View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Xiamen University Institutional Repository

小 型 微 型 计 算 机 系 统
Journal of Chinese Computer Systems
2014 年 6 月 第 6 期
Vol. 35 No. 6 2014

异质社交网络中协同排序的好友推荐算法

陈 珂[1],邹 权[2],彭志平[1],柯文德[1]

[1](广东石油化工学院 计算机科学与技术系,广东 茂名 525000)
[2](厦门大学 信息科学与技术学院,福建 厦门 361005)
E-mail: chenke2001@163.com

摘 要: 随着社交网络的复杂化和异质化,传统推荐系统中协同过滤推荐方法由于推荐效果不佳而不能满足需求.本文通过扩展原有推荐方法中的因子模型提出了基于协同排序的好友推荐算法.相比于协同过滤,本文使用用户之间的偏序关系取代原始打分,以适合不易评分的异质信息网络,并且对于 Top-k 推荐只需考虑推荐序列,不需要精确预测低序列的评分的特点,避免不必要的计算,提高计算效率.相对于普通的因子模型,本方法在好友推荐中训练集更易构建,可以简单有效的融合多种有价值的内容相关特征.测试数据表明,基于协同排序的好友推荐与以往的矩阵分解方法相比较,在 Digg2009 好友关注关系数据集上测试,MAP 提高了 15.6% 左右.

关 键 词: 异质社交网络; 协同排序; 偏序关系; 好友推荐; 矩阵分解
中图分类号: TP391 **文献标识码:** A **文 章 编 号:** 1000-1220(2014)06-1270-05

Collaborative Ranking Friend Recommendation Algorithm in Heterogeneous Social Network

CHEN Ke[1], ZOU Quan[2], PENG Zhi-ping[1], KE Wen-de[1]
[1](Department of Computer Science and Technology, Guangdong University of Petrochemical Technology, Maoming 525000, China)
[2](School of Information Science and Technology, Xiamen University, Xiamen 361005, China)

Abstract: The social network is more and more complex and heterogeneous, traditional friend recommendation algorithm cannot cope the new situation for its ineffectiveness. This paper proposes a collaborative ranking based friend recommendation method by expending the traditional factor model. Comparing to the collaborative filtering friend recommendation methods, we replace the original scoring method with the partial order relation between the users to satisfy the needs of heterogeneous social network which is suitable for some situations that are hard to convert rates and it's of no need to precisely calculate the rates of low users in Top-k recommendation so that it is helpful to enhance the efficiency. According to the experiment results, our method is easy for friend recommendation to build training data and gets better results in learning user's interest than factorization model. The method can also mixes valuable content related feature easily. After testing in dataset of Digg2009, Collaborative ranking get 15.6% higher than Matrix factorization in MAP.

Key words: heterogeneous social networks; collaborative ranking; pair-wise relationship; friend recommendation; matrix factorization

1 引 言

随着社交网络的兴趣,如国外的 Facebook、Twitter、LinkedIn,国内的人人网、新浪微博等,越来越多的人们加入到社交网络中,使得社交网络越来越异质化.在异质的社交网络中,人与人之间的信息共享和协同交互是其主要的信息传播途径.所以社交网络致力于建立显性或者隐性的好友圈子[1].但好友推荐有别于传统的推荐,并没有具体的评分,故基于偏序关系的方法成为了目前的研究重点.

社交网络错综复杂,目前主要研究的场景包括同质的社交网络[2]以及异质的社交网络[3].异质的社交网络中蕴含大量不同类型的信息以及关系,挖掘有价值的内容成为一个难点.目前好友推荐主要包括基于内容的推荐,包括用户的 profile 的相似度[4]、用户发布的话题[5]、用户的位置信息[6]等.这些推荐主要是用户的显性的内容相似度,这些方法相对较好的挖掘了用户的兴趣.但对于这种推荐方式而言,可能会存在很多用户并没有填写基本信息甚至是伪造的虚拟的信息,这种情况下,这种方式出现了弊端.目前好友推荐中也比较流行的一种方式是基于链接预测的方法[7],这种方法主要考虑的用户与用户之间的链接关系,即关注关系或好友关系,把好友推荐问题转换成链接预测问题来考虑.但这种方法也存在很大的局限性,当前用户量众多,数据也比较稀疏,该方法对于那些与其他用户并无交集的用户并不适应.如何提出适应真实场景的推荐方式是非常关键的问题,也面临着大数据以及稀疏性的挑战.

传统的推荐系统主要存在的缺陷在于:一是目前没有一

收稿日期:2013-09-13 收修改稿日期:2013-11-04 基金项目:国家自然科学基金项目(61272382,61001013,61102136)资助;广东省科技计划项目(2012B010100037)资助;广东省高等学校科技创新项目(2013kjcx0132)资助. 作者简介:陈 珂,男,1964 年生,硕士,副教授,研究方向为数据挖掘、计算智能;邹 权,男,1982 年生,博士,讲师,研究方向为数据挖掘、机器学习;彭志平,男,1969 年生,博士,教授,研究方向为云计算、数据挖掘;柯文德,男,1976 年生,博士研究生,副教授,研究方向为机器学习、智能机器人.

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

种好友推荐很好的结合了矩阵分解的方法; 二是好友推荐没有很好的在异质社交网络中进行挖掘,也没有整合到算法中. 本文首先从基本的矩阵分解[8]的方式应用到好友推荐中,使用 MAP(Mean Average Precision)评价指标[9]来进行评测. 其次结合 Learning to Rank[10]的思想,提出了基于 Collaborative Ranking[11]的好友推荐方式. 相比于协同过滤,使用用户之间的偏序关系取代了以往的原始打分的逼近,尤其适合应用于不易评分的系统的应用场景中,并且对于 Top-k 推荐只需考虑推荐序列,不需要精确预测低序列的评分,从而避免了不必要的计算,提高了计算效率. 该方法相对于普通的因子模型在好友推荐中训练集更易构建,可以更加简单有效的融合多种有价值的内容相关特征. 本文剩余部分安排如下,第一节介绍基于协同排序的好友推荐算法的原理,具体介绍协同过滤算法以及协同排序算法; 第二节介绍该基于协同排序的好友推荐算法; 第三节通过实验验证本算法,并通过与其他好友推荐算法做比较,说明本算法的优势和不足; 第四节对相关工作进行总结后完成全文.

2　基于协同排序的好友推荐算法的原理

本节对于协同过滤[12]以及协同排序的原理做了介绍,并应用在数据集 Digg2009[13]中. 在好友推荐中,用户历史关注的用户,未来也有可能会关注类似的用户.

协同过滤针对网络数据对用户的偏好进行学习,不需要具体的用户的 profile 信息以及制定的领域知识,并且不需要对于内容进行细致的分析. 协同过滤包括基于邻居模型的方法以及基于因子模型的方法[14],本节会对该两种方法都进行实验.

2.1　符号定义

首先对于其中涉及的符号进行说明:

1) 使用 $U=\{u_1,u_2,\cdots,u_n\}$ 表示用户集合,包含 n 个用户,用 $I=\{i_1,i_2,\cdots,i_m\}$ 表示被推荐的好友集合,包含 m 个好友;

2) 使用 $|U|$ 代表用户的数量,$|I|$ 代表被推荐的好友的数量;

3) 对于一个用户 u 以及一个被推荐的好友 i,使用 r_{ui} 表示用户 u 对于一个好友 i 的打分. 当 $r_{ui}=1$ 表示关注了该用户,若 $r_{ui}=0$ 表示没有关注该用户.

4) p 为 User-Factor 矩阵,q 为 Item-Factor 矩阵;

5) $\tilde{r}_{ui}$ 表示根据矩阵分解后的结果对用户 u 与被推荐的好友 i 的预测值,而 r_{ui} 为真实值;

6) 由用户的关注关系形成的矩阵为 $R=r_{ui}$,$|U|*|I|$ 的,行表示用户,列表示即将被推荐给用户的好友;

7) e_{ui} 表示真实值与预测值的误差,即 $e_{ui}=r_{ui}-\tilde{r}_{ui}$;

8) 使用 Ω 表示一个记录(u,i,t) 在时间 t 时 u 对 i 的关注. 我们使用 Ω^+ 符表示所有的正例(即 $r_{ui}=1$ 的数据),使用 Ω^- 表示所有的负例(即 $r_{ui}=0$ 的数据).

2.2　协同过滤

协同过滤主要包括两种算法: 基于邻居模型的算法以及基于因子模型的算法. 在好友推荐中,使用这两种方法作为 baseline,与协同排序方法作比较.

2.2.1　基于邻居的算法在好友推荐中的应用

在一个图 G 中,预测未连接的 user-item 对 $\langle u,i\rangle$ 的可能性 $w(u,i)$. 首先对于一个节点 x,我们定义 $\Gamma(x)$ 为 x 在 N_h 的邻居集合,定义 $\tilde{\Gamma}(x)=\cap_{c\in\Gamma(x)}\Gamma(c)$ 为 x 邻居的邻居,作为基于链路预测的好友推荐算法之一的基于邻居的方法,主要包括四种方式[15].

共同的邻居: 在图中,x 和 y 两个节点的共同邻居 $\Gamma(x)\cap\Gamma(y)$ 可以代表两个节点间的相似度;

Jaccard's Coefficient: 使用的 x 与 y 共同的邻居与 x 或 y 的邻居的比值,具体的值为: $\left|\frac{\Gamma(x)\cap\Gamma(y)}{\Gamma(x)\cup\Gamma(y)}\right|$;

Adamic/Adar: 它计算的是两个对象之间的特征相似度,公式定义为:

$$\sum_{z\text{为}x,y\text{共同的}feature}\frac{1}{\log(frequency(z))} \tag{1}$$

在好友推荐中,对象是一个节点即一个用户,而其特征则是他们的邻居节点,故在好友推荐中的该相似度计算方式为:

$$\sum_{z\in\Gamma(x)\cap\Gamma(y)}\frac{1}{\log(|\Gamma(z)|)} \tag{2}$$

Preferential Attachment: 使用 $\Gamma(x)\times\Gamma(y)$ 来衡量未来的交互信息.

对于好友推荐而言,基于邻居模型的算法有所改变. 基于邻居模型的算法主要选择与其最相似的邻居,并且将其邻居喜欢的用户推荐给该用户. 在这里,我们主要使用了两种方法进行实验:

1) 基于 user-user 相似度的方法,首先计算每两个用户的相似度,并且将与该用户相似的用户中将其关注的用户推荐给该用户. 这里选择的计算相似度的方法是矫正余弦相似度方法[16]:

$$sim(u_1,u_2)=\frac{\sum_{i\in I}(r_{u1,i}-\bar{r}_{u1})(r_{u2,i}-\bar{r}_{u2})}{\sqrt{\sum_{i\in I}(r_{u1,i}-\bar{r}_{u1})^2}\sqrt{\sum_{i\in I}(r_{u2,i}-\bar{r}_{u2})^2}} \tag{3}$$

对于给定用户 u_0,选择 $sim(u_0,u_1)$ 最大的(即最临近的) k 个用户,记作 U_k,所有这些用户关注的其他好友记为 I_{U_k},再根据 k 个用户的共同关注来进行推荐:

$$\max_k(\tilde{r}_{u,i})=\max_{i\in I_{U_k}}\left(\sum_{u\in U_k}(r_{u,i}-\tilde{r}_i)(1-r_{u_0,i})\right) \tag{4}$$

说明: 上述公式中的$(1-r_{u_0,i})$该项表示被推荐的用户是否已经被用户关注,若已关注则计算得 0.

2) 基于 item-item 之间的相似度方法,虽然在这里 item 也为用户,但对于 item 之间相似度而言,主要的出发点是被关注的关系. 对于某个用户而言,根据其关注历史,将与该用户已经关注的用户的相似用户进行推荐. 计算相似度的方法同样使用基于矫正余弦相似度方法:

$$sim(i_1,i_2)=\frac{\sum_{u\in U}(r_{u,i_1}-\bar{r}_{i_1})(r_{u,i_2}-\bar{r}_{i_2})}{\sqrt{\sum_{u\in U}(r_{u,i_1}-\bar{r}_{i_1})^2}\sqrt{\sum_{u\in U}(r_{u,i_2}-\bar{r}_{i_2})^2}} \tag{5}$$

对于一个用户 u_0,该用户关注的所有用户集合记为 I_0,选择与 I_0 最相似的 k 个用户,并且 u_0 没有关注过的用户推荐给用户.

2.2.2　基于矩阵分解的算法

因子模型是将用户矩阵以及物品矩阵看作由一系列隐含因子组成,如何将原始的矩阵进行分解成两个因子矩阵则成为一个典型的矩阵分解问题:

$$R=U\times I \tag{6}$$

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

矩阵分解中奇异值分解(SVD)[17]是最常见的模型之一,当前很少有直接应用 SVD 来进行求解,由于 SVD 分解计算量复杂,并且要求矩阵是满秩矩阵,但在推荐中,我们的矩阵是很稀疏的,所以 SVD 分解很难直接应用到推荐算法中,故使用随机梯度下降进行矩阵分解,其直接用梯度下降的方法逼近原始矩阵,最终得到我们最终的模型矩阵. 一般情况,我们都会加上不同 user 和不同 item 的偏差估计,所以带有 user bias 和 item bias 的 baseline 预测器定义为:

$$\tilde{r}_{ui} = f(b_{ui} + q_i^T p_u) \tag{7}$$

在这里 $f(\cdot)$ 是一个将是实数值映射成为我们想要的值,在这里映射为 1 或 0. p_u 是一个 d 维的用户特征向量,q_i 是一个 item 特征向量. d 在这里是一个提前设置的参数,代表特征的数目. 在这里对于 bias 这一项 b_{ui} 定义为:

$$b_{ui} = \mu + b_u + b_i \tag{8}$$

在上述公式中,μ 表示的全局的平均打分值. b_u 与 b_i 分别为用户 u 以及 item i 的偏差. 所有的这些参数可以通过梯度下降进行求解,转换成一个正规的最小二乘求解问题,最优化下面的值:

$$\min_{p^*,q^*,b^*} \sum_{(u,i)\in K} (r_{ui} - f(b_{ui} + q_i^T p_u))^2 + \lambda_1 |p_u|^2 + \lambda_2 |q_i|^2 + \lambda_3 b_u^2 + \lambda_4 b_i^2 \tag{9}$$

根据上述公式,我们最终根据逼近原始矩阵寻找最合适的参数 b_u, b_u, p_u, q_i,参数 $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ 是为了防止过拟合而引入公式中的.

随机梯度下降算法[18]是一个解决优化问题的非常有效的方法,它随机在集合 Ω 里面进行循环,并且对相关的参数每次选择一小步进行梯度下降,沿着最小化误差的方法. 具体的 SGD 参数更新方法根据下面的几个方程:

$$\begin{aligned} b_u &\leftarrow b_u + \eta(e_{ui} - \lambda_1 b_u) \\ b_i &\leftarrow b_i + \eta(e_{ui} - \lambda_2 b_i) \\ q_i &\leftarrow q_i + \eta(e_{ui} p_u - \lambda_3 q_i) \\ p_u &\leftarrow p_u + \eta(e_{ui} q_i - \lambda_4 p_u) \end{aligned} \tag{10}$$

在上述的公式中,η 是学习速率,在迭代过程中,每一轮的复杂度都是线性的.

在真实环境中,用户的反馈以及一些基于内容 feature 的融入,也是很重要的信息,并且在实际系统中,这些特征的加入也明显提高了推荐的准确率. 改进的公式如下:

$$\tilde{r}_{ui} = f(b_{ui} + q_i^T(p_u + \sum_{k\in F(u)} \beta_k y_k)) \tag{11}$$

在上述公式中,$f(u)$ 表示了 user 反馈的一些记录. 这种方法是一种 SVD ++的方法. 在推荐问题中,更适合实际的问题及场景.

2.3 协同排序

协同排序是将排序学习中的 pair-wise 的思路借鉴到推荐中,协同用户进行排序,也可以看作一个 pair-wise 的协同过滤方式.

排序学习是网页搜索中结果排序的一种算法,它使用机器学习的方法来解决排序问题,Learning to Rank 是一种有监督的学习问题,需要一个标记好的训练数据集 D. 训练集 D 是由一系列 query-impression(q_i, l_i) 对以及标记的对应的得分 y_i 组成.

$$D = \{(q_i, l_i, y_i) : i \in [1, \cdots, n]\} \tag{12}$$

相关度标记 y_i 一般是分等级的,比如 $y=1$ 太相关,$y=5$ 非常相关等. 对于 query-impression 对,是由固定长度的特征向量表示 $x_{ql} \in R^d$,特征向量是根据不同的场景进行的定义和选择,比如锚文本、URL 的长度、网页的 pagerank 值等. 学习涉及模型参数的估计,并且最终得到最佳模型,并且该模型可以输入特征向量,并能够给出相关度得分. Learning to Rank 的模型目前主要分为三类: point-wise, pair-wise 以及 list-wise. 这三类之间的主要区别在于损失函数不同,并且训练数据也存在差异.

Point-wise 模型输入一个 query-impression 对,并输出最终的预测得分,为了训练最终的参数方程,若相关度得分是连续的则可以使用一个回归损失,若是离散的则可以使用分类损失. 一般情况下,point-wise 模型主要是回归模型[19]以及分类模型,这种 point-wise 模型的优点在于学习模型相对直接,并且算法相对简单,对于大规模的数据也有很好的性能. Pair-wise 同 point-wise 一样,也是最终预测一个函数,给定输入,输出最终的预测得分值. 然而,对于 pair-wise 模型来说训练集的损失是基于对于同一个给定的 query 所有的 impression 对,即$\{(q_i, x_{q_i l_i}, y_i), (q_j, x_{q_j l_j}, y_j)\}$,在这里 $q_i = q_j$,但 $l_i \neq l_j$. 针对 pair-wise 的损失是依赖于相关度标记的顺序,若预测的序列是正确的,则损失低. 目前常见的 pair-wise 的方法包括 RankBoost、RankNet 以及 LambdaRank 方法等.

协同排序(Collaborative Ranking)则借鉴了 Learning to Rank 中的 pair-wise 方法,将两个带有序列的打分对作为损失代替之前的真实值与预测值的差值. 这样协同过滤方法由之前预测最终的打分转变成了学习两个 item 的 rank,这样就可以得到所有 item 的偏序序列. 则训练集 D 记为:

$$D = \{\langle u, i, h\rangle \mid i \in 1 \in F(u), h \notin F(u)\} \tag{13}$$

$F(u)$ 代表用户 h 喜欢的 item,而 h 则为负例,用户 h 不喜欢或者没有关注的 item. 由于一般情况下,负例是非常庞大的,我们可以通过采样技术选择一定比例的负例进行训练,这样再使用 pair-wise 的优化目标方法对模型进行学习. 则目标函数变为:

$$\min \sum_{\langle u,i,h\rangle \in D} \ln(1 + e^{-(y_{u,i} - y_{u,h})}) + regularization \tag{14}$$

将正例以及负例形成一个 pair 进行学习,后面的正规项根据不同的场景有所变化. 最终通过梯度下降学习模型中的参数.

2.4 基于协同排序的好友推荐

在上节中对于好友推荐中的协同过滤方式进行了详细的说明,并且介绍了协同排序(Collaborative Ranking)的基本原理和排序学习(Learning to Rank)的思想. 对于好友推荐而言,用户对其他用户的打分一般包括两个级别 1 或 0,1 表示用户关注了该用户,而 0 则表示用户并没有关注该用户,所以可以将关注不关注的用户形成一系列 pair 作为训练集,将该问题看作一个 pair-wise 的排序问题. 对于这种 pair-wise 的训练而言,给定一个训练集中的负例(u,i)以及一个正例(u,j),最终对于 j 的得分要优于 i 的打分才是最终的目标,所以目标的损失函数定义为:

$$g(\tilde{r}_{ui}, \tilde{r}_{uj}) = (1 + \exp(-(b_j - b_i + (q_j - q_i)^T p_u)))^{-1} \tag{15}$$

对于每一个负例 $r_{ui} \in \Omega^-$,我们随机选择一个正例 $r_{uj} \in$

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

Ω^+ 来建立训练对 (r_{ui}, r_{uj}) 并用随机梯度下降算法进行优化求解,在这里省略了 user 的正规化项,是由于对于 i 和 j 来说 user 的偏差是一致的

3　实验与数据分析

3.1　数据集

本文选择了 Digg 2009 数据集来进行实验, Digg 2009 统计了 2009 这段时间中,用户之间的好友关系,其中主要包括两种关系: 0 代表单向关注关系,1 表示双向关注关系即互相关注. 数据集中每一行表示一个关注关系具体包括:

Mutual: 表示该链接是否为双向链接,即该两个用户是否互相关注. 若为 1 则表示是双向链接,若为 0,则不是;

Friend_data: 该列是一个 Unix 时间戳,表示建立该好友关系的时间.

Userid: 关注的用户,对应唯一的 id.

Friendid: 被关注的用户,也对应唯一的 id.

3.2　评价标准

由于不同的推荐侧重点是不同的,所以针对不同的推荐选择不同的评价体系,辩证的看待评价指标带来的结果. 传统的推荐主要用到的评价指标是 MSE(Mean Square Errors) 以及 RMSE(Root Mean Square Errors) 来进行评估, MSE 以及 RMSE 一般针对的是具体的评分,但对于没有具体评分的推荐问题,还存在一定的缺陷. 对于好友推荐而言,这种推荐方法并不是我们所预期选择的.

针对上述中 MSE 以及 RMSE 评估方式的说明,这种方式并不适合好友推荐. 针对需求,MAP(Mean Average Precisions) 推荐方法作为好友推荐中的评估方式更加适合. 在 Top-k 好友推荐中,k 的选择也会对结果有一定的影响. 给出用户最可能关注的 k 个用户,并且对于序列命中率进行预估, MAP 是全部准确率的平均值. 对于 Top-k 推荐,对于一个用户,平均的准确率为:

$$ap@k = \sum_{i=1}^{k} p(i) / (\text{用户点击的 } item \text{ 数}) \qquad (16)$$

为确保推荐的准确性,在测试中隐藏每个用户的一个好友数据,并推荐预测偏好最高的 N 项偏好信息,计算预测的精度比例. 如果隐藏的好友实际上包含在 Top-N 推荐名单中.

3.3　实验结果与分析

前面几节中详细描述了三种好友推荐的具体实现方法及原理: 基于邻居模型的好友推荐、基于矩阵分解的好友推荐以及基于 Collaborative Ranking 的好友推荐,并且将这些方法实现用于数据集 Digg 2009 中的好友关系中,对结果进行对比分析.

3.3.1　数据预处理

对 Digg 2009 中的数据根据好友之间的关注关系,形成三个集合训练集、测试集还有验证集. 数据集分为 80% 的训练集和 20% 的测试集,平均分为 5 种不同的测试集(5 倍交叉验证) 用来测试. 首先将所有的数据形成正例,但是在矩阵中需要负例的存在,故对每一个用户随机选取其未关注的好友当作负例. 最终每个集合中形成了一个两层 map,对于所有的用户而言,使用 M(key,value) 来表示,key 为用户 id,而 value 则是用户对应的评分 map 值,value 表示为 $\{(u_1, v_1), (u_2, v_2), \cdots (u_n, v_n)\}$,并且所有的 value 为 1 的用户需要按照关注时间来进行排序,对于每个用户的关注序列而言,其关注时间的先后默认为其关注序列,以便于计算 MAP 值.

3.3.2　结果对比及分析

经过计算,该数据集在 4 种方法下,其推荐 5 个用户的 RMSE 值和 MAP 值如表 1 所示,其中方法 1 指的是基于 user 的关注关系的邻居模型,方法 2 指的是基于 item 的被关注关系的邻居模型,方法 3 指基本的矩阵分解算法,方法 4 指的是本文中提出的基于 pair-wise 的排序学习的模型.

表 1　实验结果比较

Table 1　Experimental results comparing

描述	RMSE	ap@5
方法 1	0.8412	0.4123
方法 2	0.8317	0.4017
方法 3	0.9016	0.5471
方法 4	X	0.6337

通过上述实验结果,我们可以看出基于矩阵分解的算法要远远优于邻居算法模型,并且基于排序学习的方法的 MAP 值在推荐五个用户时,相比矩阵分解的方法而言提高了 15.6%. 并且通过实验发现,pair 对的构建更适用于好友推荐,由于好友推荐很难将其转换成一个评分问题,仅仅用 0、1 表示其关注未关注关系并不准确,虽然一个用户未关注另外一个用户,但并不表示他对该用户没有兴趣. 所以只能说明其中的偏序关系,尤其是推荐了多个好友后,用户选择性的进行关注,这种用户行为比较明显的体现了一种偏序关系,该解决方法适用于好友推荐的场景,并且对于训练集的构建、用户行为的反馈信息都能很好的处理.

现在考察这四种方法的准确率以及召回率,结果见图 1.

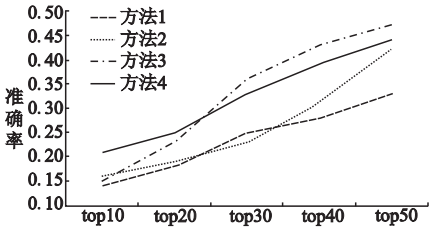

图 1　四种方法准确率对比

Fig. 1　Precision contrast of four methods

与基于 user 的关注关系的邻居模型(方法 1) 和基于 item 的被关注关系的邻居模型(方法 2) 相比,基于 pair-wise 的排序学习的模型(方法 4) 推荐准确率在 $10 \leqslant N \leqslant 50$ 范围内准确性要更高. 当 $N \leqslant 20$ 时,基于 pair-wise 的排序学习的模型推荐效果(方法 4) 的推荐准确率要高于基本的矩阵分解算法(方法 3) 的好友推荐效果,但当 $N > 20$, 基于 pair-wise 的排序学习的模型(方法 4) 推荐准确率低于基本的矩阵分解算法(方法 3) 的推荐准确率.

下面考察四种方法推荐的召回率. 在下页图 2 中,可以看出在 $10 \leqslant N \leqslant 50$ 范围内,基于 pair-wise 的排序学习的模型(方法 4) 的召回率要高于基于 user 的关注关系的邻居模型(方法 1) 和基于 item 的被关注关系的邻居模型(方法 2) 的召回率; 在 $N \leqslant 30$ 时,基于 pair-wise 的排序学习的模型(方法

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

4) 的召回率要低于基本的矩阵分解算法(方法 3) 的召回率，当 N > 30 时，基于 pair-wise 的排序学习的模型(方法 4) 的召回率高于基本的矩阵分解算法(方法 3) 的召回率.

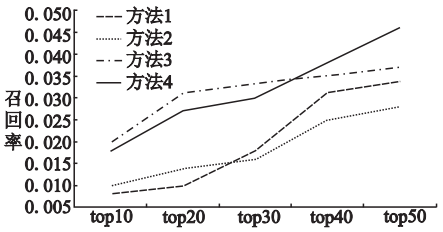

图 2 四种方法召回率对比

Fig. 2 Recall contrast of four methods

4 总 结

本文主要介绍了好友推荐中基于好友关系的算法，主要包括了基于邻居模型的算法、基于矩阵分解的算法，并且从排序学习的角度出发提出了一种新的 Collaborative Ranking 的方法应用在好友推荐中. 并且提出了一种基于正反例构建 pair 对，并对偏序序列进行估计. 在 Digg2009 数据集得出了以下的结论:

1) 基于矩阵分解的好友推荐的准确率要比基于邻居模型的准确率更高，更易挖掘潜在用户.

2) 基于协同过滤的方式在好友推荐中 MAP 值在推荐 5 个用户的时候，相比矩阵分解提高了 15.6%.

3) 基于协同过滤的方式，pair 对更易构建，相比评分方式更适合好友推荐，语义更贴近.

通过对比四种方法的准确率和召回率，可以看出，本文提出的方法要明显好于基于 user 的关注关系的邻居模型和基于 item 的被关注关系的邻居模型. 但是相对与基本的矩阵分解模型，在推荐好友数量较少时，基于 pair-wise 的排序学习的模型的准确性更高，但是召回率相对较低，当推荐好友的数量更多时，准确性下降，召回率上升. 综合上述两种评价标准，可以看出本文提出的方法相对于其他三种基准方法，有更好的好友推荐效果.

References:

[1] Zhang Zhong-feng, Li Qiu-dan. Latent friend recommendation in social network services [J]. Journal of the China Society for Scientific and Technical Information, 2011, 30(12): 1319-1325.

[2] Matthijs Kalmijn, Jeroen K Vermunt. Homogeneity of social networks by age and marital status: a multilevel analysis of ego-centered networks [C]. In: Chicago Proceedings of the Social Sciences Section of the Netherlands Society for Statistics and Operations Research (NVVS), 2004: 25-43.

[3] Deng Cai, Zheng Shao, He Xiao-fei, et al. Mining hidden community in heterogeneous social networks [C]. In: Chicago Preceeding of LinkKDD'05, 2005: 1-26.

[4] He Chao-bo, Tang Yong, Chen Guo-hua, et al. Potential friend recommendation method for large-scale social network [J]. Journal of Hefei University of Technology(Natural Science), 2013, 36(4): 420-424.

[5] Yu Hai-qun, Liu Wan-jun, Qiu Yun-fei. Second-level contacts recommendation of social network service based on subjects of users' preference [J]. Journal of Computer Applications, 2012, 32(5): 1366-1370.

[6] Zhu Rong-xin. Research on the model of latent user and location recommendation in location-based social networks [D]. Nanjing: Nanjing University of Posts and Telecommunications, 2013.

[7] Yang Zi. Predictive models in social network analysis [D]. Beijing: Tsinghua University, 2011.

[8] Wang Hai-lei, Mu Yan-chao, Yu Xue-ning. Resource recommendation in social tagging system based oncollaborative matrix factorization [J]. Application Research of Computers, 2013, 30(6): 1739-1741.

[9] Gordon V Cormack, Thomas R Lynam. Statistical precision of information retrieval evaluation [C]. In: Seattle. SIGIR'06, Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2006: 533-540.

[10] Liu Tie-yan. Learning to rank for information retrieval [J]. Foundations and Trends in Information Retrieval, 2009, 3(3): 225-331.

[11] Zheng Jia-jia. Friends recommendation based on graph ranking on social network site [D]. Hangzhou: Zhejiang University, 2011.

[12] Zhang Guang-wei, Li De-yi, Li Peng, et al. A collaborative filtering recommendation algorithm based on cloud model [J]. Journal of Software, 2007, 18(10): 2403-2411.

[13] Digg 2009 data set [EB/OL]. http://www.isi.edu/integration/people/lerman/load.html? src = http://www.isi.edu/lerman/downloads/digg2009.html, 2009.

[14] Wu Jin-long. Collaborative filtering algorithm in netflix prize [D]. Peking University, 2010.

[15] Zan Huang, Xin Li, Hsinchun Chen. Link prediction approach to collabora-tive filtering [C]. In Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries, 2005: 141-142.

[16] Wang Li-cai, Meng Xiang-wu, Zhang Yu-jie. Context-aware recommender systems [J]. Journal of Software, 2010, 23(1): 1-20.

[17] Zhu Min, Su Bo. Algorithm research of collaborative filter recommending base on singular value decomposition [J]. Network and Computer Security, 2010, 1(7): 20-21.

[18] Rainer Gemulla, Erik Nijkamp, Peter Haas, et al. Large-scale matrix factorization with distributed stochastic gradient descent [C]. The 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2011: 69-77.

[19] Xu Hai-ling, Wu Xiao, Li Xiao-dong, et al. Comparison study of internet recommendation system [J]. Journal of Software, 2008, 20(2): 350-362.

附中文参考文献:

[1] 张中峰，李秋丹. 社交网站中潜在好友推荐模型研究 [J]. 情报学报，2011，30(12)：1319-1325.

[4] 贺超波，汤 庸，陈国华，等. 面向大规模社交网络的潜在好友推荐方法 [J]. 合肥工业大学学报(自然科学版)，2013，36(4)：420-424.

[5] 于海群，刘万军，邱云飞. 基于用户话题偏好的社会网络二级人脉推荐 [J]. 计算机应用，2012，32(5)：1366-1370

[6] 朱荣鑫. 基于地理位置的社交网络潜在用户和位置推荐模型研究 [D]. 南京：南京邮电大学，2013.

[7] 杨 子. 社会网络分析中的预测模型 [D]. 北京：清华大学，2011.

[8] 王海雷，牟雁超，俞学宁. 基于协同矩阵分解的社会化标签系统的资源推荐 [J]. 计算机应用研究，2013，30(6)：1739-1741.

[11] 郑佳佳. 社交网络中基于图排序的好友推荐机制研究与实现 [D]. 杭州：浙江大学，2011.

[12] 张光卫，李德毅，李 鹏，等. 基于云模型的协同过滤推荐算法 [J]. 软件学报，2007，18(10)：2403-2411.

[14] 吴金龙. Netflix Prize 中的协同过滤算法 [D]. 北京：北京大学，2010.

[16] 王立才，孟祥武，张玉洁. 上下文感知推荐系统 [J]. 软件学报，2010，23(1)：1-20.

[17] 朱 敏，苏 博. 基于奇异值分解的协同过滤推荐算法研究 [J]. 计算机安全，2010，1(7)：20-21.

[19] 许海玲，吴 潇，李晓东，等. 互联网推荐系统比较研究 [J]. 软件学报，2008，20(2)：350-362.

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net