户聚类间评分向量余弦相似度得到用户聚类间的相似度。

根据用户聚类的相似度构建用户聚类评分相似度矩阵，如图 3 所示。计算两个用户属于每个聚类的概率，根据这个用户聚类评分相似度矩阵计算出它们的相似度期望，得到用户间的评分相似度。具体相似度计算方法以及用户聚类评分相似度矩阵生成过程将在下文详细介绍。

	I	II	III
I	1	0.48	0.62
II	0.48	1	0.53
III	0.62	0.53	1

图 3 用户聚类评分相似度矩阵

3.2.1 聚类评分相似度矩阵的构建

首先介绍下文中涉及到的一些参数。用户集 $U=\{u_1 \cdots u_n\}$，物品集 $V=\{v_1 \cdots v_m\}$，其中 n,m 分别表示用户和物品的数量，$D=\{(u_1,v_1,r_1) \cdots (u_s,v_s,r_s)\}$ 表示训练集中的评分信息，s 为评分信息总数，数据集中评分 r 在一个范围内（比如 1～5或 1～10 等）。用户聚类集合 $C_v=\{k_1 \cdots k_p\}$，物品聚类集 $C_v=\{l_1 \cdots l_q\}$，其中 p,q 分别表示用户聚类和物品聚类的数量。$P(k_x)(1 \leqslant x \leqslant p)$ 表示用户聚类 x 的概率分布；$P(l_y)(1 \leqslant y \leqslant q)$ 表示物品聚类 y 的概率分；$P(u_i|k_x)$ 条件概率表示给定聚类 k_x，用户 u_i 的分布；$P(v_i|l_y)$ 条件概率表示给定聚类 l_y，物品 v_i 的分布；$P(r|k_x,l_y)$ 表示给定聚类 k_x，l_y，评分 r 的分布。

那么用户聚类 k_x 对物品聚类 l_y 的评分为：

$$R_{x,y}=\sum_r r \cdot P(r|k_x,l_y) \tag{1}$$

用户 u_i 属于用户聚类 k_x 的概率为：

$$P(k_x|u_i)=\frac{P(u_i|k_x) \cdot P(k_x)}{P(u_i)} \tag{2}$$

为了得到模型(1)、(2)中的概率参数，本文采用期望最大化算法(EM 算法)对模型进行训练，在 E 步写出给定 u_i,v_i,r_i 时，k_x 与 l_y 的联合分布概率：

$$P(k_x,l_y|u_i,v_i,r_i)=\frac{P(k_x)P(l_y)P(u_i|k_x)P(v_i|l_y)P(r_i|k_x,l_y)}{\sum_{a=1}^{p}\sum_{b=1}^{q}P(k_a)P(l_b)P(u_i|k_a)P(v_i|l_b)P(r_i|k_a,l_b)}$$

在 M 步，使用 $P(k_x,l_y|u_i,v_i,r_i)$ 更新 E 步中的 5 个概率参数：

$$P(k_x)=\frac{\sum_{y=1}^{q}\sum_{j=1}^{s}P(k_x,l_y|u_j,v_j,r_j)}{s}$$

$$P(l_y)=\frac{\sum_{x=1}^{p}\sum_{j=1}^{s}P(k_x,l_y|u_j,v_j,r_j)}{s}$$

$$P(u_i|k_x)=\frac{\sum_{y=1}^{q}\sum_{j;u_j=u_i}P(k_x,l_y|u_j,v_j,r_j)}{P(k_x) \cdot s}$$

$$P(v_i|l_y)=\frac{\sum_{x=1}^{p}\sum_{j;v_j=v_i}P(k_x,l_y|u_j,v_j,r_j)}{P(l_y) \cdot s}$$

$$P(r|k_x,l_y)=\frac{\sum_{j;r_j=r}P(k_x,l_y|u_j,v_j,r_j)}{\sum_{j=1}^{s}P(k_x,l_y|u_j,v_j,r_j)}$$

经过 M 步的计算再更新 E 步中的联合分布概率，不断迭代 E 步、M 步直到收敛，得到模型(1)、(2)中的所有概率参数。图 2 中的用户聚类评分矩阵也得以构建。

到目前为止，上述提到的模型都是假定所有用户的评分尺度都一样。但是实际情况却不是这样，对于不同的用户，相同的评分并不意味着相同的喜爱程度，例如5星的评价对于不同的用户来说代表着不同的意义。对于用户聚类来说也是一样，因此需要对评分矩阵中的值做如下变换[7]：

$$R'_{x,y}=\frac{R_{x,y}-\mu_x}{\sigma_x}$$

其中，μ_x 为用户聚类 k_x 对物品聚类评分的均值，σ_x 为用户聚类 k_x 对物品聚类评分的标准差。

根据得到的用户聚类评分矩阵，得到每个聚类的评分向量(矩阵的行)，计算聚类间的余弦相似度，例如用户聚类 x 与用户聚类 y 的聚类评分相似度 $CRS(k_x,k_y)$：

$$CRS(k_x,k_y)=\cos(K_x,K_y)=\frac{K_x\cdot K_y}{|K_x|\cdot|K_y|}$$

其中，K_x,K_y 表示相对应的评分向量。

在计算了所有聚类间的相似度后，得到最终的用户聚类评分相似度矩阵。

3.2.2 *用户评分相似度的计算*

根据得到的用户聚类评分相似度矩阵计算用户评分相似度 $RS(u_i,u_j)$，具体的计算公式如下：

$$RS(u_i,u_j)=\sum_{x=1}^{p}\sum_{y=1}^{p}P(k_x|u_i)P(k_y|u_j)CRS(k_x,k_y)$$

3.3 拓扑结构与评分相结合的相似度

上文介绍了本文拓扑结构相似度以及基于评分的用户兴趣相似度的计算方法。如何将两者做一个有效的结合是关键。本节将阐述用户拓扑结构与兴趣相结合的相似度计算方法。

两个用户之间的拓扑结构与兴趣相结合的相似度将包含上述的拓扑相似度以及基于评分的兴趣相似度。因为拓扑相似度以及评分相似度都是使用余弦相似度来度量，所以可以将用户 u_i,u_j 拓扑结构与兴趣相结合的相似度 $FS(u_i,u_j)$ 表示如下：

$$FS(u_i,u_j)=\alpha TS(u_i,u_j)+(1-\alpha)RS(u_i,u_j)$$

其中，$0\leqslant\alpha\leqslant1$ 是比重的控制参数，表示用户拓扑结构相似度所占的权重。可以根据具体情况来调节此参数，本文在第4节的实验中选取 $\alpha=0.5$ 作为拓扑结构与兴趣相结合的相似度。当 $\alpha=0$ 时，即为仅基于评分的兴趣相似度；当 $\alpha=1$ 时，为仅基于拓扑结构的用户相似度。

4 实验

4.1 数据集和运行时间

实验中所用数据为豆瓣网的真实数据。豆瓣(douban)是一个社区网站，用户之间可以互相关注，并且提供图书、电影、音乐唱片的推荐、评论、评分。本文使用爬虫抓取豆瓣网数据，数据集分为两个部分：1)用户关注信息，包含了用户以及其关注的好友；2)用户电影评分信息，每一条评分数据包括用户ID、电影ID以及评分。整个数据集包括101个用户，2752部电影，总共36521条电影评分数据，587个好友关注关系。将好友的关注关系随机取391个关注关系作为训练集，剩余的196个关注关系作为测试集。

本文算法主要分为两个部分：离线训练和在线推荐，以保证推荐的实时性。其中离线训练主要使用EM算法训练概率模型，得到评分相似度矩阵。在线推荐部分计算用户间的拓扑相似度，并且利用离线训练得到的相似度矩阵计算用户间的评分相似度值，再整合两者得到最终的相似度值，并为当前用户推荐Top-N个潜在好友。

本文方法在Windows 7操作系统下使用C++语言在VS 2010平台上实现。选取用户聚类的大小为5，电影聚类的大小为20，在不使用任何加速的情况下离线训练的时间为3h左右。使用MPI(Multi Point Interface)加速后在一台主机上使用4进程并行计算，时间约为50min。在线推荐部分的时间大概为10ms，保证了推荐的实时性。

4.2 评价指标

将本文方法、基于好友拓扑结构相似度的方法、基于评分相似度的方法在好友推荐上进行比较。通过使用上述相似度的3种计算方法分别得到每个用户的Top-N推荐集，即向每个用户推荐最相似的 N 个潜在好友，使用准确率(Precision)、召回率(Recall)以及综合准确率和召回率的F1-measure 3个评价指标来验证方法的有效性。

$$Precision=\frac{(testing\cap top\text{-}N)}{|top\text{-}N|}$$

其中，*testing* 为测试集，*top-N* 为推荐出的好友集，即分子表示推荐出的好友中已经成为好友的数量，分母表示推荐出的好友总数量。

$$Recall=\frac{(testing\cap top\text{-}N)}{|testing|}$$

其中，分母表示测试数据集中好友的数量。

$$F1\text{-}measure=\frac{2\cdot Precision\cdot Recall}{Precision+Recall}$$

4.3 实验结果

表1－表3分别给出了3种相似度计算方法在准确率、召回率、F1-measure上的结果，其中 N 表示为每个用户推荐的潜在好友个数。其他3列分别表示3种用户相似度计算方法在好友推荐结果上的相应评价指标值。可以看出，本文提出的好友拓扑结构与基于评分的兴趣相似度相结合的计算方法在好友推荐中更为精准。

表1 3种相似度计算方法在准确率(Precision)上的对比结果

N	基于好友拓扑相似度	基于评分相似度	好友拓扑和评分结合
3	0.1617	0.1617	**0.2014**
6	0.1006	0.0874	**0.1222**
9	0.0792	0.0649	**0.0914**
12	0.0668	0.0520	**0.0792**
15	0.0580	0.0436	**0.0661**

表2 3种相似度计算方法在召回率(Recall)上的对比结果

N	基于好友拓扑相似度	基于评分相似度	好友拓扑和评分结合
3	0.1638	0.1638	**0.2040**
6	0.2040	0.1773	**0.2474**
9	0.2408	0.1973	**0.2776**
12	0.2709	0.2107	**0.3211**
15	0.2943	0.2207	**0.3345**

表3 3种相似度计算方法在F1-measure上的对比结果

N	基于好友拓扑相似度	基于评分相似度	好友拓扑和评分结合
3	0.1628	0.1628	**0.2027**
6	0.1348	0.1171	**0.1636**
9	0.1192	0.0977	**0.1375**
12	0.1072	0.0834	**0.1271**
15	0.0970	0.0728	**0.1103**

图4－图6分别给出了3种方法的准确率对比图、召回

率对比图以及 F1-measure 对比图。三角形标记表示基于好友拓扑相似度的方法，正方形标记表示基于评分相似度的方法，圆形表示本文方法，即好友拓扑与评分相结合的方法。图 4 纵坐标表示准确率的值，图 5 纵坐标表示召回率的值，图 6 纵坐标表示 F1-measure 的值。3 幅图的横坐标均表示推荐集 Top-N 的大小。从图中可以看出，本文方法很好地结合了好友拓扑结构及基于评分的用户兴趣，达到了不错的推荐效果。

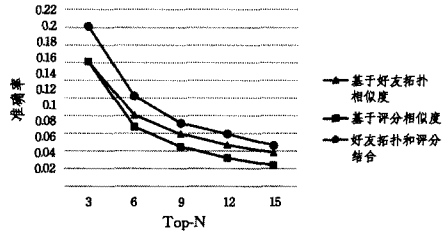

图 4　准确率对比图

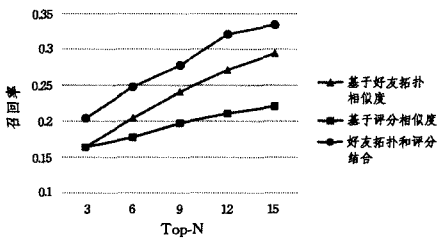

图 5　召回率对比图

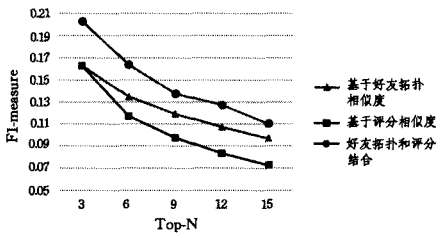

图 6　F1-measure 对比图

结束语　本文基于豆瓣网络的评分数据以及好友关系，研究了如何结合好友关系以及评分数据做出更精准的好友推荐。介绍了计算用户相似度的方法，包括用户间拓扑相似度的计算方法、评分相似度的计算方法以及最终相似度的计算。最后，通过实验证明了本文方法的有效性。

可以看到，本文方法虽然能够很好地结合用户拓扑结构与评分信息，但在拓扑结构相似度的计算上使用的是简单的余弦相似度，在今后的工作中，可以根据用户好友的数据对每个用户设置一个权重，以此提升拓扑相似度的可靠性。另外，如何将本文工作与可视化相结合也是非常有趣的研究，旨在为用户提供交互功能，让用户更能理解推荐好友结果的原因。

参 考 文 献

[1] Gou L, You F, Guo J, et al. Sfviz: interest-based friends exploration and recommendation in social networks[C]//Proceedings of the 2011 Visual Information Communication-International Symposium. ACM, 2011: 1-10

[2] Han X, Wang L, Crespi N, et al. Alike people, alike interests? Inferring interest similarity in online social networks[J]. Decision Support Systems, 2015, 69(1): 92-106

[3] Krulwich B. Lifestyle finder: intelligent user profiling using large-scale demographic data[J]. Artificial Intelligence Magazine, 1997, 18(2): 37-45

[4] Pazzani M, Billsus D. Learning and revising user profiles: the identification of interesting web sites[J]. Machine Learning, 1997, 27(3): 313-331

[5] Dahlen B J, Konstan J A, Herlocker J L, et al. Jump-starting movielens: User benefits of starting a collaborative filtering system with dead data[D]. University of Minnesota TR, 1998

[6] Linden G, Smith B, York J. Amazon. com Recommendations: Item-to-Item Collaborative Filtering[J]. IEEE Internet Computing, 2003, 7(1): 76-80

[7] Hofmann T. Latent semantic models for collaborative filtering [J]. ACM Transactions on Information Systems, 2004, 22(1): 89-115

[8] Yang X, Guo Y, Liu Y. Bayesian-Inference-Based Recommendation in Online Social Networks [J]. IEEE Transactions on Parallel and Distributed Systems, 2013, 24(4): 642-651

[9] Hofmann T. Probabilistic latent semantic analysis[C]// UAI' 99. Morgan Kaufmann, 1999: 289-296

[10] Ye M, Liu X, Lee W C. Exploring social influence for recommendation: a generative model approach[C] // Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2012: 671-680

[11] Tan P, Steinbach M, Kumar V, et al. Introduction to data mining [M]. Pearson Addison Wesley Boston, 2006

[12] Salton G, McGill M. Introduction to Modern Information Retrieval[M]. McGraw Hill, New York, USA, 1983

[13] Adamic L, Adar E. Friends and neighbors on the Web[J]. Social Networks, 2003, 25(3): 211-230

[14] McPherson M, Smith-Lovin L, et al. Birds of a Feather: Homophily in Social Networks[J]. Annual Review of Sociology, 2001, 27(1): 415-444

[15] Si L, Jin R. Flexible mixture model for collaborative filtering [C] // ICML. 2003, 3: 704-711

[16] Barabasi A, Albert R. Emergence of scaling in random networks [J]. Science, 1999, 286(5439): 509-512

[17] Zhou T, Lu L, Zhang Y C. Predicting missing links via local information[J]. European Physical Journal B, 2009, 71(4): 623-630

[18] Katz L. A new status index derived from sociometric analysis [J]. Psychometrika, 1953, 18(1): 39-43

[19] Jeh G, Widom J. SimRank: A measure of structural-context similarity[C]//Proc. of SIGKDD'02. 2002: 538-543

[20] Leicht E, Holme P, Newman M. Vertex similarity in networks [J]. Physical Review E, 2006, 73(2): 26120

[21] Zhao P, Han J, Sun Y. P-Rank: a comprehensive structural similarity measure over information networks[C]//Proc. of CIKM' 09. 2009: 553-562

[22] Gou L, Chen H, Kim J, et alSocial Network Document Ranking [C]//Proc. of JCDL'10. 2010: 313-322

[23] Gou L, Chen H, Kim J, et al.. SNDocRank: a Social Network-Based Video Search Ranking Framework[C] // Proc. of ACM MIR'10. 2010: 367-376

[24] Li C, Han J, He G, et al. Fast computation of simrank for static and dynamic information networks[C] // Proc. of EDBT' 10. 2010: 465-476

[25] Zeng W, Zeng A, Liu H, et al. Uncovering the information core in recommender systems[J]. Scientific Reports, 2014, 4: 6140