

开源社区中已有开发者的合作行为分析

陈 丹¹ 王 星¹ 何 鹏^{1,2} 曾 诚^{1,2}

(湖北大学计算机与信息工程学院 武汉 430062)¹ (湖北省教育信息化工程技术研究中心 武汉 430062)²

摘 要 理解开发者的合作行为是促进开源社区协作开发的关键一步。已有文献主要对社区成员的加入与迁移展开研究,而对社区中已有开发者之间的实际合作行为关注甚少。主要从开发者的交互关系与个人技能两个角度出发,探析社区已有开发者如何建立新合作以及哪些因素影响他们的合作。应用 Sourceforge.net 平台公开的数据进行实验,结果表明不同的开发者的偏好合作模式不同,拓扑距离上的优势(“朋友的朋友”关系)对开发者间首次合作的影响不大,开发者之前的合作次数越多越有利于他们再次建立合作,而开发环境(操作系统)与开发者(管理)经验对合作也有明显的影响。所得结论有助于提高开源社区已有开发者的合作意识,促进社区的稳定与可持续性。

关键词 开源软件社区,开发者合作网络,行为分析,偏好合作

中图法分类号 TP311 文献标识码 A

Towards Understanding Existing Developers' Collaborative Behavior in OSS Communities

CHEN Dan¹ WANG Xing¹ HE Peng^{1,2} ZENG Cheng^{1,2}

(School of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China)¹

(Hubei Province Engineering Technology Research Center for Education Informationization, Wuhan 430062, China)²

Abstract Understanding developers' cooperative behavior is an essential step to meet the needs of collaborative development activities. A number of studies have been carried out to explore the joining script of newcomers and the immigration process of developers in open-source software communities. However, few know about the actual collaborative behavior of existing developers in those communities. In this paper, we conducted an empirical study to gain an insight into how existing developers collaborate and which factors affect their collaborative behavior from the perspectives of both collective interaction and individual expertise. According to the data sets collected from the Sourceforge.net, the results show that different existing developers prefer different collaboration patterns, and short topological distance (also known as “friends of friends”) has very limited effect on their first collaboration, whereas the previous collaborations are positive to build new collaboration between the existing developers. By contrast, development environment (operating system) and administrator experience seem to be important factors that affect their potential collaboration. The findings are valuable for existing developers to maintain sufficient collaboration awareness, so as to improve the stability and sustainability of OSS communities.

Keywords OSS community, Developer collaboration network, Behavior analysis, Preferential collaboration

近十多年来,一个开源软件项目的成功很大程度上取决于其有一个适当规模的开发者社区^[1]。如今,开源软件已经被广泛接纳,全球开发环境下的开源社区也越来越流行,且逐渐吸引了大量来自不同国家、拥有不同开发经验的开发者自愿加入。开源社区是一个自组织、自愿参与的社会合作网络,这些开发者出于娱乐、收获经验、个人兴趣等原因以志愿者身份加入一个开源项目的开发^[2]。尽管这些开发者可能分布在世界各地,素不相识,但他们依旧能够像一个团队一样紧密合作,彼此交流开发经验,共享知识。

随着开源软件的流行,人们愈发渴望理解开源社区中的

诸多现象,如开源软件成功的决定因素^[3,4]、开发者自愿参与的动机^[5]、开发者合作机制与合作意识^[6]以及开发者社区的演化特性^[7]等。据我们所知,已有工作多倾向于关注新成员在社区的行为,关于已有开发者之间合作交互的工作还很少。而开源社区低门槛、松散合作的特性使得更多的贡献来自于那些长期留在社区的开发者,或愿意一直跟进某一个开源项目的忠实开发者。因此,本文主要从开发者的协同交互与个人技能两个角度出发,探析社区已有开发者如何建立新合作以及影响他们合作的因素。

两个已有开发者建立新的合作(包括首次与再次合作),

本文受国家“九七三”重点基础研究发展规划项目基金(2014CB340401),国家自然科学基金(61273216,61272111,61202048,61202032),湖北省重大科技创新计划(2013AAA020),武汉市青年科技晨光计划(2014070404010232),国家科技支撑计划子课题(2012BAH07B01)资助。

陈 丹(1992—),女,硕士生,主要研究方向为软件度量、软件维护;王 星(1991—),女,硕士生,主要研究方向为软件度量、复杂网络;何 鹏(1988—),男,讲师,CCF 会员,主要研究方向为软件度量、软件维护、复杂网络,E-mail: penghe@hubu.edu.cn(通信作者);曾 诚(1976—),男,副教授,CCF 会员,主要研究方向为服务发现和服务推荐。

一方面可能因为他们在拓扑结构上占优势,如“朋友的朋友”关系;另一方面可能因为共同的开发经验。基于这两个方面,提出以下两个研究问题:

问题₁:开发者合作网络的拓扑结构否影响社区已有开发者的合作行为?

问题₂:开发者的技能属性是否影响社区已有开发者的合作行为?

拓扑结构从网络的角度刻画开发者的行为特性,开发者技能属性主要考虑开发者自身特性对合作的影响。文章的主要贡献可归纳为:

(1)与已有工作相比,从开发者的协同交互与个人技能角度分析已有开发者间的首次与再次合作行为,为理解开发者社区的稳定性与可持续提供了一个新的视角。

(2)我们的发现也有助于为目标开发者提供潜在的同行推荐和辅助社区管理者进行合作的协调。

本文第1节为相关工作的介绍;第2节是所提方法框架与一些指标的定义;第3节是案例分析;第4节为实验结果的讨论与文章不足之处;最后为全文的总结与展望。

1 相关工作

Ye和Kishida^[1]认为获得内心满足感与角色晋级是促使开发者自愿加入开源软件社区的一种重要动力。研究指出,开发者的动机比较复杂,包括任务的责任感、好的管理环境、团队合作气氛等^[5,9]。另外,也有工作关注新成员与社区已有开发者之间的合作,如Bird等人^[8]研究发现新开发者之前的行为对其在团队中的成长有很大的影响;Krogh等人^[10]也调查了新成员的加入行为,发现新成员要想得到其他同行的认可则需要参加更多的技术讨论;Fronza等人^[11]发现对等编程(pair-programming)有利于一个新开发者加入敏捷项目开发;Herraiz等人^[12]发现开发者的加入行为是逐步进行的,不同于公司的软件开发。

关于影响开发者行为的因素,Dagenais等人^[13]总结了一些影响开发者融入团队过程的主要因素,如早期的实践经历;Ducheneaut^[14]指出一个开发者在社区的技术水平与影响力对其在一个项目的地位有影响;Hahn等人^[15]验证了一个开发团队的形成以及开发者之前的合作关系对团队形成的影响。此外,一些研究者也尝试采用社会网络分析方法发掘社区中的一些重要的成员,如“中介”成员识别^[16]、团队领导者(leader)的挖掘^[17]。Datta等人^[19]基于IBM的Jazz平台研究了敏捷项目开发中开发者合作网络的演化特性,发现随着版本的升级,开发者间合作更频繁。

Huang等人^[20]根据开发者提交的日志信息,构建开发者模块网络来描述开发者之间的交互,并建立LPP边缘学习过程模型用于划分开发团队中成员的角色。Allaho和Lee^[21]共同分析了开发者网络的交互关系,发现团队中专家与小角色成员之间有明显的交互行为,并且开发者的出入度与他们的贡献大小呈正相关。同时,在我们之前的工作中也发现,社区中新加入的开发者更倾向于与(度数或介数)中心性更大的已有开发者建立合作^[23]。

根据以上描述,虽然开源社区中开发者合作行为的分析已经引起了研究者的广泛关注,但与已有工作不同,本文将重点关注社区中已有开发者之间的合作,利用定性定量指标分析影响他们合作的可能因素,而不是关注那些新加入成员的行为。

2 实验设计

图1是本文的研究框架,主要包括4个部分,首先是开发者、项目数据信息收集,其次是构建开发者网络,然后基于开发者网络从网络结构与开发者自身属性两个方面分析开发者的合作行为,并进一步实施验证。

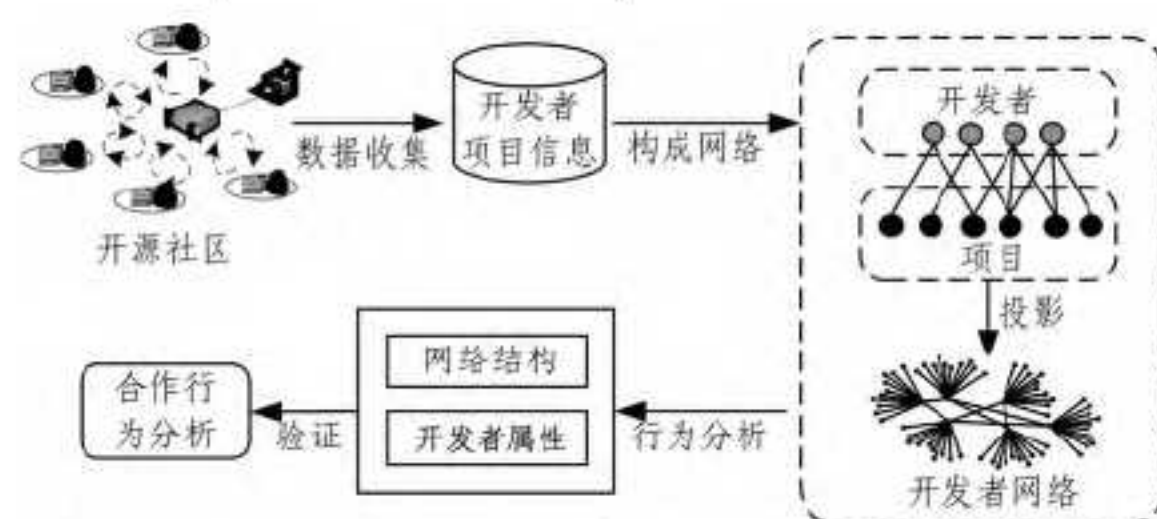


图1 整体研究框架

2.1 数据收集

本文以SourceForge(SF)开源平台为研究对象,从Flossmole网站^[22]提供的数据库导出9张SF项目信息表(时间跨度从Feb 2007—Dec 2008)用于分析,分别为:开发者表、项目表、开发者_项目表、项目主题表(Topic属性)、项目状态表(State属性)、项目操作系统表(OS属性)、项目开源认证表(Lisence)、项目开发语言表(Program-Language属性)和项目目标受众表(Intended-audience属性)。使用这些表中存储的项目属性信息来刻画每个开发者在社区中的实践技能情况。

为分析两个开发者加入社区的时间间隔对合作的影响,爬取每个开发者的加入时间。在数据处理过程中,将属性信息存在缺失的项目给予排除,另外针对只参加了一个项目的开发者和只有一个开发者的项目也不予考虑。

2.2 开发者合作网络

根据开发者与项目之间的参与关系,假设参与同一个项目开发的两个开发者之间存在一条合作连边,从而构建开发者合作网络,其中节点表示开发者,边为合作关系,合作次数为边的权重,简称为边权。Flossmole网站提供的是对SF社区项目每隔两个月导出的数据,即2007年2月、4月、6月,直到2008年12月。为分析已有开发者间的合作,将得到的12个时间段数据进行处理,生成12个开发者网络。图2为一个简单的开发者网络,两个月后开发者3和4之间的合作由无到有,而4和8,5和6之间再次建立新合作。

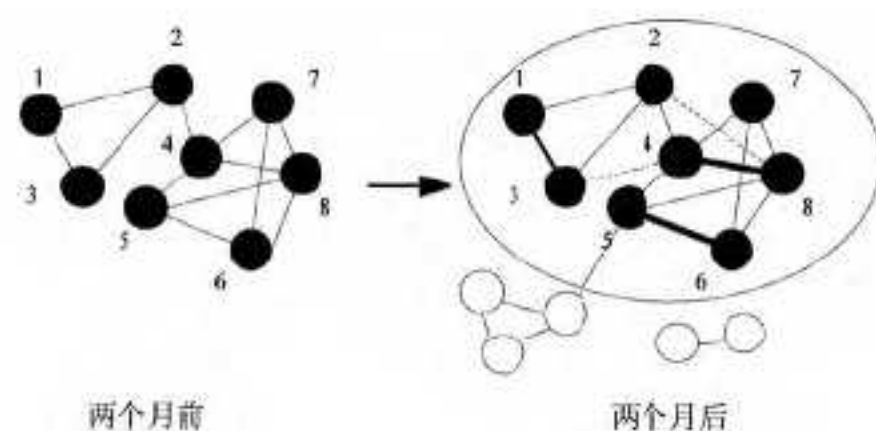


图2 一个开发者网络实例

表1为开发者网络的统计信息,其中 $|Node_s|$ 代表第一个时间段(Feb 2007,简化为0702)构建的开发者网络中的节点总数, $|Edge_s|$ 为对应开发者网络的边数。此处 $s(s=1)$ 和 $e(e=12)$ 分别用来表示12个开发者网络中的第一个和最后一个网络。

表 1 开发者合作网络的统计信息

$ Node_s $	$ Edge_s $	$ Node_e $	$ Edge_e $
10621	69498	11949	86992

2.3 合作行为分析

在开源社区中,一个开发者网络中已有开发者间存在两种模式的合作(见图 2):一种是两个未曾合作过的开发者首次合作(如节点 3 和 4),另一种是两个已经合作过的开发者之间再次合作(如节点 4 和 8,5 和 6)。为了便于表达,分别将这两类合作关系分别表示为 O1O 和 O2O。

在社交网络中,“朋友的朋友”之间倾向于成为朋友。在开发者网络中,具有这种间接关系的两个开发者间是否也倾向于建立合作(O1O)? Hahn 等人^[15]验证了开发者之前的合作连边会影响团队的形成,那么,两个开发者之前的合作次数是否会促进他们再次合作(O2O)? 我们将从网络拓扑结构对以上两个问题进行分析。

在复杂网络中,BA 模型^[24]常用来生成无标度网络,这类网络的演化过程具有带偏好连接特性。根据之前的工作^[23],发现开发者合作关系的建立一定程度上也具有偏好连接特性。因此,社区中开发者 i 连接任何一个开发者 j 的概率可表示为:

$$p_{ij} = \frac{k_j}{\sum_m k_m} \quad (1)$$

其中, k_j 为开发者 j 的度, m 为所有开发者个数。需要注意的是,由于合作是相互的,因此开发者 i, j 之间的偏好合作概率应该为两者偏好合作概率的最大值,即 $\max(p_{ij}, p_{ji})$ 。

考虑到开发者自身的属性,利用 2.1 中提到的 6 个属性刻画开发者在社区中的技能(expertise)水平,数学表示为 $Exp = (AT_p, AT_s, AT_t, AT_{pl}, AT_{is}, AT_{lic})$ 。如果两个开发者具有相似的技能,他们之间建立合作的可能性会更大。使用 AT 表示开发者的某一个技能属性,如 $AT_{pl} = \{pl_1, pl_2, \dots, pl_n\}$ 表示开发者的开发语言技能属性,其中 pl_i 为开发者使用过的第 i 类语言,因此,两个开发者 A, B 在开发语言技能上的相似性可表示为:

$$Sim(AT_{pl}^A, AT_{pl}^B) = \frac{AT_{pl}^A \cap AT_{pl}^B}{AT_{pl}^A \cup AT_{pl}^B} \quad (2)$$

3 案例分析

问题 1: 开发者合作网络的拓扑结构是否影响社区已有开发者的合作行为?

在开发者合作网络中,“朋友的朋友”关系等价于两个相距跳数为 2(hop=2) 的节点之间的关系,如图 2 中的节点 4 和节点 6 的关系。首先分析了其对 O1O 合作的影响,图 3(a) 显示基于“朋友的朋友”关系建立合作关系的比例比较低,严格来讲,与其他社交网络不同,开源社区中两个未曾合作过的开发者更可能自由选择合作对象,而不是依赖间接(两跳)的合作关系。这种现象的产生可能与开源社区本身的自组织、松散合作的特性有关,另外,一般的社交网络都提供相应的推荐功能,而在开源社区中还没有类似服务,从而一定程度上并没有发挥这种间接关系的作用。因此,开源社区中拓扑距离优势对未曾合作过的已有开发者间的影响不明显。

对于 O2O 合作,主要调查两个开发者之间边权(之前合作的次数)对他们的再次合作的影响。图 3(b) 表示已有开发

者间再次合作的比例受边权的影响,整体上合作次数越多的开发者们再次合作的比例越大,尤其当边的权重超过 10 时。所以已有开发者之前的合作经验有利于促进他们之间再次合作。此外,还分析了合作网络中开发者的度_度相关性,除了第二个开发者网络,其他几个开发者网络中开发者的度_度相关性均为正,表明度大的开发者更倾向于与其他度大的开发者建立合作,其中有 9 个网络呈现强相关性(相关系数大于 0.6),可对应为社会网络中的“强强合作”关系(见图 3(c))。

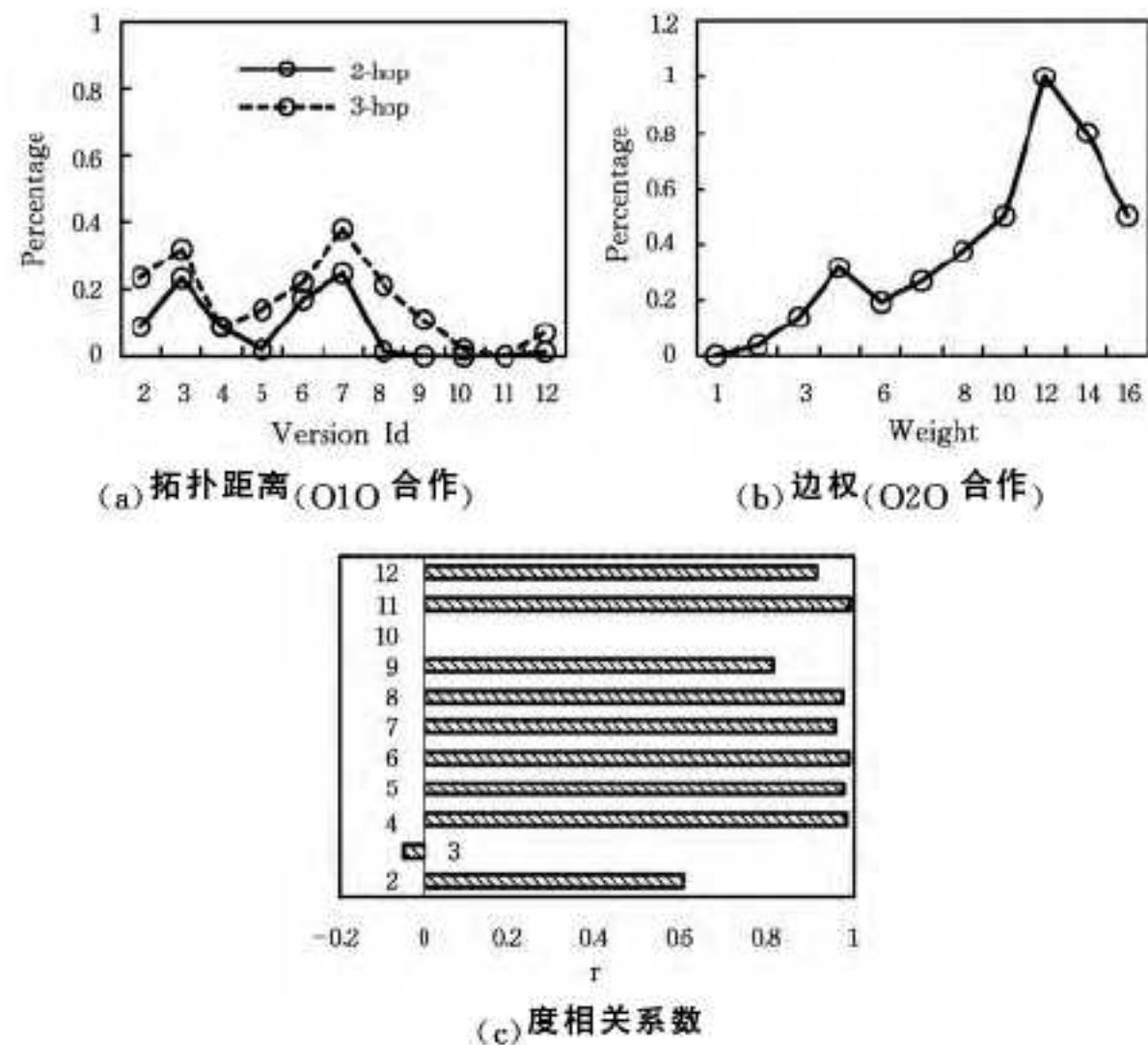


图 3 开发者网络指标的变化趋势

综上所述,拓扑距离优势对已有开发者建立首次合作的影响不明显,不同于常规的社交网络,而之前的合作经验可有效促进已有开发者间再次合作。

问题 2: 开发者的技能属性是否影响社区已有开发者的合作行为?

问题 1 从网络结构角度分析已有开发者的合作行为,问题 2 将重点从开发者自身的属性分析他们的合作行为。在开源社区中,找到具有特定技能的合作对象也是一个关键问题。令一个开发者在项目中为管理员,则他的管理员经验指标 Is_admin 为 Y, 否则为 N, 所以 YY 表示一次合作的两个开发者都有管理员经验, YN 表示其中有一个满足, NN 表示都没有管理员经验。首先,从开发者是否具有项目管理员经验指标出发,图 4 显示一半以上的新合作都是建立在有一个开发者是项目管理员的情况下,其次为都有项目管理员经验的两个开发者之间,其中至少有一个开发者有管理员经验的比例占 90% 以上。对于 YY 和 YN 两种类型的合作,相对于 O2O 合作, O1O 合作的比例明显更大,分别为 79% 和 78%, 表明开发者的管理员经验对合作有影响。

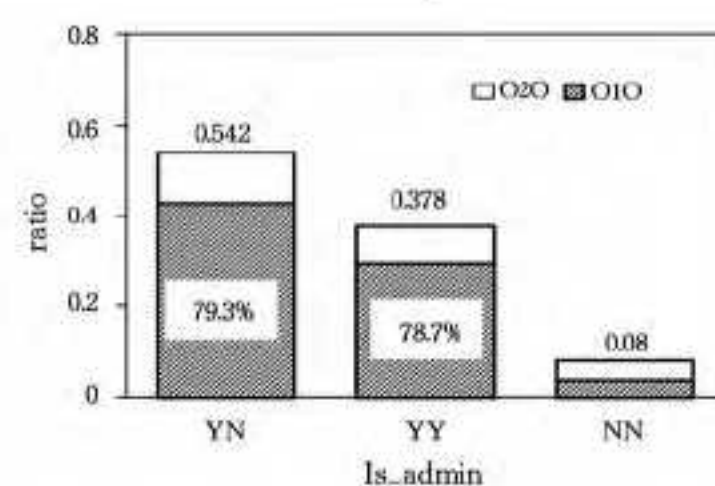


图 4 两类合作的开发者在管理员经验属性上的分布

其次,统计了其中 8 个属性指标值在两类合作模式下的分布差异。在图 5 中,很明显角色、操作系统、开发语言等几

个属性在两类合作上有明显的分布差异。例如对于角色属性, O2O 合作的开发者间角色属性相似度的中位数明显比 O1O 合作的开发者间角色属性相似度大 ($0.685 > 0.160$), 即一个已有开发者应该优先与跟他具有相似角色属性的其他开

发者合作。然而, 两个开发者加入社区的时间间隔大小对两类合作的影响刚好相反, 其中 O1O 的中位数为 2.28, O2O 只有 1.29。一种可能的解释是在社区中后期加入的开发者会倾向于与较早加入的开发者建立合作关系。

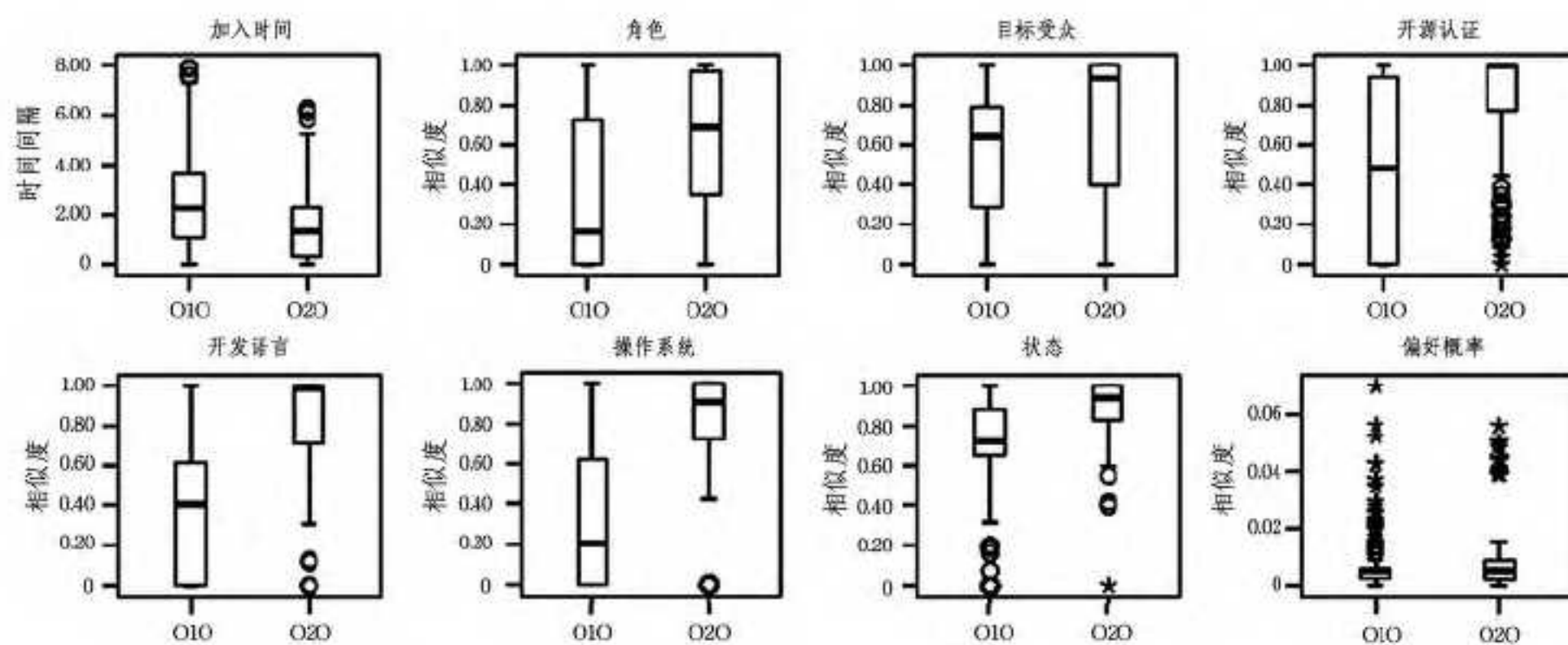


图5 关于开发者两类合作

综上所述, 开发者的管理员经验以及项目开发环境(开发语言和操作系统)对已有开发者间的合作影响显著, 而合作开发者加入的时间间隔的长短对其影响不大。

4 讨论与不足

我们发现, 相对于一般的社交网络, 开发者合作网络中“朋友的朋友”关系对已有开发者间建立新的合作没有明显影响。开源社区是一个专业性比较强的虚拟社会网络, 社区本身的自组织性与松散合作性可能使得它与一般的社交网络不同。另外, 在收集的数据集时间段, 开源平台上并没有提供任何类似于社交网络上的推荐功能, 而且在构建合作网络时对一些只参与一个项目开发的开发者进行了过滤, 所以这也可能是导致该结果的一种原因。从实践的角度, “双赢”合作方式为理解社区开发者的合作行为与激励社区开发者合作提供依据。在软件工程推荐系统领域, 这种“双赢”合作方式可为目标开发者推荐潜在的合作对象。另外, 根据开发者合作网络中合作边权对合作的影响, 发现有合作经历的两个开发者更可能在以后的开发中建立合作。他们之间的合作一定程度上也将提高软件开发质量。

另外, 项目管理员不仅具有更高的资源访问权限, 而且管理员的声誉直接影响新成员的加入, 这也是有更多合作建立在具有管理员经验的开发者之间的原因。社区管理员角色的争取在一定程度上鼓励开发者变得更加“勤快”。开发者熟练的开发语言与习惯的操作系统环境直接关系到所开发的项目在社区的受欢迎度, 所以社区开发者会优先选择与那些具有相同技能属性的开发者建立合作。

本文分析了社区已有开发者的合作行为, 发现了一些影响合作的因素, 并进行了验证。但文章也存在一些不足之处: (1) 只选取了 2 年的数据集作分析, 近年来随着社交网络的流行, 社区中部分开发者可能会通过社交平台交互, 使得在开发者网络上并没有合作的开发者实际上有合作关系, 这使得文中部分结论可能受到考验; (2) 使用了一些典型的定性、定量指标分析开发者的合作行为, 在社会网络分析方法中还有很多其他指标可供使用, 以及在刻画开发者技能属性时没有考虑其他更多的属性, 更多的指标与属性有待考虑, 进而验证与丰富实验结果; (3) 除了 Sourceforge 社区, 还有一些比较流行

的如 GitHub 和 Google code 社区, 在这些社区中已有开发者是否同样具有相同的合作行为也有待进一步的验证。

结束语 本文主要目的是理解开源社区中已有开发者是如何选择合适的合作对象建立新的合作以及分析哪些因素可能会影响他们的合作行为。以 Sourceforge 平台为实验对象, 从开发者网络拓扑结构与开发者的技能属性角度进行分析。实验结果表明, 拓扑距离上的优势(即“朋友的朋友”关系)对社区中未曾合作的已有开发者间建立合作的促进作用很小, 而已有开发者间之前合作次数以及度数较高的开发者间均倾向于再次合作(即“熟人”合作与“双赢”合作)。另外, 我们发现开发者是否具有管理员经验与开发环境(开发语言和操作系统)属性的相似性对两个已有开发者建立合作有明显的促进作用。

参考文献

- [1] Ye Y, Kishida K. Toward an Understanding of the Motivation of Open Source Software Developers[C]// Proceedings of 25th Int'l Conf. on Softw. Eng. (ICSE). 2003:419-429
- [2] Madey G, Freeh V, Tynan R. The Open Source Software Development Phenomenon: An Analysis Based on Social Network Theory[C]// Proceedings of the Americas Conf. on Information Systems (AMCIS). Dallas, 2002:1806-1813
- [3] Lee S Y T, Kim H W, Gupta S. Measuring open source software success[J]. Omega, 2009, 37(2):426-438
- [4] Crowston K, Howison J. Defining Open Source Software Project Success[C]// Proceedings of the Int'l Conf. on Information System. 2003
- [5] Roberts J A, Hann I H, Slaughter S A. Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects[J]. Management Science, 2006, 52(7):984-999
- [6] 何鹏, 李兵, 杨习辉, 等. 开源软件社区开发者偏好合作行为研究[J]. 计算机科学, 2015, 42(2):161-166
- [7] Ngamkajornwivat K, Zhang D, Koru A G, et al. An Exploratory Study on the Evolution of OSS Developer Communities[C]// Proceedings of the 41st Annual Hawaii Int'l Conf' on System Sciences(HICSS). 2008:305-305

(下转第 501 页)

结束语 应用程序中部分软件故障的产生与变量的状态有关,为检测程序中的这些错误,需要对应用程序进行数据流分析以确定变量之间的定值与引用关系。数据流分析可以通过程序代码的静态分析进行,或者通过程序的动态运行得到。由于程序动态运行不能覆盖程序的所有分支,会对完整性分析产生影响,因此,本文提出了静态的数据流分析方法。该方法从程序的控制结构出发,充分考虑了程序中分支结构对数据流的影响,可以准确反映程序动态运行中的数据流状态,为软件故障的检测提供有力的支持。

参 考 文 献

- [1] Hardekopf B, Lin C. Flow-sensitive pointer analysis for millions of lines of code[C]//International Symposium on Code Generation and Optimization, 2011. Chamonix, France, IEEE, 2011: 289-298
- [2] Kooli M, Bosio A, Benoit P, et al. Software testing and software fault injection; Design & Technology of Integrated Systems in Nanoscale Era (DTIS), 2015[C]//Naples. IEEE, 2015: 1-6
- [3] Yu Hong-tao, Xue Jing-ling, Huo Wei, et al. Level by level: making flow-and context-sensitive pointer analysis scalable for millions of lines of code[C]//Proceedings of the 8th annual IEEE/ACM International Symposium on Code Generation and Optimization, 2010. New York, USA; 2010: 218-229

- [4] Li L, Cifuentes C, Keynes N. Precise and scalable context-sensitive pointer analysis via value flow graph[J]. ACM Sigplan Notices, 2013, 48(11): 85-96
- [5] Denaro G, PezzAl M, Vivanti M. On the right objectives of data flow testing[C]//Proceedings of the IEEE International Conference on Software Testing, Verification, and Validation, 2014: 71-80
- [6] Sui Y, Ye S, Xue J, et al. Making context-sensitive inclusion-based pointer analysis practical for compilers using parameterised summarisation[J]. Software Practice & Experience, 2014, 44(12): 1485-1510
- [7] Nair S, Jetley R, Nair A, et al. A static code analysis tool for control system software[C]//IEEE 22nd International Conference on Software Analysis, Evolution and Reengineering, 2015: 459-463
- [8] Khan M E. Different approaches to white box testing technique for finding errors[J]. International Journal of Software Engineering and Its Applications, 2011, 5(3): 1-11
- [9] Khan S A, Nadeem A. A tool for data flow testing using evolutionary approaches (etodf) [C] // International Conference on Emerging Technologies (ICET). 2013: 1-6
- [10] 严蔚敏, 吴伟民. 数据结构[M]. 北京: 清华大学出版社, 2011: 180-183

(上接第 479 页)

- [8] Bird C, Gourley A, Devanbu P, et al. Open Borders? Immigration in Open Source Projects[C]//International Conference on Software Engineering Workshops, 2007: 6
- [9] Beecham S, Baddoo N, Hall T. Motivation in Software Engineering: A systematic literature review[J]. Information and Software Technology, 2008, 50(9): 860-878
- [10] Krogh G, Spaeth S, Lakhani K R. Community, joining, and specialization in open source software innovation: a case study[J]. Research Policy, 2003, 32(7): 1217-1241
- [11] Fronza I, Sillitti A, Succi G. An interpretation of the results of the analysis of pair programming during novices integration in a team[C]//Procs of the 3rd Int'l Symposium on Empirical Softw. Eng. and Measurement, IEEE Computer Society, 2009: 225-235
- [12] Herraiz I, Robles G, Amor J J. The processes of joining in global distributed software projects [C] // Proceedings of the Int'l Workshop on Global Softw. Dev. for the Practitioner, 2006: 27-33
- [13] Dagenais B, Ossher H. Moving into a new software project landscape[C]//Proceedings of Int'l Conf. on Softw. Eng. (ICSE), 2010: 275-284
- [14] Ducheneaut N. Socialization in an Open Source Software Community: A Socio-Technical Analysis[J]. Computer Supported Cooperative Work, 2005, 14(4): 323-368
- [15] Hahn J, Moon J Y, Zhang C. Emergence of New Project Teams from Open Source Software Developer Networks?; Impact of Prior Collaboration Ties[J]. Information Systems Research, 2008, 19(3): 369-391
- [16] Toral S L, Martínez-Torres M R, Barrero F. Analysis of virtual communities supporting OSS projects using social network anal-

- ysis[J]. Information and Software Technology, 2010, 52(3): 296-303
- [17] Crowston K, Wiggins A, Howison J. Analyzing Leadership Dynamics in Distributed Group Communication[C]//Proceedings of the Annual Hawaii Int'l Conf' on System Sciences(HICSS), 2010: 1-10
- [18] Hossain L, Zhu D. Social networks and coordination performance of distributed software development teams[J]. The Journal of High Technology Management Research, 2009, 20(1): 52-61
- [19] Datta S, Sindhgatta R, Sengupta B. Evolution of Developer Collaboration on the Jazz Platform: A Study of a Large Scale Agile Project[C]//Proc. of the 4th India Softw. Eng. Conf. (ISEC), 2011: 21-30
- [20] Huang S K, Liu K M. Mining Version Histories to Verify the Learning Process of Legitimate Peripheral Participants[C]//The 2nd Int'l Workshop on Mining Softw. Repos(MSR), 2005: 1-5
- [21] Allaho M Y, Lee W C. Analyzing the Social Ties and Structure of Contributors in Open Source Software Community[C]//Proceedings of Int'l Conf. on Advances in Social Networks Analysis and Mining, 2013: 56-60
- [22] Howison J, Conklin M, Crowston K. FLOSSmole: A collaborative repository for FLOSS research data and analyses[J]. International Journal of Information Technology and Web Engineering, 2006, 1(3): 17-26
- [23] He P, Li B, Huang Y. Applying Centrality Measures to the Behavior Analysis of Developers in Open Source Software Community[C]//International Conference on Cloud and Green Computing (CGC), 2012: 418-423
- [24] Albert R, Barabási A L. Statistical mechanics of complex networks[J]. Reviews of Modern Physics, 2002, 74(1): 47-97