一种基于社区分类的社交网络用户推荐方法

赵 勤[1] 王 成[1] 王鹏伟[2]
(同济大学嵌入式系统与服务计算教育部重点实验室 上海 201804)[1]
(东华大学计算机科学与技术学院 上海 200051)[2]

摘 要 社交网络上的用户推荐是目前计算机领域研究的热门问题。已有的社交网络推荐算法对于多主题的社交网络下的相关用户的推荐效果不佳。针对此问题,对社交网络的主题分类方法进行了研究与讨论,在此基础上提出了基于主题的用户社区分类方法,并根据分类信息给出一种新的社交网络用户推荐方法。经实验验证,该方法能有效地提高推荐的准确性并降低时间复杂度。

关键词 社交网络,信息推荐,信息检索
中图法分类号 TP311 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.5.036

Novel Method on Community-based User Recommendation on Social Network

ZHAO Qin[1] WANG Cheng[1] WANG Peng-wei[2]
(The Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 201804, China)[1]
(School of Computer Science and Technology, Donghua University, Shanghai 200051, China)[2]

Abstract The user recommendation on social network is a hot spot of research. The existing work's effort is not good in multiple-topics social networks. For solving this problem, we analyzed the topic classification of social network, and proposed a novel method of recommendation on relevant users. The experimental result shows that this method can improve the accuracy of recommendation and decrease the time complexity efficiently.

Keywords Social networks, Information recommendation, Information retrieval

1 引言

近几年,社交网络在世界范围内迅速发展,像 Facebook、Twitter 之类的互联网社交网络服务在全世界被普遍使用。在国内,也有像新浪微博这样的社交网络服务被广泛使用。这种基于人际关系的信息传播方式的传播速度要比传统媒体更快,很多新闻都是第一时间通过社会网络传播到世界各地。因为用户在转发信息时会根据自身情况过滤掉无关的信息,所以这种基于人际关系的信息推送方式的准确性也要明显优于传统媒体。

然而,同传统媒体一样,用户不可能关注到自己可能感兴趣的所有信息发布者,很多有用的信息可能因为社交网络上的信息总量太大而被湮没在大量的无关信息中。因此,针对社交网络的用户推荐成为了目前研究的热点问题。

目前,主流的社交网络的用户推荐算法主要分为3类。

第一类是基于内容的推荐算法,这一类算法主要比较不同用户所发表内容的相似度,将与用户发表过或转发过的内容相似的信息推荐给用户。此类的代表性算法为余弦相似度[1]算法,其通过计算不同文档之间关键词的词频-逆文档频率值[2](Term Frequency-Inversed Document Frequency, TF-IDF)的余弦夹角来判断两个文档的相似程度。这类方法在社交网络的用户推荐中存在一些问题。在实际中,用户希望获取的很可能不是相似的内容,而是相关主题的信息。比如一个用户转发了一条关于苹果产品的微博,他可能希望系统推荐给他一些其它的与苹果公司相关的内容,而不是与他转发内容相似的信息。而现有的基于内容的推荐算法不能很好地发现主题相同但内容不相似的信息,具有相似内容的信息往往被优先推荐给用户。

第二类是协同过滤算法[3]。它是一种通过对大量拥有较多信息的用户进行数据挖掘,进而对用户信息中所缺失的部分进行预测和补充的方法。这种算法被广泛应用于电子商务中,如亚马逊、易趣和淘宝等在线商城等,用于对用户的商品推荐。在社交网络的推荐中,这种算法也经常被使用,例如一些聚焦于影视评论的社交网络会根据那些与用户在部分电影观影体验上相似的人对其它电影给出的评价,来向用户推荐其中其他用户评分较高的电影。在单一领域的社交网络中,这是一种被证明为行之有效的方法。但是,大部分用户使用的都是综合性的社交网络,如 Facebook、Twitter 之类的大型社交网络,用户都可以同时关注很多不同类别的信息发布者。用户在某个方面上兴趣爱好相似不能代表他们在另一个方面上也是相似的。另一方面,信息发布者也可能有多种身份,比如,比尔·盖茨是微软公司的前 CEO,同时他也是一名著名

到稿日期:2016-01-24 返修日期:2016-03-20 本文受国家自然科学基金重大研究计划(91218301)资助。
赵 勤(1982—),男,博士生,主要研究方向为数据挖掘、大数据处理,E-mail:2qzhao@tongji.edu.cn;王 成(1980—),男,博士,教授,主要研究方向为无线传感网、知识发现;王鹏伟(1984—),男,博士,讲师,主要研究方向为 Web 服务计算、云计算等。

的慈善家，所以关注他的人的目的未必相同，他们所希望接收到的信息也是不同的。也就是说，协同过滤不能很好地解决在主题混杂的情况下的信息推荐问题。

第三类是基于社交关系的信息推荐。这一类方法主要是关注用户之间的相互关注关系，其中最常用的是 FOAF 算法(Friend of a Friend)[4]。它的一个基本假设就是朋友的朋友很可能也是自己的朋友，共同的朋友越多，越可能是自己的朋友。这种方法在基于现实生活的社交网络圈内是比较准确的，但是在虚拟社会里，这一假设则未必成立。而且，随着社交圈的扩大，用户所关注的发布者与自己的关系越来越远，由于这种方法的推荐方法没有考虑到区分用户关心的主题，单纯的基于关注关系的推荐很可能出现很多自己并不关心的主题。文献[12]提出了一种社交圈的生成方法，即计算用户关注关系的相似程度，通过聚类算法将相似的用户聚入同一个社交圈内。这种方法同样只考虑了用户的关注关系，忽略了用户发布的信息对其兴趣的影响。

为克服这 3 类信息推荐算法的不足之处，一些混合的推荐算法被提出。Chen Ji-lin 等人提出了 Content-plus-link 算法[4]用于推荐相似的用户，这种算法在考虑内容相似性的同时，还考虑了内容的发布者与用户的相互关系，如果两者之间存在实际的社交链接关系，则将发布者推荐给用户的优先级会被提升。它的具体做法是：首先从用户发布的内容中提取出常见的关键词构成表示用户兴趣的关键词向量，然后用每个词的 TF-IDF 值描述它对用户的重要度。系统通过计算两个用户的关键词向量的余弦相似度来确定他们之间的相似性，如果两个用户除了内容上相似以外，还存在实际的社交链接，则将他们的相似度提高 50%。由于同时考虑了社交关系在用户推荐中的作用，这种推荐方法的准确性要比传统的单一方法高。但是，它也存在着一些问题：第一，由于它计算相似度的方法仍主要是基于内容，因此它同样无法找到与用户兴趣相关但不相似的内容，它推荐给用户的信息发布者仍是那些拥有相同关键词的用户。第二，由于用户的兴趣并非只有一类，而多个主题的存在无疑将降低在某一个主题上爱好非常相似但在其它主题上共同点较少的用户之间的相似度。根据这个算法，在多个主题上拥有较少相似度的用户往往比只在某一主题上有较高相似度的用户更容易得到推荐。而用户往往只关心推荐给他的用户是否在某个兴趣上相似，并不需要两者在每个主题上都有共同话题。第三，它忽略了主题在描述链接类型上的作用。这种方法只考虑两者之间是否存在链接，而未考虑这条链接是否是因为用户之间共同的兴趣造成的。举例而言，两个用户共同关注了一个好友，前者是围棋和 IT 爱好者，后者则是足球和数码爱好者。这个共同的好友是前者的棋友，而对后者则是球友。在此例中，显然这个共同的关注关系并不应该加强两个用户之间的相似性，因为这种关注关系并不能反映出两个用户的共同爱好。文献[13]提出一种结合全局与双重局部信息的社交推荐方法，这种方法将用户的声誉值作为用户评分值的权重标准，并将社交关系分成明确关系和隐含关系，综合以上两点提出基于声誉的局部关系细分推荐算法。文献[14]提出一种动态调整用户标签权重的推荐方法，这种方法将时间作为用户评分的权重，越新的标签所占权重越大。与文献[4]相同，这些方法同样没有考虑到用户之间关注关系的形成原因，只能适用于单一领域的社交网络推荐。

为了解决现有方法存在的以上问题，文中提出了一种基于主题的社交网络信息发现与推荐方法。这种方法的主要思想是：将社交网络划分成多个不同主题的子网络，每个子网络相当于一个社区，专注于某一主题，而每个用户都可以处在多个不同的社区中。在单一主题的社区里，用户之间的相似度无疑比在多个主题的情况下更能描述两者之间的相似关系；而且，在单一主题的社区里，链接关系是由于共同兴趣产生的概率也比在多个主题的情况下有大大提升；此外，不同于传统的推荐方法，首先，同时考虑了用户的内容相似度与关注相似度，以发现一些缺少链接关系的用户。首先，可以根据某个社区中的用户信息，如相互关注关系、用户所发布的内容等，对用户缺失的信息进行预测；然后，综合多个社区中用户的信息进行合理的信息推荐。

本文第 2 节对背景知识进行了介绍；第 3 节叙述了基于主题的社交网络用户分类方法，通过现有的算法对社交网络中的用户进行主题划分，将用户分入不同的社区中；第 4 节描述了基于社区的用户推荐算法，对其思想进行了描述，并给出了具体的公式和算法；第 5 节通过实验评估了所提方法的有效性和实用性；最后是对本文内容的总结。

2 背景知识

2.1 模糊 C 均值聚类算法

模糊 C 均值聚类算法[5]，即 FCM 聚类算法，是对传统的 K-Means 聚类算法的一种改进。1973 年，Bezdek 在 K-Means 聚类算法的基础上提出了 FCM 算法来实现模糊聚类。与 K-Means 聚类算法不同，在 FCM 聚类算法中，每个点不是仅分入唯一的类中。FCM 聚类算法采用了模糊集的概念，每个点对于一个簇的隶属度是属于[0,1]的一个区间。在聚类过程中，每个点对所有簇的隶属度的总和不一定为 1，所以需要对每个点的归属度进行归一化处理，使得每个点对于各个簇的隶属度总和为 1。

与 K-Means 类似，FCM 聚类算法的主要思想是将 n 个点分成 c 个模糊组，对每一个组分别求聚类中心，使得类之间的非相似性的价值函数最小。FCM 聚类算法的价值函数公式如下所示：

$$J(U,c_1,c_2,\cdots,c_c)=\sum_{i=1}^{c}\sum_{j}^{n}u_{ij}^{m}d_{ij}^{2} \tag{1}$$

其中，U 是大小为 $c\times n$ 的所有点的隶属矩阵；c_i 是第 i 个类的聚类中心；u_{ij} 是第 j 个数据点相对于第 i 个类的隶属度，其取值范围为[0,1]，m 为大于 1 的指定加权参数；d_{ij} 是第 j 个数据点与第 i 个类的聚类中心的非相似性指标值，这个指标在计算中通常使用两个点之间的欧氏距离，本文中因为描述的是关键词共同的主题关系，所以采用了语义相关度指标来刻画两个点之间的非相似性指标。

由价值函数可以推出使其最小的聚类中心和隶属度计算公式：

$$c_i=\frac{\sum_{j=1}^{n}u_{ij}^{m}x_j}{\sum_{j=1}^{n}u_{ij}} \tag{2}$$

$$u_{ij}=\frac{1}{\sum_{k=1}^{c}\left(\frac{d_{ij}}{d_{kj}}\right)^{\frac{2}{m-1}}} \tag{3}$$

其中，x_j 表示第 j 个点。

根据以上两个公式，FCM 聚类算法使用迭代来计算聚类结果。其具体步骤如下所示：

(1)用随机数值初始化点的隶属矩阵 U，并将其归一化；

(2)使用式(2)计算各个类的聚类中心 c_i；

(3)根据式(1)计算聚类结果的价值函数，如果价值函数的值小于一个给定的阈值或者较上次迭代的结果的变化量小于一个阈值，则停止迭代，将当前结果作为最终结果输出；

(4)使用式(3)计算新的隶属矩阵 U，然后回到步骤(2)中开始新的一轮的迭代计算。

通过以上步骤，就可以得到所需要的聚类结果。

2.2 协同过滤算法

协同过滤[3](Collaborative Filtering)的主要思想是利用已有的用户对信息的偏好数据来对其它信息的偏好进行预测。它是基于以下的推论：如果两个用户对相同内容的评价大体上相近，则其中一个用户对他未给出评价的内容的观点可能与另一用户对此内容的评价相似。这一推荐算法的优点在于可以通过已有数据对用户的喜好做出合理的预测，其缺点在于对缺少初始数据的新用户来说帮助不大。

协同过滤算法从相似的类型上主要细分为两种，一种是基于用户的协同过滤，一种是基于物品的协同过滤。下面分别对两种类型的协同过滤算法进行介绍。

2.2.1 基于用户的协同过滤算法

基于用户的协同过滤算法的思想是比较目标用户与其它用户的相似度，并根据与其相似的其它用户对物品的喜好或者评分来对目标用户进行推荐。对于现有的用户 u，一件物品 i 相对于他的推荐度计算公式如下所示：

$$g(u,i)=C\sum_{v\in U}sim(u,v)\times rat(v,i) \tag{4}$$

其中，C 是正规化参数，$sim(u,v)$ 表示用户 u 与用户 v 之间的相似度，v 为与用户 u 相似的用户集合 U 中的任一用户，$rat(v,i)$ 为用户 v 对物品 i 的评分。

从物理意义上来说，式(4)表示一件物品 i 对用户 u 的推荐度由与其相似的用户对物品 i 的平均评分值决定。

2.2.2 基于物品的协同过滤算法

与基于用户的协同过滤算法相似，基于物品的协同过滤算法也是通过计算相似度的方法来进行推荐。与前者不同的是，基于物品的协同过滤算法使用的是物品之间的相似度，它根据用户的已有评分记录来对与这些已评分的物品相似的其它物品进行推荐。

类似于基于用户的协同过滤算法，基于物品的协同过滤算法的推荐度计算公式为：

$$g(u,i)=C\sum_{j\in I}sim(i,j)\times rat(u,j) \tag{5}$$

其中，C 是正规化参数，$sim(i,j)$ 表示物品 i 与物品 j 之间的相似度，j 为与物品 i 相似的物品集合 I 中的任一物品，$rat(u,j)$ 为用户 u 对物品 j 的评分。

3 基于主题的用户社区分类

传统的用户推荐方法只考虑了用户之间的关注关系，却未考虑关注关系的形成原因。如前文所述，由于用户的多角色属性，多个用户关注同一个用户的原因却可能是不同的。为了避免多主题混杂环境下信息推荐受到的干扰，本文提出了一种基于社区分类的社交网络信息推荐方法。我们认为，用户之间关注关系的产生很有可能是由于他们所发表的拥有共同主题的信息所造成的。可以认为，用户之间共同的主题是对他们之间关注关系的语义描述。为了找到用户之间共同的主题，首先需要对社交网络进行社区分类，把用户按照关注的主题分入多个不同的社区之中，其中，每个用户都可能属于多个不同的社区。在单个社区内，用户之间的关注关系更有可能是由该社区的主题所造成的。

在社交网络中，用户的信息主要包括用户自己发布的内容、用户转发的内容以及用户所关注的人，其中用户发布和转发的内容都能很好地表示用户所关心的主题。在社交网络中，推荐用户感兴趣的信息可以转换成推荐给用户感兴趣的主题。而在社交网络中，可以用关键词向量的形式来描述信息的主题。所以，为了进行主题的分类，首先要从用户所发布的内容中提取常用的关键词作为主题描述的备选词。

为了发现更多相同主题的信息，使用信息之间的语义关系来代替内容相似度。本文使用统计学的方法来衡量关键词之间的语义相关性。我们认为，如果两个关键词频繁地出现在同一个文档中，则两个关键词具有一定的语义相关度。为了避免一些对描述主题没有帮助的词造成影响，如“的”、“你”等，采用了停用词列表来过滤那些对描述主题无用的词。在获取到所有的关键词之后，根据两个关键词出现在同一条内容中的概率计算它们之间的语义相关度。在本文中，使用关键词之间的支持度[1]作为它们的语义相关度指标：

$$rel_s(x,y)=\frac{supp(x,y)-MIN_s}{MAX_s-MIN_s} \tag{6}$$

其中，$supp(x,y)=\frac{N_{x\cap y}}{N}$，$N_{x\cap y}$ 表示两个关键词同时出现的文档个数，N 表示文档的总个数。

在实际中，一个词可能有多种不同的语义，如“苹果”一词，可能表示一种水果，也可能表示一家科技公司。用户在发布一条包含“苹果”的信息时所表达的含义是不确定的。因此，在对关键词进行聚类时需要将同一个词按照不同语义分入多个不同的类中。所以，采用了模糊聚类算法 FCM。根据 2.1 节所给出的 FCM 聚类算法，将关键词相关度向量的余弦距离作为它们之间的距离函数，计算后可以得到 n 个由关键词组成的类，每一类表示一个特定的主题，其中 n 为聚类前指定的主题个数。每一个主题类包含聚类得到的关键词集合，并代表一个特定的社区。

在本文的方法中，采用 FCM 聚类算法的优点是可以自动生成分类。在实际应用中，一些社交网络服务会采用人工定义关键词向量进行分类的方法对信息进行处理，或者采用贝叶斯概率进行分类的方法，这些方法都可以对现有分类方法进行替换，并不影响整体的信息推荐算法。

通过聚类算法得到主题分类之后，就可以对用户进行社区分类。如前文所述，用户具有多角色的属性，每个用户在不同社区中所需要推荐的信息也是不同的，因此也需要将一个用户分入多个不同的社区中，然后根据不同的社区主题进行相关的用户推荐。下面给出用户分类的具体算法。

对于用户 u_i，由他发布或转发的内容得到他的主题向量 $\vec{V}_i=(c_{i1},c_{i2},\cdots,c_{in})$，其中，$c_{ix}$ 的计算公式如下所示：

$$c_{ix}=\begin{cases}1, & u_i \text{ 的内容包含 } t_x \text{ 中的关键词}\\ 0, & u_i \text{ 的内容未包含 } t_x \text{ 中的关键词}\end{cases} \tag{7}$$

其中，t_x 表示第 x 个主题类，由聚类算法得到的相应分类的关键词集合组成。c_{ix} 的物理含义是：如果用户 u_i 发布或转发的内容中包含第 x 个主题所包含的关键词，则认为 u_i 与此主题有关。

设社交网络中共有 m 个用户，根据用户之间的关注关系，可以得到关注矩阵 $SN=\begin{pmatrix} f_{11} & \cdots & f_{1m} \\ \vdots & \ddots & \vdots \\ f_{m1} & \cdots & f_{mm} \end{pmatrix}$，其中 f_{ij} 的取值为：

$$f_{ij}=\begin{cases}1, & u_i \text{ 关注 } u_j \\ 0, & u_i \text{ 未关注 } u_j\end{cases} \tag{8}$$

因为用户对自身并没有关注关系，并且在推荐时也不需要考虑用户自己，所以将 f_{ii} 的取值定为 0。

根据上面得到的用户主题向量和关注矩阵 SN，可以计算出用户 u_i 在第 x 个社区中的关注关系向量：

$$\vec{F}_{ix}=(f_{i1_x},f_{i2_x},\cdots,f_{im_x}) \tag{9}$$

其中，f_{ij_x} 的取值为：

$$f_{ij_x}=\begin{cases}f_{ij}, & c_{ix}=1 \text{ 且 } c_{jx}=1 \\ 0, & \text{其它情况}\end{cases} \tag{10}$$

f_{ij_x} 表示 u_i 在第 x 个社区中对 u_j 的关注关系，当两个用户同属于第 x 个社区且 u_i 在全局社交网络中关注 u_j 时，则取值为 1，否则取值为 0。

由此，可以得到第 x 个社区中的 $m\times m$ 维用户关注关系矩阵 SN_x：

$$SN_x=(\vec{F}_{1x},\vec{F}_{2x},\cdots,\vec{F}_{mx})^{\mathrm{T}} \tag{11}$$

需要注意的是，采用这一方法所得到的用户社区分类存在两个问题。首先，由于关键词存在多义性，用户可能会被分入无关的社区中。比如说，用户发表了与苹果手机相关的信息，但在分类时，由于关键词“苹果”存在多义性，其与“农业”也有一定的相关性，因此，用户可能会被分入与“农业”相关的社区中。其次，社区分类与消息的数量无关，用户可能只发布过一条与某社区相关的信息，他也会被分入此社区内。为了解决这两个问题，给出了一种用户的社区权重计算方法，其主要思想是：用户发布或转发与某个社区主题有关的信息越多，或者他所关注的用户发布或转发与某个社区主题有关的信息越多，则他对此社区的兴趣度越大。当用户发布的信息中不仅包括“苹果”还包括“iPhone”等关键词时，他对于“科技”社区的权重就要高于“农业”社区。而如果用户仅仅发布过一条与“苹果”相关的信息，且没有发表过其它与“科技”相关的信息，他对于“科技”社区的权重就会很低。具体的社区权重计算方法将于 4.3 节中详述。

4 基于社区的用户推荐算法

4.1 用户的关注相似度

为了推荐用户可能感兴趣的人，做出以下的假设：

(1)同一社区中的两个用户如果关注相似的人，则两个用户的兴趣是相近的；

(2)用户关注的人很有可能是他所感兴趣的人；

(3)与某用户兴趣相近的人所关注的用户也可能是他所感兴趣的用户。

由以上的假设可以推出，找到与用户关注列表相似的其他用户是用户推荐时首先需要解决的问题，所以需要首先计算用户之间的关注相似度。在本文的方法中，第 x 个社区中两个用户 u_1 和 u_2 的关注相似度定义如式(12)所示：

$$sim_{Fx}(u_1,u_2)=\frac{|I_x(u_1)\bigcap I_x(u_2)|}{\min[|I_x(u_1)|,|I_x(u_2)|]} \tag{12}$$

其中，$I_x(u_1)$ 和 $I_x(u_2)$ 分别表示两个用户在第 x 个社区中所关注的用户列表，$|I_x(u_1)|$ 表示用户 u_1 在此社区内关注的用户数量。

用两个用户关注的用户数量中较小的一个来作为分母，这是因为我们认为如果一个用户大多数的关注都与另一个用户的关注相似，就可以认为他与另一个用户的兴趣基本相似，即使另一用户还关注了一些他未关注的人。

根据用户在某个社区内的相似度和关注关系，可以通过协同过滤计算用户对其他未关注用户的评分值。但是，传统的协同过滤算法存在一个问题，即无法发现那些与已关注用户相似但未被其他相似用户关注的用户。比如说，某用户较关注计算机方面的内容，他关注了一些发布计算机内容的用户；而一名最近刚加入社交网络的用户同样经常发布计算机方面的内容，但由于他加入较晚，因此关注数较少，传统的协同过滤算法不能通过相似用户的方法将其推荐给对计算机内容感兴趣的用户。因此，我们考虑加入内容相似度，对协同过滤算法进行修正。

4.2 用户的内容相似度

使用关键词对某个用户在某个主题中的 TF-IDF 值作为衡量其重要性的标准。对于某关键词 w，其对于用户 u 在第 x 个社区下的 TF-IDF 计算公式为：

$$tfidf_x(w,u)=tf_x(w,u)\times idf_x(w) \tag{13}$$

其中，

$$tf_x(w,u)=\frac{n_{xwu}}{n_{xu}} \tag{14}$$

$$idf_x(w)=\log\frac{N_x}{N_{xw}} \tag{15}$$

其中，$tf_x(w,u)$ 表示关键词 w 对于用户 u 在第 x 个社区下的词频，$idf_x(w)$ 表示关键词 w 对于用户 u 在第 x 个社区下的逆文档频率。n_{xwu} 表示用户 u 发布的含有关键词 w 的信息数目，n_{xu} 表示用户 u 发布的与第 x 个社区相关的关键词数目，N_x 表示第 x 个社区中所有信息的数目，N_{xw} 表示第 x 个社区中含有关键词 w 的信息数目。

根据计算得到的关键词对用户的 TF-IDF 值，使用余弦相似度来计算两个用户在同一社区内所发布的信息的相似度，即内容相似度。在第 x 个社区中，两个用户之间的内容相似度计算公式如下：

$$sim_{Cx}(u_1,u_2)=\frac{\sum_{i=1}^{n}tfidf_x(w_i,u_1)\times tfidf_x(w_i,u_2)}{\sqrt{\sum_{i=1}^{n}tfidf_x^2(w_i,u_1)}\times\sqrt{\sum_{i=1}^{n}tfidf_x^2(w_i,u_2)}} \tag{16}$$

4.3 用户的推荐值计算

类似于用户对物品的打分，可以认为如果一个用户关注了另一个用户，则前者对后者的评分值为 1；如果一个用户未关注另一用户，则前者对后者的评分值为 0。但是这种方法会遗失前文所提到的内容相似用户的评分信息。由 4.2 节计算得到的用户之间的内容相似度，可以对用户之间的评分值进行扩充。对一个用户来说，如果一个信息发布者与该用户

关注的其他用户在内容上相似，则认为此信息发布者有一定概率是用户希望关注的对象。在所提方法中，对用户之间的评分值公式计算如下：

$$rat_x(u_1,u_2)=\begin{cases}1, & u_1 \text{ 关注 } u_2\\ 0.5, & \text{存在 } sim_{Cx}(u_i,u_2)>\theta\\ 0, & \text{其它情况}\end{cases} \tag{17}$$

其中，u_i 为 u_1 在第 x 个社区中的关注列表中的任一用户，θ 为指定的阈值，我们认为内容相似度超过此阈值的用户所发布的信息是大体相似的。

根据式(17)计算得到的用户评分值以及式(12)得到的用户关注相似度，用户 u_1 对用户 u_2 在第 x 个社区中的推荐值计算公式如下：

$$R_x(u_1,u_2)=\frac{1}{|N(u_1)|}\sum_{u_i\in N(u_1)} sim_{Fx}(u_1,u_i)\times rat_x(u_i,u_2) \tag{18}$$

其中，$N(u_1)$为与 u_1 关注相似度最相似的用户列表，该用户列表可以由对用户的相似用户进行关注相似度降序排序得到，其数量可以由相似度的阈值来控制，也可以指定固定的用户数目，$|N(u_1)|$表示这个集合包含的用户个数。

在实际的推荐过程中，根据用户在各个社区的权重，对各个社区内推荐值进行加权，从而实现全局的用户推荐。我们认为，用户相对于某个社区的权重应由两部分组成：1)用户所发布的内容与该社区的关联性；2)用户所关注用户发布内容与该社区的关联性。前者说明用户经常参与该社区的内容创作，后者说明用户倾向于接收与该社区相关的内容。

对用户 u_i 的社区权重向量的初始化定义如下：

$$\vec{V}_i=(w_{i1},w_{i2},\cdots,w_{in}) \tag{19}$$

其中，$w_{ix}=\frac{N_{ix}}{N_i}$。$N_{ix}$ 表示用户 u_i 发布的内容中包含与第 x 个社区相关关键词的记录数，N_i 表示用户 u_i 发布的所有记录数。在这里，如果多个关键词出现在同一条信息中，只记为一条记录。

由每个用户的社区权重向量可以得到全局的用户权重初始化矩阵 W：

$$W=(\vec{V}_1,\vec{V}_2,\vec{V}_n) \tag{20}$$

然后，对全局的用户权重矩阵进行迭代计算，经过若干次迭代后，得到最终的社区权重矩阵：

$$W_k=\alpha W+(1-\alpha)SN\times W_{k-i} \tag{21}$$

其中，α 是用户给定的权重，SN 是第 3 节中给出的全局的关注矩阵，而 $W_0=W$。迭代计算的次数可以由矩阵收敛到差异值小于一个指定阈值决定，也可以指定一个固定的次数。

需要注意的是，这个矩阵需要定期进行更新，这是因为社交网络会不断有新用户加入。由于将用户的偏好从关键词抽象到了社区，向量的长度会大幅缩短，因此迭代计算社区的权重矩阵会比计算关键词的权重矩阵的运算量减小很多。

经过迭代计算后得到的社区权重矩阵可以用来对已有的社区推荐值进行加权，进而得到最终的推荐值。给出全局的推荐值计算公式如下：

$$R(u_1,u_2)=\max[w_{u_1 i}\times R_i(u_1,u_2)] \tag{22}$$

其中，$w_{u_1 i}$表示 u_1 对第 i 个社区的权重。没有将所有社区的加权推荐值相加是因为我们认为不应该让在多个社区弱相关的用户的推荐值超过在单个社区内强相关的用户。因此，取用户在多个社区中加权推荐值的最大值作为最终的推荐值。

在推荐可能感兴趣的用户时，推荐值最大的 N 个用户会被按照其推荐值大小降序排列推荐给用户。

5 实验设计与评估

为了验证并评估文中所提推荐算法的有效性，使用了一个国内最大的社交网络“新浪微博”的真实用户数据集，并对其使用本文中所提出的推荐方法与已有推荐算法进行比较与验证。数据集由 3 部分组成，第一部分为用户数据，其中包含 63641 条新浪微博用户信息，每个用户信息包括用户 ID(uid)、用户昵称、用户姓名及用户性别等；第二部分为微博数据，其中包含 84168 条微博的相关信息，每条信息包括微博的 ID 号(mid)、发布时间、微博内容、发表的用户 ID(uid)以及微博的主题等；第三部分为关注数据，由 1391718 条用户关注关系组成。由于新浪微博的网站限制，对每个用户只能抓取最多 200 条该用户的关注关系，因此用户之间的关注关系有部分被遗失。

5.1 实验评估方法

首先对数据集中的用户进行了社区分类。根据用户发表微博的内容，将 63641 名用户分入了 13 个不同主题所对应的社区中，其中，每个用户都可以存在于多个不同的社区中。为了衡量用户推荐的准确性，首先检查算法所推荐的用户是否发表过与被推荐用户关注的主题相关的微博，如果算法所推荐的用户发表过相关主题的微博，且其存在于用户的关注列表中，则认为这次推荐是准确的。这是出于一个假设，即只有存在共同话题的关注对象才是有意义的。我们认为，如果一个用户对另一用户感兴趣，那么他应该发表或转发过另一用户所发表主题相关的内容。可以通过算法推荐的用户在已关注用户中的命中率来预测未关注的推荐用户的准确率。

设 $top_{A,N}(v)$表示 A 算法对用户 v 推荐值最大的 N 个用户的集合，使用召回率来评估算法的准确性：

$$recall_{A,N}(v)=\frac{|C_{wp_{A,N}(v)}|}{N} \tag{23}$$

其中，$C_{wp_{A,N}(v)}$ 表示算法 A 所推荐的 N 个用户中根据前文所述标准被认为推荐准确的用户集合，$|C_{wp_{A,N}(v)}|$表示其包含的用户数量。

在计算出所有推荐用户的召回率以后，用平均召回率来衡量算法在整个数据集上的整体效果：

$$\Delta_{recall}(A,N)=\frac{\sum_{v\in V} recall_{A,N}(v)}{|V|} \tag{24}$$

其中，V 表示数据集中用户的集合，$|V|$表示数据集所包含的用户数量。

实验的硬件环境为 Intel i5-3230 2.6GHz 双核处理器、8GB 内存，系统环境为 Windows 10 操作系统，开发环境为 Microsoft Visual Studio，开发语言为 C#。

5.2 实验结果

分别使用了本文所提出的推荐算法与协同过滤算法[3]对数据集进行推荐值计算，并根据数据集中所包含的用户关注关系对不同算法所得到的 Top 10 推荐用户的准确率进行评估，评价的指标为召回率。

为保证评估结果的公平性，对本文算法与协同过滤算法使用了相同的评分值计算公式，即式(17)，区别在于协同算法使用的是全局的评分值。我们将用户的关注关系作为评分值的主要指标，并将用户之间的内容相似度作为补充。

如图1所示，文中算法在推荐准确性的召回率上要明显优于协同过滤算法，最多可提高43%的准确性。这是因为其引入了语义相似度的比较，算法可以发现更多内容上相似的用户。传统的协同过滤算法忽略了用户发布内容与关注对象之间的逻辑关系，不能找到用户关注的原因。根据主题分类的用户关系能够更有效地刻画用户之间的关注原因，从而更好地根据用户的兴趣推荐更多可能有类似兴趣的人。

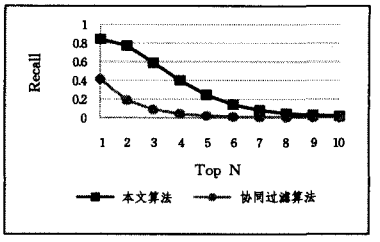

图1 算法的召回率比较

本文所提出的推荐算法除了在召回率上要明显优于协同过滤算法以外，在运行时间上也有明显的优势。如图2所示，在实验所采用的数据集上，协同过滤算法耗时17186s，而基于社区分类的推荐算法仅耗时2185s，节省了87.2%的运行时间。这是因为计算用户之间相似度和推荐值的算法时间复杂度均为$O(n^2p)$，其中n为用户的数量，p为用户最大的关注人数。本文所提出的方法虽然需要计算多个社区的用户相似度和推荐值，但由于每个社区所包含的用户数量要远远少于社交网络的用户总量，因此用户数量的平方值会减小很多，这使得运行时间极大地减少。此外，由于各个用户社区的推荐值是相互独立的，文中算法更适合使用分布式并行计算来进行处理。

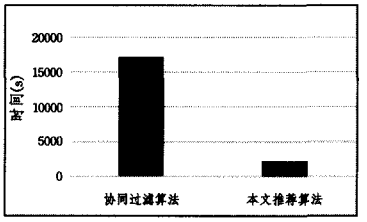

图2 算法的运行时间比较

综上所述，通过实验可以发现，本文所提出的基于社区分类的用户推荐算法可以在极大地提高计算效率的基础上，有效地提高用户推荐的准确性。

结束语 本文提出了一种基于主题的社区网络信息发现与推荐算法，将社交网络按照不同的主题进行了划分，将用户按兴趣分入不同的社区，使用用户发布的信息内容来描述关注关系的类型。综合考虑了传统用户推荐方法中协同过滤算法以及相似度算法的思想，提出了一种混合的用户推荐算法。与现有的推荐方法相比，所提算法克服了不同主题间的相互干扰，避免了由与主题无关的关注关系带来的准确性问题。

下一步工作中需要解决主题分类的准确性问题。现有的主题分类方法为了避免在信息分类时的速度过慢，选择了时间复杂度较低但准确性稍差的主题分类方法。未来将研究分类准确性更高且时间复杂度可接受的信息分类算法。

参 考 文 献

[1] Singhal A. Modern information retrieval. : A brief overview [J]. IEEE Data Engineering Bulletin, 2001, 24(4): 35-43

[2] Salton G, Buckley C. Term-weighting approaches in automatic text retrieval [J]. Information Processing & Management, 1988, 24(5): 513-523

[3] Bellogín A, Cantador I, Castells P. A comparative study of heterogeneous item recommendations in social systems [J]. Information Sciences, 2013, 221: 142-169

[4] Chen Ji-lin, Geyer W, Dugan C, et al. Make New Friends, but Keep the Old-Recommending People on Social Networking Sites [C]// Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2009. New York: ACM, 2009: 201-210

[5] Nock N, Nielsen F. On weighting clustering [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28 (8): 1223-1235

[6] Yu S. The dynamic competitive recommendation algorithm in social network services [J]. Information Sciences, 2012, 187: 1-14

[7] Lv Lin-yuan, Matus M, Yeung C, et al. Recommender systems [J]. Physics Reports, 2012, 519: 1-49

[8] Carrer-Neto W, Hernandez-Alcaraz M, Valencia-Garcia R, et al. Social knowledge-based recommender system: Application to the movies domain [J]. Expert Systems with Applications, 2012, 39: 10990-11000

[9] Kardan A, Ebrahimi M. A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups [J]. Information Sciences, 2013, 219: 93-110

[10] Zhou Xian-ke, Wu Sai, Chen Chun, et al. Real-time recommendation for microblogs [J]. Information Sciences, 2014, 279: 301-325

[11] Zhao Qin, Wang Cheng, Jiang Chang-jun. HSim : A Novel Method on Similarity Computation by Hybrid Measure[C]// 6th International Conference on Information and Communication Systems, 2015. Amman: IEEE, 2015: 160-165

[12] Wang Yu, Gao Lin. Social Circle-Based Algorithm for Friend Recommendation in Online Social Networks [J]. Chinese Journal of Computer, 2014, 37(4): 801-808(in Chinese)
王玙，高琳. 基于社交圈的在线社交网络朋友推荐算法[J]. 计算机学报，2014，37(4)：801-808

[13] Qian Fu-lan, Li Qi-long. Social Recommendation Combining Global and Dual Local Information [J]. Computer Science, 2016, 43(2): 57-59, 94(in Chinese)
钱付兰，李启龙. 结合全局与双重局部信息的社交推荐[J]. 计算机科学，2016，43(2)：57-59，94

[14] Jiang Sheng, Wang Zhong-qun, Xiu Yu, et al. Collaborative Filtering Recommendation Method Based on Dynamic Social Behavior and Users' Background Information [J]. Computer Science, 2015, 42(3): 252-255, 265(in Chinese)
蒋胜，王忠群，修宇，等. 基于动态社会行为和用户背景的协同推荐方法[J]. 计算机科学，2015，42(3)：252-255，265