社交网络中模块关系树的相似性算法的研究

原福永[1], 韩 丽[1], 赵英梅[1,2]
(1. 燕山大学 信息科学与工程学院, 河北 秦皇岛 066004; 2. 河北外国语职业学院, 河北 秦皇岛 066311)

摘 要: 提出一种基于模块关系树的分析方法,考虑每个实体与用户之间的兴趣、住址和共同好友等相关因素,制定不同的关系树,然后根据路径长度计算各因素的相关度值,最后综合每个实体模块,从而筛选出关系最密切的实体。实验结果证明,该算法能过滤掉大量无关信息,有效找出最相关的实体,提高了搜索结果的准确率。

关键词: 兴趣关系树; 地址关系树; 共同好友; 相关度; 社交网络; 信息过滤
中图分类号: TP391 **文献标志码**: A **文章编号**: 1001-3695(2012)02-0698-03
doi:10.3969/j.issn.1001-3695.2012.02.079

Research on module relation tree similarity algorithm in social networks

YUAN Fu-yong[1], HAN Li[1], ZHAO Ying-mei[1,2]
(1. *School of Information Science & Engineering, Yanshan University, Qinhuangdao Hebei* 066004, *China*; 2. *Hebei Vocational College of Foreign Languages, Qinhuangdao Hebei* 066311, *China*)

Abstract: This paper proposed an analysis method based on the modular relational tree. It considered the correlative factor of interest, address and common factors between users and each entity, formulated different modular relational tree, then calculated the correlation of each factor according to the length of the path, and finally integrated module for each entity to filter out the most closely related entity. The experimental results show that the algorithm can filter out a lot of irrelevant information and identify the most relevant entity effectively, and the accuracy of search results is great improved.

Key words: interest relational tree; address relation tree; common friends; correlation; social network service; information filtering

随着互联网的发展,越来越多的人选择在网上交友,因此社交网络(social network service,SNS)日益成为人们必不可少的联系工具。目前,国外比较流行的社交网站有 Facebook、MySpace 和 Orkut 等,国内做得比较出色的有人人网、开心网等。SNS 的主要作用是为一群拥有相同兴趣与活动的人创建在线社区[1]。社交服务网站使用户与朋友保持紧密的联系,扩大用户的交际圈,其提供的搜索好友工具帮助用户找到失去联系的朋友[2]。但是目前国内社交网站的搜索结果多是随机显示,或者按姓氏首字母来排列,未根据搜索实体与用户之间的相关度筛选出最期望实体。除此之外,当搜索某个好友时,还会遇到一些具体问题,例如:a)不能确定要搜索实体的正确名字,只记得发音或者名字的一部分;b)能确定实体正确名字,但显示很多同名结果,不能筛选出期望实体。因此,怎样筛选出用户最想找的实体是本文的主要任务。

1 相关工作

关于树结构的研究很多,它可以使关系层次化,降低计算的复杂度,还可以解决数据的存储、索引和提取等相关问题[3~5]。相关度的分析和研究有很多[6,7],最常用的方法是余弦算法[8],但其缺陷是不能很好地解决同义词问题,从而导致漏掉许多正确结果,降低精确度。目前,将树结构和相关度结合起来分析问题的实例比较少见[9]。

人人网中,好友搜索的结果显示的是与名字相关的全部实体信息,仅基于有无共同好友这一因素。如果当两者没有共同好友时,精确度就会大大降低。无关信息的显示增大了用户查找负担。考虑到树的层次结构的优势,将它利用到相关度的分析中,不仅可使实体间的关系更加明朗,还可以提高查找效率,从而筛选出最相关的实体,过滤掉大量无关信息。

本文将提出的方法分别和利用余弦计算相似性搜索方法及人人网搜索方法进行了比较。

2 建立系统

2.1 系统描述

模型主要分为兴趣模块、地址模块和共同好友模块三个模块。为要匹配的搜索实体建立一个模型,分别计算每个模块的得分,最后将三部分综合起来作为分析用户和搜索实体之间相关度的依据。总的流程如图 1 所示。U 表示用户,每个搜索实体 E 表示为一个三元组 $E=(H,A,F)$。其中:H 代表兴趣;A 代表地址;F 代表共同好友。兴趣用三元组 $H=(S,R,B)$ 表示,其中,S 代表运动,R 代表娱乐,B 代表书籍。

2.2 预处理

预处理部分主要解决引言中所提到的问题 a):不能确定要搜索实体的正确名字,只记得发音或者名字的一部分。由于搜索结果总是会将相似或者同音的名字全部列出,因此搜索的结果与用户给出的名字将出现不同的匹配现象。以三个字的名字为例,可归结为五种类型:前匹配——最后一个字不匹配;

收稿日期: 2011-06-21; **修回日期**: 2011-07-24

作者简介:原福永(1958-),男,黑龙江鸡西人,教授,硕导,主要研究方向为网络信息检索、数据库技术等(fyyuan@ysu.edu.cn);韩丽(1986-),女,河北廊坊人,硕士,主要研究方向为搜索引擎、信息处理等;赵英梅,助教,硕士研究生,主要研究方向为计算机相关.

后匹配——最前面的字不匹配;两边匹配——中间的字不匹配;中间匹配——两边的字不匹配;全匹配——全部匹配和不匹配——完全不匹配。利用式(1)计算搜索结果中某个名字 x 在预处理部分的得分 S_{pre}。

$$S_{\text{pre}} = P(L_N(x)) \times P(L_{E,i}(x)) \tag{1}$$

其中:$P(L_N(x))$ 表示在搜索显示的结果列表 L_N 中,名字为 x 出现的概率,计算公式如式(2):

$$P(L_N(x)) = \frac{\text{名字为 } x \text{ 的数量 } n}{\text{结果表中名字总数 } N} \tag{2}$$

$P(L_{E,i}(x))$ 表示在名字 x 的实体列表中,某实体的概率,计算公式如式(3):

$$P(L_{E,i}(x)) = \frac{1}{\text{名字为 } x \text{ 的总数 } n} \tag{3}$$

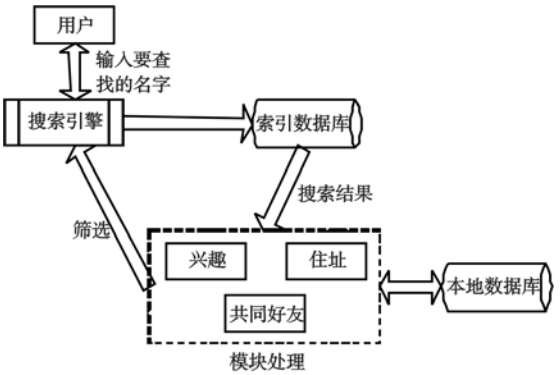

图1　模块关系树计算相似性的数据流程

举例说明:假设用户输入的名字为张山,搜索得到的结果可能为:张山(5 个)、张珊(2 个)、张姗姗(2 个)、张 * 山(3 个)。则姓名为张山的各个计算值为

$$P(L_N(x)) = \frac{5}{12}, P(L_{E,i}(x)) = \frac{1}{5}, S_{\text{pre}} = \frac{5}{12} \times \frac{1}{5}$$

2.3　兴趣模块

兴趣可描述为 $H = (S, R, B)$。其中:运动 S 主要包括田径、球类、水上和其他类;娱乐 R 主要包括明星、影视、音乐、游戏和其他类,关系树的部分结构如图 2 所示;书籍 B 的划分可遵循《中国图书馆图书分类法》进行分类,这里只考虑文艺方面等常见的书籍。实验部分中搜集很多常见的相关词汇放在数据库里,并对数据的类别进行分层。

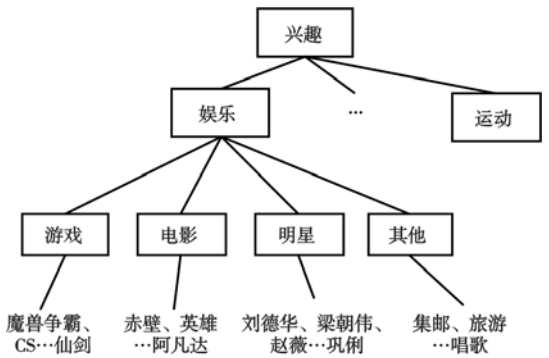

图2　兴趣关系树的部分结构

利用式(4)计算兴趣在关系树中传播所得的分数为

$$S_E(T(i) \rightarrow T(j)) = S_E(T(i)) \times \sigma^{d(T(i),T(j))} \tag{4}$$

式(4)表示每一步传播所得的分数。其中:σ 是衰减因子;$d(T(i),T(j))$ 表示两个点之间的路径长度。最后的总得分 S_E 表示为

$$S_E = \sum_{i,j \in T} S(T(i) \rightarrow T(j)) \tag{5}$$

S_E 的初始值为 S_{pre},由此可知,得分最高的实体就是和用户兴趣最相近的人。可用式(6)表示在实体列表中第 j 个实体得分最高。

$$\hat{L_{E,j}}(x) = \max S_E \tag{6}$$

基于兴趣关系树查找路径的算法描述为

输入:关系树 T,用户 U 的兴趣集 U_E,所有搜索实体的兴趣集 R,初始路径长度设为 0。

输出:每个实体的得分 S_E,并找出最大得分 $\hat{L_{E,j}}(x)$。

a)从用户的兴趣集 U_E 中抽取出一项 e_i,判断是否为最后一项,false 则在 T 中进行比对,找到所对应的节点,true 则转到 f);

b)将 e_i 与搜索实体中的每一项进行匹配;

c)有相匹配的,路径长度 L 加 1,记录 L,并计算相匹配实体的得分 S_E,然后抽取下一项 e_{i+1};

d)若没有相匹配的,路径长度 L 加 1,上升到父节点层,判断是否为根节点,false 则重复 b),true 则转到 e);

e)抽取下一项 e_{i+1};

f)计算每个实体的总得分,并找出最高分者;

g)结束。

例如,用户的兴趣包括:刘德华、《赤壁》、唱歌;某个搜索实体 i 的兴趣包括:梁朝伟、魔兽争霸、唱歌。则将用户的每一项和实体 i 进行最短路径查找,结果如表 1 所示。

表 1　最短路径

最短路径	路径长度
刘德华—明星—梁朝伟	2
《赤壁》—电影—娱乐—明星—梁朝伟	4
唱歌—唱歌	1

2.4　地址模块

现实生活中居住在邻近的地方成为好友的可能性比较大。因此,把居住地址作为考虑用户和搜索实体的相关性因素。从地理词典中收集各省内的地理名称,可以观察到,一般用户写个人籍贯资料时最低只到县级,因此只收集县级及以上地理名称。

实现方法与兴趣模块相类似,只是算法稍作修改。因为在兴趣模块中,相比较的总是在同一层中。但是,地址最低一级可能分属于不同层。此外,地址匹配是精确匹配,而兴趣匹配可以是相似匹配。用地址关系树来存储各个实体的住址信息,根节点为国家名称,向下范围逐层递减,依次为省级名称、市级名称、县级名称。各个节点包括从地理词典中收集的相应的地址名词。算法的简单描述如下:

a)建立地址关系树 T',从顶向下,地理范围逐层递减。

b)分别找到从根节点通向用户和每个实体住址的路径,此路径一定是唯一的,因此出现一个用户和实体路径的交叉点,记录这个点。

c)利用这个交叉点计算两者之间的路径长度,通过式(7)计算得分 S_A:

$$S_A = S_{\text{pre}} \times \eta^{d(T'(i),T'(j))} \tag{7}$$

其中:S_{pre} 为初始值,η 为衰减因子。

例如:用户实体 u_1 和 u_2 的住址分别为河北省秦皇岛市昌黎县和河北省石家庄市赵县。则自顶向下找到的路径分别为:中国—河北省—秦皇岛市—昌黎县,中国—河北省—石家庄市—赵县。交叉点为河北省,两者之间的路径为昌黎县—秦皇岛市—河北省—石家庄市—赵县,路径长度为 4。

2.5　共同好友模块

如果两个人有共同好友,则表明两个人的熟识度很高,因此考虑这一项。共同好友可能是直接的,也可能是间接的。例如,某个搜索实体的好友的好友可能为用户的好友。因此,设置不同的权重 α′、β′,且 α′ + β′ = 1,分别表示直接和间接共同

好友的权值。用 S_F 表示本模块的得分：

$$S_F = S_{\text{pre}} \times (\alpha' \times \frac{n_1}{N} + \beta' \times \frac{n_2}{N}) \tag{8}$$

初始得分为 S_{pre}，n_1、n_2 分别表示直接和间接共同好友的个数，N 表示直接和间接好友的总个数。

2.6 综合模块

为增加搜索结果的准确性，对于每一个搜索实体，综合以上三个模块，进行多方面多层次的分析，使得结果更加满足用户的需求。利用式(9)计算总得分 S_T：

$$S_T = \alpha \cdot S_E + \beta \cdot S_A + \gamma \cdot S_F \tag{9}$$

其中：α、β、γ 为比例因子，且 $\alpha+\beta+\gamma=1$，$0<\alpha,\beta,\gamma<1$。最高 S_T 值所对应的实体就最可能是用户最期望的实体。根据得分对搜索实体进行筛选，找出 S_T 值最高的前几项，避免结果唯一性所带来的准确率的降低。

3 实验结果和评价

3.1 实验设置

本地数据库位于一台拥有 2.9 GHz 处理器、2 GB 内存和 Windows XP 的 PC 机上，使用 SQL Server 2000。数据库中的数据集主要为兴趣集、地址集和好友集。公式中的参数范围均在 0～1。手动从人人网、QQ 校友、周围同学中抽取与收集 1 000 个名字和实际信息，其中包括兴趣、地址和好友等。数据库中已存放大量的各类词汇，并手工地进行标注分类。部分信息如表 2 所示。

表 2　部分分类信息

分类	兴趣			地址		
	运动	娱乐	书籍	省级	市级	县级
数量	100	2717	100000	34	333	2862
总和		102817			3229	

3.2 实验分析

对实验结果采用准确率(precision)和召回率(recall)两种评价指标，其定义如下：

$$\text{precision} = \frac{\text{找出正确实体的次数}}{\text{总的查询次数}}, \text{recall} = \frac{\text{搜索出的结果数}}{\text{总的相关数}}$$

采用分组实验方法来分析准确率和召回率的关系。先根据搜索结果计算召回率，再按照召回率从小到大进行分组，最后计算每组内的准确率。通过搜索次数的增加，将模块关系树方法与利用余弦方法及人人网所用方法的准确率进行比较，如图 3 所示，可见准确率有明显的改善。人人网只有当两个实体有共同好友时准确率较高，因此，随着召回率的增加，平均准确率下降得很快。而相似性算法由于考虑的细节不周全，灵活性差，使得结果的准确率不够理想。

衰减因子对实验结果产生的影响如图 4 所示。衰减因子选得过小，会使得层级之间的特征值不明显，导致结果筛选不准确；选得过大，又会扩大某些层级的作用，导致遗漏其他可能的正确结果。所以，经过反复实验，选取了一个适度的 σ、η 值。

综合模块中涉及到的 α、β、γ 的值的确定方法为：分别对每个单独模块进行实验，算出每个模块的准确率，然后对三个准确率进行归一化处理，如式(10)所示：

$$\alpha = \frac{a_1}{\sum_{i=1}^{3} a_i}, \beta = \frac{a_2}{\sum_{i=1}^{3} a_i}, \gamma = \frac{a_3}{\sum_{i=1}^{3} a_i} \tag{10}$$

其中，a_i，$i=1,2,3$ 分别表示三个模块的准确率。

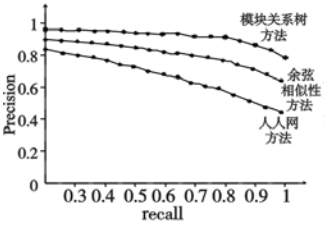

图3　三种方法的准确率比较

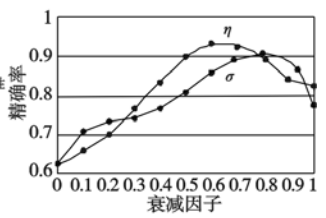

图4　衰减因子对准确率的影响

实验中一些结果的准确率不是很高。经过分析，其原因为某些用户想找的人和自己的紧密度很小，所考虑的三个因素没有起到很大的作用，因此效果不明显，这种情况不属于本文的研究范围。当用户搜索和自己相关的人时，所考虑的各个因素起到很大的作用，准确率很高。

4 结束语

针对实际搜索不能满足用户要求的这一问题，本文提出一种基于模块关系树的相关度的计算方法。相对于传统的方法，该方法能够将问题转换为关系树计算路径问题，快速找出和用户最相关的人。通过对实体得分的排序，缩小了查询空间，过滤掉很多无关的实体，减轻了用户的查询负担，增加了用户的满意度。不可否认，该系统有些地方还不完善，还需要进一步改进和讨论，如：a)只考虑一部分兴趣名词及其之间的包含关系，还需要收集更多信息，并进行详细的分类；b)只考虑几个相关方面，还需要考虑更多影响因素，如考虑专业等方面；c)比例因子的选择都是根据经验值，还需要通过理论分析，找出能达到最佳性能的比例因子。

参考文献：

[1] SHIN D-H. The effects of trust, security and privacy in social networking: a security-based approach to understand the pattern of adoption [J]. Interacting with Computers, 2010, 22(5): 428-438.

[2] AMITAY E, CARMEL D, HAREL N, *et al*. Social search and discovery using a unified approach[C]//Proc of the 20th ACM Conference of Hypertext and Hypermedia. [S.l.]: ACM, 2009: 199-208.

[3] 田建伟，李石君. 基于层次树模型的 Deep Web 数据提取方法[J]. 计算机研究与发展，2011，48(1)：94-102.

[4] WU Shuang-yuan, WANG Zhao-qi, XIA Shi-hong. Indexing and retrieval of human motion data by a hierarchical tree[C]//Proc of the 16th ACM Symposium on Virtual Reality Software and Technology. [S.l.]: ACM, 2009: 207-214.

[5] 温春，石昭祥，杨国正. 一种利用度属性获取本体概念层次的方法[J]. 小型微型计算机系统，2010，31(2)：322-326.

[6] BENJAMIN M, CIRO C, FILIPPO M. Evaluating similarity measures for emergent semantics of social tagging[C]//Proc of the 18th International Conference of World Wide Web. [S.l.]: ACM, 2009: 641-650.

[7] MICHAEL D L, HANAN S, JAGAN S. Geotagging: using proximity, sibling, and prominence clues to understand comma groups[C]// Proc of the 6th Workshop of Geographic Information Retrieval. [S.l.]: ACM, 2010: 1-8.

[8] WAN Xiao-jun, XIAO Jian-guo. Exploiting neighbor hood knowledge for single document summarization and keyphrase extraction[J]. ACM Trans on Information Systems, 2010, 28(2): 1-34.

[9] QIN Teng, XIAO Rong, FANG Lei, *et al*. An efficient location extraction algorithm by leveraging Web contextual information[C]// Proc of the 18th SIGSPATIAL International Conference of Advances in Geographic Information Systems. [S.l.]: ACM, 2010: 53-60.