arXiv:2304.05766v1 [cs.SE] 12 Apr 2023

我们生活在一个社会中：OSS 生态系统中的贡献动机

Supatsara Wattanakriengkrai、RaulaGaikovina Kula、Christoph Treude、Kenichi Matsumoto奈良先端科学技术大学院大学，日本，墨尔本大学，澳大利亚邮箱：{wattanakri.supatsara.ws3,raula-k, matumoto}@is.naist.jp, christoph.treude@unimelb.edu.au

摘要—**由于针对开源库生态系统的攻击数量不断增加，协助维护者已成为首要任务。这一点尤其重要，因为维护者通常不堪重负。尽管人们已经广泛研究了开源开发者的动机，但维护者在多大程度上协助其所依赖的库，尚不得而知。通过对NPM开发者开展调查，我们的早期结果表明，在为软件生态系统做出贡献时，维护自己的库与职业因素之间存在动机差异：前者更多由个人驱动，而后者则注重技能和专业知识。最后，我们的主题分析显示，开发者在为生态系统做出贡献时面临着不同的动机和障碍。这些结果表明，开发者在为生态系统做出贡献时，根据其所扮演的角色不同，会有不同的动机和障碍。**

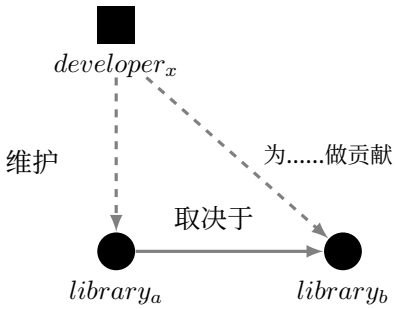

图 1. 生态系统中的不同类型贡献

> “开源软件可以被描述为经历三个角色的过程：作者、贡献者、维护者。这些角色的职责、动机等截然不同。人们非常重视开源软件的创作与贡献，却很少关注那些维系整个开源软件生态系统并推动其发展的维护者。
>
> 参与者 19

一、引言

第三方库对于现代软件应用至关重要，不仅对开源软件如此，在工业环境中也是如此 [1]。然而，日益加剧的不稳定威胁引发了人们对库维护者工作负担的担忧。例如，谷歌和微软最近宣布了多项举措，如 Alpha-Omega 项目，以提升全球开源软件供应链安全。他们表示“*Alpha* 将与最关键的开源项目的维护者合作，帮助他们识别并修复安全漏洞，提升其安全态势。*Omega* 将找出至少 *10,000* 个广泛部署的开源软件项目，对其进行自动化安全分析和评分，并为其开源维护者社区提供修复指导。” [2]。维护者挫折感（log4Shell） [3]甚至劫持（faker、colors、rc） [4]，也可能导致不稳定，其中涉及维护者蓄意破坏。

人们普遍认为，典型的生态系统如同一个社会：库的维护者应在生态系统内接收并提交外部贡献，因为人们认为他们有着相似的目标和动机。吸引此类外部贡献，有助于确保贡献源源不断（即修复缺陷、添加新功能）。

由于大多数第三方库依赖志愿者贡献（通常无偿且工作过度），这些贡献者的离开会给这些库的可持续性带来潜在风险 [5]。例如，Valiev 等人 [6] 表明，贡献者较多”的库不太可能进入休眠状态。Wattanakriengkrai 等人 [7] 也指出，库确实依赖生态系统提供贡献，而且这些贡献与库进入休眠状态的可能性呈负相关。此外，先前研究表明，库的可持续性不仅受库内部因素影响，也受生态系统因素影响 [8]，，例如依赖关系 [9]。具体而言，漏洞 [10, 11] 和传递依赖变更 [12] 等威胁可能使库面临过时的风险[13]。其他先前研究发现，这些生态系统形成了复杂的依赖网络，各个库彼此依赖。Wittern 等人 [14] 分析了部分 NPM 库，并指出 NPM 是一个动态的生态系统，其中库持续快速增长，彼此之间的依赖关系也日益增多。Abdalkareem 等人 [15] 观察到，NPM 中的简单库依赖大量库；如果这些简单库实现完善且经过测试，依赖它们既有用，风险也很低。对其他库的依赖往往处于脆弱的依赖链中，这意味着单个库的局部可持续性问题可能造成广泛的网络后果 [16]。这些研究只关注技术层面的依赖关系，并未聚焦于这些生态系统中存在的社交网络和人为因素。

本文的新思路是从不同角度理解开发者动机，**探讨是什么促使开发者为生态系统做出贡献（即为库做出贡献**

Motivations	○ Why do you contribute to other NPM libraries, i.e., libraries which you do not own? ○ What is the reasoning behind choosing the other library to maintain/contribute to?
Expertise and Barriers	○ Which factors do you consider when maintaining/contributing to NPM libraries? ○ Please leave your comments or thoughts especially on any factors that we did not include. ○ What are some reasons to slow down/stop or increase contributions to own/other libraries? ○ Please leave your comments or thoughts especially on any other reasons that we did not include.

图 2. 按动机、专业知识和障碍分类的定性调查问题

，也就是他们所依赖的库），而不是维护自己的库。图 1 展示了客户端与库之间的关系：某个库（$library_a$）的维护者（$developer_x$）依赖另一个库（$library_b$），并为其依赖项做出贡献。

我们对在 GitHub 上托管项目的 NPM 开发者开展了一项调查。我们选择了 NPM 生态系统，因为它是规模最大的生态系统之一，拥有超过两百万个库，而且如上文所述，近期曾遭到攻击。我们调查了 49 名参与者，他们都是在 GitHub 上托管 NPM 库的维护者。通过主题分析，我们的初步结果表明，有四个主题改变了开发者的动机。有趣的是，我们发现，开发者在为开源软件生态系统做出贡献时，其动机可能与为自己的库做出贡献时有所不同。

II. 个体动机观点

现有研究对开源软件项目贡献者的动机进行考察时，通常采用一般化且以个人为中心的视角。例如，Lee 等人 [17] 研究了一次性代码贡献者的动机，发现开发者做出贡献主要是为了修复妨碍其工作的漏洞。同样，Coelho 等人 [18] 报告称，核心开发者为开源软件项目做出贡献的主要原因是改进项目，因为他们自己也在使用该项目。Gerosa 等人 [19] 最近对开源软件开发者动机开展的一项研究表明，学习和智力激发等内在动机与内化动机对许多开发者都十分重要。

专业知识可被视为驱动开发者为开源软件项目做贡献的因素。例如，Vadlamani 等人 [20]对软件开发者的专业知识进行了定性研究，以了解驱动开发者为开源软件项目做贡献的因素。他们发现，在激励 GitHub 开发者方面，个人驱动因素（即自我需求和业余爱好）比职业因素（即技能和专业知识）更为重要。

近期研究也承认新核心贡献者面临障碍。例如，Avelino 等人 [21]调查了主要开发者弃置开源软件项目的现象。作者表明，项目能够通过吸引新核心贡献者来渡过此类困境；然而，新核心开发者在为这些项目做贡献时会遇到各种障碍，例如时间不足，以及难以获得代码仓库的推送权限。

III. 研究方法

在本节中，我们介绍本研究的问卷设计、招募和分析方法。

A. 问卷设计与招募

在获得匿名研究机构的伦理审查委员会批准[1] 后，我们通过电子邮件联系了开发者，问卷数据可在线获取[2]。按照先前的研究 [10],，我们通过一份精心整理的开发者名单收集并发送邀请。在向广泛的参与者发放问卷之前，我们开展了试点调查，以确认参与者理解问卷内容；他们的反馈帮助我们重新措辞了部分问题。最终版问卷包含以下问题：询问参与者的人口统计信息（经验年限、技能水平，以及他们是否通过贡献来提升技能）后，我们提出了一组与贡献相关的问题，询问参与者在自己的库和其他库中的活动（贡献偏好、库的数量、贡献频率、库的类型以及贡献的类型）。

图2详细介绍了我们向参与者提出的定性问题。我们将问题分为两部分，并允许参与者补充评论和其他观点。第一部分涉及维护者为其他库做出贡献的动机。对于定量部分，我们设置了五个选项（发现缺陷、依赖该库、新功能、感兴趣的问题以及对贡献者的熟悉程度）。这些选项基于 Subramanian 等人 [22]。第二部分用于询问维护者在做出贡献时面临的专业知识和障碍。第一道问题采用了 Vadlamani 和 Baysal[20] 所提出的相同驱动因素。如前所述，个人因素更多与自我需求和爱好有关，而职业因素则更多与其专业技能和专业知识相关。对于第二道问题，我们

[1]伦理审批文件将提供[2]作者将在接受后提供链接

Demographics		
Years of experience in Node.js	2-3 years	10%
	4-5 years	31%
	6-7 years	22%
	8-9 years	16%
	10 years or more	21%
Developer skill level with JavaScript or Node.js	general skills	24%
	specialized skills	72%
	others	4%
Means to improve developer skills	contributions in GitHub	47%
	others	53%

Participation in NPM libraries		
Skill level of contributions	contributions match skill level	44%
	contributions can improve skills	43%
	others	13%
Preference for contributions	maintain > contribute	63%
	maintain < contribute	6%
	maintain = contribute	31%
Number of libraries	Maintain	Contribute
Less than 10 libraries	57%	84%
10-20 libraries	20%	8%
21-30 libraries	4%	4%
31-40 libraries	6%	2%
51 libraries or more	13%	2%
Frequency of contributions	Maintain	Contribute
Less than 2 times	61%	92%
2-3 times	14%	2%
4-5 times	4%	4%
6-7 times	4%	0%
8-9 times	3%	0%
10 times or more	14%	2%
Kinds of libraries	Maintain	Contribute
Development (help with beta)	30%	17%
Production (stable)	31%	46%
In active (revive)	10%	11%
No matter	29%	26%
Kind of contributions	Maintain	Contribute
Documentation	17%	19%
New features	19%	20%
Bug	18%	26%
Refactoring	18%	11%
GIT related issues	12%	8%
Test cases	16%	16%

图 3. 49 位 Node.js 开发者的回答概览。

根据 Vadlamani 和 Baysal [20] 列出了四项障碍（库的稳定性、时间不足、动力以及库生命周期的结束）。

B. 人工编码

为了分析对定性问题的回答（图2），我们采用了与以往研究类似的卡片分类方法 [23],[24]。编码采用并扩展了 Gerosa 等人提出的内在和外在分类法 [19]。具体过程如下。首先，指定前两位作者执行卡片分类任务。他们审阅了所有回答，并为每条回答制作一张“卡片”。为进行验证，两位作者没有分别并行编码卡片并检查编码结果的一致性，而是共同为卡片编码 [25],[26]。编码人员采用开放式编码方法 [24]分析卡片，在编码过程中不断涌现新的主题。编码人员在过程中协商达成一致 [27],，也就是说，当编码人员意见不同时，我们会中断这一过程

Motivations	
	Contribute
Found a bug	30%
Depend on	28%
New feature	21%
Issue of interest	10%
Contributor familiarity	8%
Others	3%

Expertise drivers		
	Maintain	Contribute
Personal drivers	38%	36%
Professional drivers	35%	48%
Challenges	27%	16%

Barriers		
	Maintain	Contribute
Library stability	13%	20%
Lack of time	36%	32%
Motivation	28%	26%
End of library lifecycle	23%	22%

图 4. 维护者的动机、专业知识和障碍。

先讨论这一差异，再继续。

C. 受访者人口统计特征

图3展示了49位参与者的人口统计特征。我们发现，参与者的经验涵盖资深人士（拥有十年或以上经验，占21%）、中级人士（拥有四至九年经验，占69%）和初学者（拥有两至三年经验，占10%）。大多数参与者表示自己具备专业技能。有趣的是，参与者并未特别利用自己的贡献来提升技能水平。

IV. 定量研究结果

当被问及贡献偏好时，参与者更倾向于维护自己的库（即 63%）。图 3 还展示了参与者如何在维护自己的库和为生态系统中的其他库做贡献之间分配时间。谈到贡献，参与者大部分时间都花在维护自己众多的库上，而非为其他库做贡献，这并不令人意外（即每周超过两次：39% 的时间用于维护自己的库，8% 的时间用于为其他库做贡献）。

发现1 开发者花更多时间维护自己的代码（即 63 %），而非为生态系统做贡献（6%）。

在为自己的库或其他库做贡献时，参与者通常专注于已投入生产且稳定的库（即生产环境中的库；维护自己的库占 31%，为其他库做贡献占 46%）。不过，为其他库做贡献时，参与者倾向于优先修复漏洞（26%），其次是

表 I 我们编码得出的主题

主题	参与者	回复数量
自用客户	P2、P7、P8、P13、P16、P14、P20、P22、P26、P29、P30、P31、P32、P33、P34、P35、P36、P38、P41、P45、P49	21
通用修复/功能	P10、P11、P17、P24、P27、P37、P39、P40、P42、P46、P47	11
一般自利	P3、P5、P9、P15、P21、P23	6
归属感／利他主义	P1、P4、P6、P12、P18、P19、P25、P28、P43、P44、P48	11

添加新功能（20%）和编写文档（19%）；而在维护自己的库时，参与者通常专注于添加新功能（19%）以及维护工作，例如重构、编写文档和编写测试用例（分别占 18%、17% 和 16%）。

图 4 展示了有关动机、专业知识驱动因素和障碍的调查问题的定量回答。总的来说，我们看到，维护自己的库更多是由个人驱动因素（38%）推动的，而对其他库的贡献则更多由职业驱动因素（48%）促成。此外，大家一致认为，缺乏时间是维护其他库并为其做出贡献时最常被提及的障碍。

发现2 维护自己的库是由个人驱动因素（38%）推动的，而对生态系统的贡献则由职业因素驱动（48%）。

V. 定性研究发现

我们识别出了以下主题及其动机和障碍，如表 I 所示。

A. 自用客户 - 修复其明确声明所依赖的库中的问题的开发者。

源自 Gerosa 等人[19], 的自用 分类，自用客户 动机是最常见的主题，共有 21 条回答。正如 Gerosa 等人[19], 所解释的，这是一种内化的外在动机，即一种基于外部激励和干预措施的自我调节行为。在这种情况下，开发者为其他库做出贡献，是因为他们想修复自己所依赖的库中的问题。这种倾向也体现在定量回答中：如图 4 所示，它以 28% 的占比位居第二。

这个主题的一个例子是，有某些事情阻碍了开发者的工作：

> “当它们出了问题、需要修复时（而这.....影响到我的工作时），通常我就会介入。” - 参与者 2

在下一个示例中，参与者明确表示，动机是修复开发者在其应用程序中所依赖的库（软件包）中的缺陷。示例如下：

> “我有时会修复自己使用的软件包中的缺陷，或对其进行改进” -参与者33

其他原因则更多源于维护者自身，促使其协助处理现有问题。以下是一个例子：

> “有时，我依赖的其他软件包会出现问题，而维护者没有足够的精力来修复。” - 参与者 31

B. 通用修复/功能——通过修复库中的问题或为其添加新功能来作出贡献，但并未明确说明自己对这些库的依赖关系的开发者。

与第一个主题类似，通用修复/功能主题基于自用动机。这里的重点是明确修复其他库中的漏洞，而他们并非这些库的**客户端**（11 位受访者）。这种动机的外在性较弱，更多是为了帮助其他维护者。如图 4 所示，我们发现，受访者认为发现漏洞是为其他库作出贡献时最常见的动机（即 30%），例如：

> “因为我遇到了其中一些需要修复的问题，而最简单的方式就是自己修复它，并将更改贡献回去。” - 参与者 17

该主题的另一个例子是，参与者表示需要修复自己遇到的任何类型的缺陷。在这种情况下，参与者表达了以下需求：

> “我会尽力修复遇到的漏洞” - 参与者 24

其他回答则具体涉及修复库中的某些部分，或引入新功能来满足他们自己的需求：

> “如果它们在某些方面有问题，或者我需要添加一项功能来提升易用性。” - 参与者47

C. 一般自利 -因个人原因做出贡献的开发者，例如解决自己的需求、想要某些东西，或为了自身利益。

第三个主题也与一般的外在动机有关，这正是人们对开源开发者的普遍刻板印象。例如，开发者出于自身利益而贡献代码。示例包括：

> 自身需求。某些事情阻碍了我继续推进，而这个改变需要由他们那边来完成。即便如此，我通常也会尝试绕开问题，而不是提交贡献。- 参与者5

另一个例子是，开发者觉得有必要向生态系统中的另一个库建议一项更改。下面展示了这个例子：

> “通常是因为有一些我希望他们能拥有的改动”- 参与者 21

在基于自身利益的贡献障碍方面，我们还发现，开发者有时会认为，只需通过分叉就能满足相当多的自身利益需求：

> “很多开发者会分叉软件包并修改它，只为自己使用。他不会提交拉取请求，也不会向软件包所有者提出问题。他只是为自己、为自己的业务而做。”- 参与者9

另一种障碍是，开发者因自己担任维护者所承担的繁重职责而感到倦怠。例如：

> “多年来，我独自维护一个拥有数千个星标的项目，已经精疲力竭。”- 参与者20

D. 归属感／利他主义——开发者出于对社区的归属感或责任感而做出贡献。

与其他三个主题不同，这一主题源于内在动机，体现了开源维护者的本质。按照定义，内在动机会促使人们为了其中的乐趣或挑战而行动，这与前面三个主题有所不同。示例如下：

> “回馈社区”。- 参与者4
>
> “这与我创建自己的软件包出于同样的原因：更好地为整个生态系统解决问题。”- 参与者43

谈到亲缘关系方面的障碍时，一位参与者提到，开源项目的成熟度可能会成为某人尝试做出实质性贡献的障碍：

> “这份问卷让我思考自己为什么（在*npm*生态系统中）贡献如此之少。这可能是因为我觉得（也许只是想象）提交拉取请求并为之争取的过程有官僚作风。很多时候，这只是在白费精力（不过这是基于企业软件包的情况；并非业余爱好者）。此外，如果代码仓库没有适当的测试，我就会更不想贡献，因为我觉得自己可能会毁了别人的一天……”- 参与者5

也有人对整个生态系统表达了类似的看法。在下面这个案例中，参与者表达了对生态系统规模之大的感受，其中一些部分可能包含类似的库。

> “*Node.js* 社区非常难以合作。人们不断重复发明同样的东西，这让新成员入门时感到困惑。”- 参与者 15

发现3： 我们确定了开发者为生态系统做贡献时的四种动机：（即自用客户、通用修复/功能、普遍的利己主义，以及亲缘关系／利他主义）。最普遍的贡献动机是解决开发者所依赖的库中的问题（即自用客户），而庞大的生态系统则被视为维护者回馈生态系统的潜在障碍。

六、有效性威胁

第一个威胁与调查参与者误解调查问题有关。为降低这一威胁，我们开展了试点调查，以确认调查参与者对问题的理解。此外，我们决定采用现有研究来涵盖所有动机（Subramanian 等人 [22] 和 Vadlamani 和 Baysal [20]）。最后一个威胁与我们的结果能否推广到其他库生态系统有关，而我们无法对此作出保证。未来，我们计划将研究扩展到其他生态系统（Python 的 PyPI 和 Java 的 Maven），以降低这一威胁。

> 问：你为什么会为其他 ***NPM*** 库做贡献，也就是那些并不属于你的库？“答。我们生活在一个社会中”
>
> 参与者28

七、结论与未来方向

作为一篇新思想论文，我们现指出关键启示和未来工作。

正如社会中个人的动机可能会因其角色而改变，生态系统中贡献者的动机也可能会根据其所扮演的角色而改变。发现3表明，客户端更有可能修复其所依赖的库，但也暗示了维护者在进行贡献时所面临的困难。发现2还表明，开发者所扮演的角色不同，其动机也会有所不同，无论其是该库的维护者还是用户。更好地理解这种关系，强化了我们的观点：我们生活在一个要求生态系统成员之间相互贡献的社会中。

将客户端视为其所依赖的库的潜在贡献者。 在开源软件生态系统的背景下，发现1和发现2表明，他们决定为生态系统做出贡献时，其动机存在差异。这些贡献有可能修复重要漏洞，并实现库可能需要的新功能。

未来计划。 这些结果为未来研究提供了许多可能的方向。一个重要的研究方向是开展更多实证研究，以更好地了解客户端对库的贡献，从而使我们能够将这些贡献与生态系统中其他参与者所作的贡献区分开来。并非所有客户端都一样——了解

不同的客户端与库之间的关系及其影响，是另一个重要的研究方向。

从人的角度来看，我们也可以研究生态系统中不同角色之间的互动与沟通，例如客户端与维护者在代码审查期间展开的讨论。此外，探究客户端为其所依赖的库作出贡献的动机与激励因素，或许能为鼓励这一群体更多地参与提供有价值的见解。

参考文献

[1] E. Larios Vargas、M. Aniche、C. Treude、M.Bruntink 和 G. Gousios，《第三方库的选择：从业者视角》，载于 *ESEC/FSE*，2020 年，第 245–256 页。[2] 《Alpha-omega——开源安全基金会》，https://openssf.org/community/alpha-omega/，（访问日期：2022 年06 月 16 日）。[3] 《Yfrytchsgd/log4jattacksurface》，https://github.com/YfryTchsGD/Log4jAttackSurface，（访问日期：2022 年 06 月16 日）。[4] 《npm 库“colors”和“faker”遭其维护者蓄意破坏以示抗议——现在该怎么办？——Security Boulevard》，https://securityboulevard.com/2022/01/npm-libraries-colors-and-faker-sabotaged-in-protest-by-their-maintainer-what-to-do-now/，（访问日期：2022 年 07月 12 日）。[5] E. Constantinou 和 T. Mens，《GitHub 中 Ruby 生态系统的社会技术演化》，载于*SANER*，2017 年，第 34–44 页。[6] M. Valiev、B. Vasilescu 和 J. Herbsleb，《开源项目持续活动的生态系统层面决定因素：PyPI 生态系统案例研究》，载于 *ES-EC/FSE*，2018 年，第 644–655 页。[7] S. Wattanakriengkrai、D. Wang、R. G.Kula、C. Treude、P. Thongtanunam、T.Ishio和 K. Matsumoto，《回馈：软件生态系统中与库依赖变更相一致的贡献》，*TSE*，第 1–13 页，2022 年。[8] C. Bogart、C. K¨astner、J. Herbsleb 和F. Thung，《如何破坏 API：三个软件生态系统中的成本协商与社区价值》，载于*FSE*，2016 年，第109–120页。[9] M. Golzadeh，《分析 Cargo 软件包依赖网络中的社会技术一致性》，载于 *FSE*，2019年。[10]B. Chinthanet、R. G. Kula、S. McIntosh、T. Ishio、A. Ihara 和K. Matsumoto，《npm 漏洞修复的发布、采用与传播滞后》，*EMSE*，第1–28 页，2021 年。[11] Z. Durumeric、F. Li、J. Kasten、J. Amann、J. Beekman、M. Payer、N. Weaver、D. Adrian、V. Paxson、M. Bailey 和 J. A. Halderman，《Heartbleed 事件》，载于*Proceedings of the 2014Conference on Internet Mea-surement Conference*，2014 年，第475–488 页。[12]T. Dey、Y.Ma 和A.Mockus，《基于 npm 生态系统参与模式的努力贡献、需求和用户分类模式》，载于 *PROMISE*，2019 年，第 36–45 页。[13] R. G. Kula、D. M. German、A. Ouni、T. Ishio 和 K. In-oue，“开发者会更新他们的库依赖吗？”*EMSE*，第 23 卷，第 1 期，第 384–417 页，2018 年。[14] E. Wittern、P. Suter 和 S. Rajagopalan，“JavaScript 包生态系统动态概览”，载于 *MSR*，2016 年，第 351–361 页。[15] R. Abdalkareem、O. Nourry、S. Wehaibi、S. Mujahid 和 E. Shihab，“开发者为何使用琐碎包？关于 NPM 的实证案例研究”，载于 *FSE*，2017 年，第 385–395 页。[16] R. Kikas、G. Gousios、M. Dumas 和 D. Pfahl，“包依赖网络的结构与演变”，载于*MSR*，2017 年，第 102–112 页。[17] A. Lee、J. C. Carver 和 A. Bosu，“理解自由/开源软件项目中一次性代码贡献者的印象、动机与障碍：一项调查”，载于 *ICSE*，2017 年，第 187–197 页。[18] J. Coelho、M. T. Valente、L. L. Silva 和 A. Hora，“我们为何参与自由/开源软件：来自核心开发者的回答”，载于*CHASE*，2018 年，第 114–121 页。[19] M. A. Gerosa、I. S. Wiese、B. Trinkenreich、G. J. P. Link、G. Robles、C. Treude、I. Steinmacher 和 A. Sarma，“动机的流沙：重新审视驱动开源贡献者的因素”，*ICSE*，第 1046–1058 页，2021 年。[20] S. L. Vadlamani 和 O. Baysal，“研究 Stack Overflow 和 GitHub 上的软件开发者专业能力与贡献”，载于 *ICSME*，2020 年，第 312–323 页。[21] G. Avelino、E. Constantinou、M. T. Valente 和 A. Sere-brenik，“开源项目的弃用与存续：一项实证调查”，*ESEM*，第 1–12 页，2019 年。[22]V. N. Subramanian、I. Rehman、M. Nagappan 和 R. Kula，“分析 GitHub 上的首次贡献：新手都做些什么？”*IEEE Software*，第 39 卷，第 01 期，第 93–101 页，2022 年。[23] D. Spencer，卡片分类：设计易用的类别。Rosenfeld Media，2009 年。[24] T. Zimmermann，“卡片分类：从文本到主题”，载于软件工程数据科学视角。爱思唯尔，2016 年，第 137–141 页。[25] A. Begel 和 T. Zimmermann，“分析一下！面向软件工程领域数据科学家的 145 个问题”，载于 *ICSE*，2014 年，第 12–23 页。[26]A. Guzzi、A. Bacchelli、M. Lanza、M. Pinzger 和 A. Van Deursen，“开源软件开发邮件列表中的沟通”，载于 *MSR*。IEEE，2013 年，第 277–286 页。[27] J. L. Campbell、C. Quincy、J. Osserman 和 O. K. Pedersen，“半结构化访谈的深入编码：单元划分、编码员间信度与一致性问题”，《社会学方法与研究》，第 42 卷，第 3 期，第 294–320 页，2013 年。