借贷类型？开发者人格与技术债务研究

Lorenz Graf-Vlachy*† 和 Stefan Wagner*
*斯图加特大学软件工程研究所德国斯图加特

†多特蒙德工业大学德国多特蒙德{
lorenz.graf-vlachy|stefan.wagner}
@iste.uni-stuttgart.de

arXiv:2303.02244v1 [cs.SE] 3 Mar 2023

摘要—背景：**技术债务（TD）在软件工程研究中得到了广泛讨论，相关研究也逐渐开始探讨其与开发者特征之间的联系。然而，在这一背景下，开发者人格尚未得到研究。**目标与方法：**我们探讨开发者的多种人格特质（五因素模型、调节焦点和自恋）与技术债务引入和消除之间的关系。为此，我们在现有技术债务数据集中补充了通过调查开发者收集的新型自我报告人格数据，并分析了来自 19 名开发者的 2,145 次提交。**结果：**我们发现，尽责性、情绪稳定性、经验开放性和预防焦点与技术债务呈负相关。外向性、宜人性、促进焦点或自恋方面的结果均不显著。**结论：**我们将研究结果视为初步证据，表明开发者人格会系统性地影响技术债务的引入和消除。这不仅对未来研究具有启示，例如可以研究人格对技术债务下游后果（如缺陷）的影响，也对软件工程从业者具有启示；例如，他们可以在人员配置决策中考虑开发者人格。

索引词—**技术债务、人格、五因素模型、大五人格、调节焦点、自恋**

一、引言

A. 技术债务及其后果

自 Cunningham 那句名言“首次交付代码就像负债” [1, p. 30], 以来，“技术债务”（TD）这一概念在从业者群体和学术界都日益受到关注 [2]。尽管对 TD 的定义略有不同，研究者已逐渐形成共识：TD 源于技术妥协，这些妥协会产生“一系列设计或实现构件，它们在短期内便于使用，却会形成一种技术背景，使未来的变更成本更高，甚至无法进行”。因此，TD“构成了实际或或有负债” [3, p. 112]。

尽管技术债务（TD）可能带来益处，尤其是在开发者工作效率方面，但相关文献经常讨论其潜在的不利影响 [4]。技术债务最常被提及的不良后果之一，是对开发者士气的负面影响 [5]。开发者通常不喜欢累积技术债务 [6],，而且研究发现这会带来心理负担 [7]。因此，技术债务会产生打击积极性的影响 [2]。其他研究则扩大了考察范围，考虑了诸如由于可维护性降低而导致开发者生产力下降等后果代码、代码质量下降（导致更多缺陷及后续成本），以及不确定性和风险增加 [8]。

B. 将技术债务视为需要管理的冒险行为

事实上，在很大程度上，TD 问题可以被视为冒险行为。先前的学者已明确指出，根据上述定义也可得出，TD 关乎权衡决策 [3], [9]。一方面，关于 TD 的这类决策是在当下节省精力与未来必须偿还本金并支付利息之间进行权衡。因此，这些决策构成冒险行为，因为隐含利率只有到了未来才能完全明了，而未来充满不确定性，因而存在风险 [10], [11]。另一方面，TD 的风险性还因以下事实而加剧：不仅利息金额不明确，而且还存在“债务可能永远不必偿还的可能性” [12,p.52] 。关于这一问题，最透彻的讨论之一或许见于 Schmid 对 TD 隐喻局限性的研究。他区分了“潜在技术债务”与“有效技术债务” [10, p. 64],: 前者被他明确描述为“类似于一种风险”，后者则是与软件系统未来演进实际相关的 TD 数量（即未来确实需要修复的那部分，但当然无法对其进行完美识别和预测）。

相应地，技术债务通常被视为一种或许需要在某个时候偿还的债务，但至少也应当得到审慎管理。在这一背景下，至少存在两条重要的研究脉络。首先，研究人员投入了大量精力来衡量，也就是评估系统中技术债务的规模。研究人员开发了不同的指标，例如代码异味 [13]。还有一些研究人员开发了各种工具，利用代码度量指标等方式自动衡量技术债务 [14]。SonarQube 等工具提供可维护性的综合质量指标，尤其受到初级开发人员的青睐 [15]。最终，研究人员不出所料地投入了大量精力来估算技术债务成本 [4], [16]。

其次，研究人员一直致力于识别、排序并偿还技术债务。例如，Digkas 等人研究了 Apache生态系统中如何处理 [17] 并偿还 [18] 技术债务。Nayebi 等人提供了一项

关于医疗通信产品背景下如何识别并偿还架构技术债务的详尽纵向案例研究。Maipradit 等人最近开发了一种自动分类器，用于识别代码 [19]中的自承认技术债务。除这些单项研究外，Alves等人还概述了用于识别和管理技术债务的工具，例如成本效益分析和组合方法 [13]。近期还有一些与技术债务优先级排序相关的综述（既包括技术债务之间的排序，也包括技术债务与新功能之间的排序），从中归纳出一系列用于确定技术债务优先级并进行管理的策略。例如，Alfayez 等人归纳了24种不同的策略 [20], , 而 Lenarduzzi 等人则将他们归纳出的策略整合为四个更宽泛的类别，以及一个额外的组合方法类别 [21]。

C. 技术债务的前因

为了更好地管理技术债务，研究人员也开始明确探讨其原因和前因。例如，一些研究人员指出，企业的商业模式创新等因素是推动技术债务产生的原因 [22]。另一些研究发现，项目中的提交数量和代码行数（LOC）与技术债务呈正相关 [23]。还有一类研究着重描述企业采用哪些激励与惩罚措施，促使开发者避免产生或偿还技术债务 [24]。总体而言，众多因素似乎都会影响技术债务。例如，Rios 等人开展了一项定性研究，确定了57项推动技术债务产生的因素，其中因截止期限迫近而产生的时间压力排名最高 [25]。随后，Rios 等人 在一项大规模调查中将这一数量增加到 78 [26]。之后，这项调查的跨国复制研究归纳出技术债务前因的八大类别，并再次指出截止期限是技术债务最重要的成因 [2]。有趣的是，从业者对技术债务主要成因的看法会随着经验变化。经验更丰富的软件开发人员较少将技术问题视为技术债务的成因，而更多关注人为因素 [27]。

因此，或许正如预期的那样，一支具体的研究分支已开始关注个体开发者，以了解技术债务的起源。例如，Amanatidis 等人研究了多个 PHP 项目，发现不同开发者所造成的技术债务数量存在显著差异 [28]。Alfayez 等人以及 Codabux 和 Dutchyn 在其他代码库中再次证实了这些研究发现，并进一步明确刻画了个体开发者的特征，识别出多种能够预测其引入和消除技术债务行为的特征 [29], [30]。

然而，迄今为止，我们尚未发现有研究将开发者的人格与技术债务联系起来。这令人惊讶，因为心理学研究的核心原则之一是，人格能够有力地预测个体的各种行为 [31] ，因此，人们可能会认为，人格也会影响某位开发者是否更愿意承担或偿还技术债务，而这又可能产生重要影响，例如影响项目团队人员配置。因此，在本研究中，我们提出以下研究问题：开发者人格与引入和消除技术债务有何关联？

II. 背景与假设

本文研究开发者人格的若干不同方面及其与技术债务之间的关系。具体而言，我们考察五因素模型涵盖的人格特质、调节焦点和自恋。下文将阐述各方面的理论背景，概述其在软件工程研究中的既有应用，并提出假设，探讨每种特质可能如何与技术债务的变化相关。

A. 五因素模型人格特质

五因素模型（FFM；有时也称为“大五人格”）可以说是心理学中最成熟的人格特质模型之一 [32]–[34]。该模型采用心理词汇学方法构建，包含五种宽泛的人格特质——外向性、宜人性、尽责性、情绪稳定性（或其反向特质神经质）以及经验开放性——这些特质在不同情境下都相当稳定。研究证据表明，这些特质对于不同性别、种族、文化和年龄群体同样有效，因此具有普遍性 [35]。

五因素模型已被广泛用于软件工程研究。因此，以下内容仅对近期研究进行举例说明。[1] 例如，Rastogi 和 Nagappan [38],研究了 GitHub 贡献者的五因素模型人格特质。他们发现，贡献更多代码的开发者在经验开放性、尽责性和外向性方面得分更高，但在情绪稳定性和宜人性方面得分更低。然而，Calefato 等 [37] 后来使用改进的人格测量工具重复了这项研究，却未发现这些影响。

Karimi 等人 [39]研究了学生程序员的人格与编程风格及表现（包括代码质量）之间的关系。他们发现，经验开放性与广度优先编程风格呈正相关，尽责性则与深度优先风格呈正相关。他们没有明确将五因素模型特质与编程表现联系起来，但发现广度优先编程风格与更出色的表现相关。

Paruma-Pab´on 等人 [40] 通过软件项目邮件列表邮件，提取开发者的五因素模型人格，以及开发者的需求和价值观，并将人格类型相似的开发者归为一类。他们证明，拥有提交权限的开发者的人格与其在项目中的行为相关。

与此相关，Calefato等人 [37]使用五因素模型研究了 Apache 开发者的人格，发现存在三种常见的人格档案。神经质和宜人性是最能区分这些档案的两种特质。他们还发现，五因素模型特质不会因开发者角色、成员身份或对其项目的贡献程度而有所不同，并证明

[1]近来有几篇期刊论文较为全面地综述了五因素模型在软件工程研究中的应用 [36], [37]。

开发者人格不会随时间变化。此外，他们发现，开发者的经验开放性与其做出项目贡献的可能性呈正相关。

最后，Iyer等人 [36] 研究了开源项目中拉取请求的接受情况。他们发现，开放性和尽责性得分较高的作者提交的拉取请求更有可能获批。他们还发现，外向性与获批概率呈负相关。此外，他们发现，由尽责性和外向性得分较高的程序员关闭的拉取请求，获批的可能性更大。相比之下，关闭者的情绪稳定性与获批可能性降低有关。

下文将把软件开发人员的每种五因素模型人格特质与诱发性技术债务联系起来。首先，外向性表示个体与外部世界互动的程度。外向性的子维度包括友善、合群性、果断性、活跃性和寻求刺激。因此，外向者可以被描述为性格外向、常常感受到积极情绪，并寻求刺激性活动。相比之下，内向者可以被描述为不那么外向、害羞且更喜欢独处 [32], [33], [41]。由于外向性更强的开发者更倾向于活跃和寻求刺激，我们提出，他们会“快速行动，打破常规”，从而遗留下比内向开发者更多的技术债务。现有文献还表明，外向性与冒险行为呈正相关 [42]。鉴于先前的软件工程定性研究指出，“风险偏好较高可能影响人们做出产生技术债务的决策” [8, 第 1504],页，这进一步表明外向性与技术债务可能呈正相关。正式表述如下：

假设*1*（*H1*）：外向性与诱发性技术债务呈正相关。

宜人性指个体对合作与社会和谐的关注程度。宜人性的子维度包括信任、道德、利他主义、合作、谦逊和同情心。宜人性较高的人可以被描述为友善、乐于助人且善解人意。相比之下，宜人性较低的人则不那么友善、不太合作，并且容易引发分歧 [32], [33], [41]。因此，我们预计，宜人性较高的开发者更愿意支持他人改进代码，从而偿还技术债务，而不是增加技术债务。关于冒险行为的文献也表明，宜人性与较少的冒险行为相关 [42]。因此，我们提出以下推测：

假设*2*（*H2*）：宜人性与诱发性技术债务呈负相关。

尽责性表示对人类冲动的控制程度。尽责性的子维度包括条理性、可靠性、成就追求和自律。因此，尽责的人可以被描述为谨慎、有组织且可靠。相比之下，不尽责的人可以被描述为冲动、缺乏系统性且不太可靠 [32], [33], [41]。因此，我们预计，尽责性更高的开发者会不喜欢技术债务，并且很少引入技术债务，甚至会将其消除。关于冒险行为的心理学文献也持有类似观点，发现尽责性与冒险行为之间总体上呈负相关关系 [42]。我们提出：

假设*3*（*H3*）：尽责性与诱发性技术债务呈负相关。

情绪稳定性通常通过其反面特质——神经质来描述。神经质体现了一个人产生负面情绪的程度。神经质包含的子维度包括焦虑、愤怒、抑郁、自我意识和脆弱性。神经质较高的人往往紧张、经常心情不好且情绪化。相比之下，情绪稳定的人通常比较平静，不会持续处于心情不佳的状态 [32], [33], [41]。神经质较高的人在不确定情境中容易感到焦虑，因此往往会果断采取行动来消除这种焦虑 [43] ，并尽量减少可能出现的潜在威胁，即使这意味着接受一些明确的负面结果 [44]。换言之，他们在不确定情境中经常采用“宁可谨慎也不冒险策略” [45, p. 1005]。如上所述，关于技术债务的选择总是在不确定性下做出，并涉及权衡当前收益与潜在的未来负担 [11], ，这表明神经质可能会促使人们通过避免技术债务来规避不确定性。心理学文献还探讨了神经质与自我导向完美主义之间可能存在的联系 [46], ，后者可能促使程序员编写他们认为“更好”的代码，而这类代码很可能技术债务更少。关于冒险行为的文献也支持这一观点，并显示情绪稳定性与冒险行为之间存在正相关关系 [42]。因此，我们提出以下假设：

假设*4*（*H4*）：情绪稳定性与诱发性*TD*呈正相关。

经验开放性描述了个体富有想象力和创造力的程度。开放性的子维度包括想象力、艺术兴趣、冒险性、自由主义和智识。开放性较高的人可以被描述为具有个人主义倾向、不墨守成规，并且能够意识到自己的感受。相比之下，开放性较低的人可以被描述为脚踏实地、传统，并且不太能意识到自己的感受 [32],[33], [41]。因此，我们提出，经验开放性较高的开发者会较少受到编写所谓“整洁”代码这一理念的束缚，因而更有可能诱发TD。此外，既有文献还将经验开放性与更高程度的冒险行为联系起来 [42]。因此，我们提出：

假设*5*（*H5*）：开放性与诱发性*TD*呈正相关。

B. 调节焦点

尽管五因素模型（FFM）涵盖面广，但心理学文献中常会将其与其他特质结合起来研究 [47]。因此，我们也研究调节焦点这一人格心理学中已有确立、但尚未受到软件工程研究关注的构念。一个人的调节焦点由两种相互独立的自我调节取向（或焦点）构成，它们会影响其目标追求行为：预防焦点和促进焦点 [48], [49]。[2] 促进

[2]请注意，调节焦点可以被概念化为一种状态或一种特质。正如心理学文献中通常采用的做法一样，我们只关注后者，这一成分有时也被称为“慢性”调节焦点 [48]。

焦点与积极进取、勇于冒险以及以获取收益等积极结果为导向有关。相比之下，预防焦点与谨慎小心以及以避免损失等负面结果为导向有关。鉴于产生技术债务（TD）属于冒险行为，且相关文献指出调节焦点与冒险行为有关 [50], ，我们因此提出，促进焦点越强的开发者越可能引入技术债务，而预防焦点越强的开发者则不那么可能。

假设*6*（*H6*）：促进焦点与诱发性*TD*呈正相关。

假设*7*（*H7*）：预防焦点与诱发性*TD*呈负相关。

C. 自恋

尽管学界仍在讨论该构念的确切定义 [51], ，但学者们普遍认为，自恋是一种人格特质，它结合了“宏大却脆弱的自我感和权利感，以及对成功的专注和对赞赏的需求” [52,第440–441]页。虽然有关自恋型软件开发人员的各种博客文章表明，自恋可能是软件工程实践中一个值得关注的话题，但据我们所知，目前尚无科学研究明确探讨软件工程师中的这一人格特质。相比之下，心理学领域已有大量相关研究，并发现自恋与一系列相关因素有关，例如追求地位 [53], 、很少顾及他人的关切，以及将他人视为低人一等 [54]。

自恋的一个特别突出的相关因素是冒险行为。具体而言，研究人员发现，自恋与一系列高风险活动存在关联。这些活动包括高风险性行为和性侵犯、攻击性驾驶、毒品和酒精使用、强迫性锻炼以及赌博 [55]。鉴于承担技术债务本质上属于冒险行为，我们提出，更具自恋倾向的开发者可能会在其提交中诱发更多技术债务。

假设*8*（*H8*）：自恋与诱发性技术债务呈正相关。

III. 方法

A. 技术债务度量

为衡量技术债务，我们采用了版本为 2 [56] 的“技术债务数据集”。该数据集使用广受欢迎的 SonarQube 工具，对 29 个 Apache 开源项目的主分支进行了全面分析，并已用于此前的研究 [30]。

我们衡量焦点提交中诱发性TD的主要指标沿用以往研究的做法，依据 SonarQube [29]识别出的所有可维护性问题的估算修复工作量。具体而言，焦点提交的数值是焦点提交与其父提交之间估算修复工作量的差值。因此，正值表示技术债务增加，负值则表示技术债务得到偿还。在另一项范围更广的诱发性TD次要指标中，我们还纳入了 SonarQube 识别出的所有可靠性和安全问题的估算修复工作量。

由于比率作为因变量可能存在问题 [57], ，我们偏离了以往研究的做法 [29] ，不将TD指标除以焦点提交与父提交之间的代码行数（LOC）差值。相反，而且可以说这种方法更为精确，我们分别控制焦点提交中新增和删除的代码行数。这考虑到了代码行数增加可能使技术债务更容易产生这一事实 [29]。

B. 调查测量

为获取人格数据，我们对在技术债务数据集中至少提交过一次的开发者进行了调查。由于数据集本身不包含开发者的联系信息，第一步是使用 PyDriller 从 GitHub 获取这些项目的所有提交信息 [58]。由此共获得 99,972 条提交记录，其中部分记录的时间范围超出了该数据集的时间范围。我们找出数据中所有被列为“作者”的个人，并手动清理名单以删除重复项。对于不同提交中使用不同姓名但电子邮件地址相同，或使用不同电子邮件地址但姓名相同或极为相似的个人，我们也合并了其记录。需要作出判断时，我们尽可能采取保守的处理方式。最终，我们获得了一份包含 1,555 名不同个人的名单。对于拥有多个电子邮件地址的人，我们只选择其中一个，并优先选择个人电子邮件地址而非工作邮箱，因为这些人对项目的提交有些已经相当久远，他们可能在此后更换了所属组织。

我们向全部 1,555 名开发者发送了电子邮件，邀请他们参加我们的调查 [59] （该调查是多项研究所开展的一项更大规模数据收集工作的一部分）。我们在邀请邮件中附上了调查的个性化链接。邮件中承诺，每收到一份完整答卷，我们就向联合国世界粮食计划署捐赠 2 美元 [60]。我们还发送了两封提醒邮件 [59],，其中一封附有大学官方网站的链接，以确认调查的真实性，因为一些开发者担心邀请邮件可能是骗局。

由于有 165 封电子邮件被退回，我们实际联系到了 1,390 名开发者（电子邮件送达率为 89.4%）；其中 194 名开发者开始填写调查，124 人完成了调查。我们剔除了所有年龄值不合理的受访者（$\leq$10 或 $\geq$100），最终得到 121 名开发者的初步样本。尽管这个数字看起来可能偏低，但与其他研究中获得的总人数规模相近；那些研究评估人格时，并非依靠自我报告，而是通过挖掘海量电子邮件语料进行分析。例如，Calefato 等人使用了来自 46,304 名开发者的约 135 万封电子邮件，但仅为其中 211 人（<0.5%）获得了完整的人格档案，而其中只有 118 人提交过源代码 [37]。我们的回复率为 8.9%，与其他针对 GitHub 开发者开展的调查相近。例如，Graziotin 等人报告的回复率为 7% [61]。他们报告的电子邮件送达率为 96.6%，也与我们的情况相近。

调查的第一页向参与者介绍了将收集哪些类型的数据。我们向开发者保证，他们的数据将得到严格保密，不会与其他方共享。

我们强调，参与调查完全出于自愿，开发者可以随时退出。我们还明确说明，开发者继续浏览至下一页即表示同意参与我们的研究。我们邀请了三名软件工程专业的博士生对调查进行预测试，并根据他们的反馈做了少量修改。

为测量受访者的五因素模型人格特质，我们采用了广泛使用的十项人格测量表（TIPI） [41]。由于 TIPI 旨在通过每个维度各一个题项，测量大五人格的所有方面，并具有内容效度和效标效度，因此 Cronbach's α 等可靠性指标并无参考意义 [62]。因此，我们不报告这些指标。

为测量调节焦点，我们从经过广泛验证的调节焦点综合量表（RF-COMP） [63]中选取了六个题项（促进焦点和预防焦点各三个）。为检验内部一致性，我们对样本中 RF-COMP 的全部十个题项进行了因子分析。结果显示，出现了多于预期的两个不同因素。遇到此类情况时，通常的做法是减少题项数量，直到出现两个清晰的因素——一个反映促进焦点，另一个反映预防焦点。随后，我们得到促进焦点的 Cronbach's α 为 .66，预防焦点为 .60。鉴于每个构念的题项数量较少，这些数值尚属可接受 [64]。

我们使用自恋人格量表的简版（NPI-16量表） [52]测量自恋。Cronbach's α为.70，属于可接受的数值 [65] ，甚至略高于最初开发该量表的研究中所得到的数值 [52]。

由于我们是在分析提交完成后才测量人格的，这可能会引发对分析中因果方向不明确的担忧。不过，这不太可能构成问题，因为人们普遍认为人格在成年期保持稳定 [66], [67]，而纳入样本的开发者年龄均在24至65岁之间。与此一致，近期一项专门研究开发者人格的研究未发现人格随时间变化的任何证据 [37]。

为了测量开发者在每次提交时的年龄，我们请开发者直接提供其年龄（以年为单位）。随后，我们从其提供的年龄中减去2022年与焦点提交年份之间的差值。

C. 最终样本

我们仅保留普通提交，剔除了合并提交和孤立提交，因为前者无法计算诱发性TD（由于存在多个父提交），而后者可能具有某些特殊特征。请注意，纳入孤立提交不会改变我们的结果。

由于我们的 TD 测量要求对目标提交和父提交完成 SonarQube 分析，而技术债务数据集中存在大量缺失数据，因此最终我们的样本缩减为来自 19 名开发者的 $N = 2145$ 个提交；对于这些提交，我们同时拥有完整的 TD 数据和人格数据。虽然我们的样本因此无论如何都算不上大，但与例如 Calefato 和 Lanubile 关于人格与信任的研究中包含 28 名个体的样本相比，也并没有小太多（该研究考察了 Apache 开发者之间的信任），并且远大于 Rigby 和 Hassan 在其关于高生产力 Apache 开发者五因素模型特质的研究中所采用的四名程序员样本 [69]。

D. 分析策略

我们采用面板回归，研究人格与技术债务测量指标之间的关联。这种方法适用于反复观测面板单位（在我们的研究中为开发者）的情形。我们在面板单位层面计算聚类稳健标准误，以考虑同一开发者的多次提交在统计上并非相互独立这一事实。模型中纳入了提交时开发者的年龄、新增代码行数和删除代码行数作为控制变量。此外，我们在分析中为每个项目纳入虚拟变量（固定效应）。这些变量捕捉了项目特有的所有时不变因素，例如某些编码规范。

E. 复现包

遗憾的是，我们无法向其他研究人员提供分析数据，因为我们明确向受访者承诺了充分的保密性。即便情况并非如此，我们也会对共享数据有所顾虑，因为我们无法对数据进行匿名化处理：本文的目的正是将人格与软件项目中可识别到个人的贡献联系起来。不过，技术债务数据集本身可公开获取 [56]。我们还提供了一个包含所有分析脚本的程序包。[3] 该程序包还包含一个虚拟数据文件，其中列出了所用调查测量的准确表述。

IV. 结果

A. 描述性统计

表 I 提供了最终数据集的描述性统计。图 1 展示了样本中每位开发者提交数量的直方图和核密度图。

表 I 描述性统计

	Mean	SD	Min.	Max.
诱发性TD	65.41	1,059.15	-14,160.00	32,880.00
诱发性TD（广义）	67.92	1,091.94	-14,320.00	33,829.00
外向性	3.57	1.36	1.00	6.00
宜人性	4.18	1.10	2.50	6.00
尽责性	5.82	1.11	3.50	7.00
情绪稳定性	4.83	1.06	1.50	6.50
经验开放性	4.82	1.25	3.00	7.00
经验				
促进焦点	15.30	2.39	10.00	21.00
预防焦点	13.13	3.52	9.00	18.00
自恋	3.66	2.14	0.00	9.00
提交时年龄	36.48	6.88	23.00	50.00
新增代码行数	217.63	1,565.86	0.00	40,180.00
删除代码行数	81.12	732.66	0.00	25,886.00

表 II 展示了除项目虚拟变量外所有变量之间的相关性。显然，这两种

[3]可访问地址：https://dx.doi.org/10.6084/m9.figshare.21802968

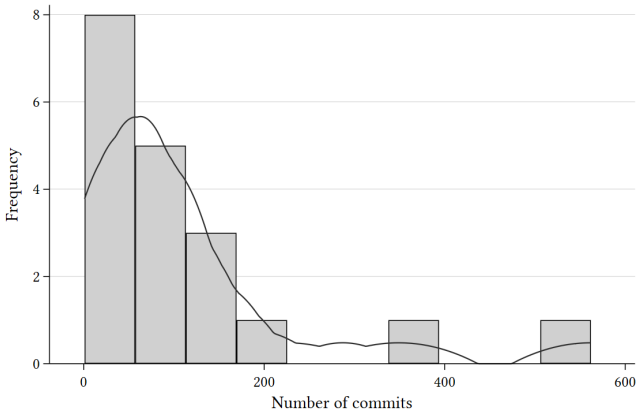


图 1. 每位开发者的提交次数直方图和核密度图

技术债务的衡量指标几乎完全相关，因此本质上衡量的是同一个构念。不出所料，诱发性TD与新增代码行数之间也存在很强的相关性。

已有充分文献记录，因此五因素模型特质与自恋之间存在相关性并不令人意外 [70]。例如，在我们的样本中，我们复现了心理学文献中广泛报告的结果：自恋与外向性呈正相关，与宜人性呈负相关 [70]。正如我们根据其他群体的调节焦点测量结果所预期的那样，促进焦点与预防焦点之间呈显著正相关 [49]。

B. 研究发现

表 III 报告了面板回归分析的回归系数和标准误，其中一次以诱发性TD的主要测量指标为因变量，另一次以诱发性TD的次要、较宽泛测量指标为因变量。从表中可以明显看出，两种因变量的结果相似。具体而言，外向性和宜人性的系数为负，且不显著。因此，这些结果并不支持 *H1* 和 *H2*。尽责性的系数为负且显著，支持*H3*。与我们的假设相反，情绪稳定性和经验开放性的系数均显著为负。因此，这些研究发现构成了反对 *H4* 和 *H5*的证据。调节焦点的结果则不尽相同。促进焦点的系数出乎意料地为负，但不显著。预防焦点的系数如预期为负，且显著。因此，这些结果不支持*H6* ，但佐证了*H7*。最后，自恋的系数为正，但不显著，因此不支持*H8*。

尽管文献中对于是否有必要进行多重检验校正，以及这种校正是否有助于解读结果、何时有助于解读结果，存在很大分歧 [71]–[73]（尤其是在多元回归的背景下），我们仍尝试控制假发现率方法是为所有八个感兴趣的自变量计算调整后的p值（经锐化处理的q值） [74]。研究结果再次证实了我们最初的发现。

我们未提出假设的控制变量的系数均显著，且方向并不出人意料。提交代码时的开发者年龄与诱发性TD呈负相关。新增代码行数和删除代码行数分别与诱发性TD呈正相关和负相关。

V. 讨论

A. 意外结果

虽然一些假设因回归分析中不显著的结果而未得到支持，但一些显著结果却直接与我们的假设相悖，因此有必要进一步讨论。首先，情绪稳定性与TD呈显著负相关。一种可能的解释是——这与心理学研究关于积极情感有助于增强问题解决和决策，以及促进灵活、创新、全面且高效的认知处理的结论一致 [75]——情绪更加稳定的开发者通常具有积极的情绪，这使他们能够找到更优的方法来生成高度可维护的代码，从而诱发更少的TD。

其次，经验开放性与TD呈负相关。一种合理的解释是，开放性较高的个体具有特定的审美敏感性 [34]，这种特质可能会体现在创作特别优美的代码上，而这类代码反过来可能只包含很少的TD。另一种解释是，这些个体具有很强的创造力 [34]，因此可能只是能够找到在代码中表达逻辑的良好方式，而一开始就不会产生TD。

B. 效度威胁

有几个潜在问题可能会威胁本研究的效度。我们将它们分为四类 [76]。

1) 构念效度： 有人可能会质疑我们的测量准确性。这一点对于自变量和因变量都成立。对于人格测量，我们采用了短量表。虽然这可能有助于提高调查的回复率，但与较长的量表相比，这些测量可能引入了更多的测量误差 [77]。此外，我们的人格测量基于自我报告。尽管此类测量是心理学中最受信任且使用最频繁的测量方式，但它们可能并非没有问题 [78]。例如，受访者可能会努力以尽可能有利的形象呈现自己，而不是完全准确地展示自己。软件工程师群体的回复率有限，再加上这一因素，使研究人员开始寻求其他数据来源。例如，近来从文本语料库推断人格变得十分流行 [36]–[38], [40]，但此类方法至少和自我报告一样存在问题。首先，近期研究表明，使用心理语言学方法从开发者交流（例如邮件列表）中捕捉人格，可能会导致严重的不准确 [79] ，而不同的语言学工具可能会使研究人员得出截然不同的结论 [80]。其次，此类研究大多只

表 II 相关性

	诱导TD	诱导TD（宽泛）	外向性	认同-性-ness	尽责-性-ness	Emo-ional 稳定-lity	开放性 体验 经历	Pro- 情绪 专注	Pre-vention 专注	自恋-sism	Age at 提交	LOC 添加
诱发性TD（宽泛）	1.00***											
外向性	0.02	0.02										
宜人性	0.02	0.02	-0.30***									
责任-责任心	-0.04*	-0.04*	-0.27***	-0.55***								
情绪稳定性	-0.02	-0.02	-0.31***	0.12***	0.26***							
经验开放性经验	0.04	0.04	0.42***	0.13***	-0.41***	-0.58***						
促进焦点	-0.04*	-0.04*	-0.20***	0.29***	-0.03	-0.13***	-0.15***					
预防焦点	-0.02	-0.02	0.15***	0.30***	-0.43***	-0.72***	0.46***	0.57***				
自恋	-0.02	-0.02	0.45***	-0.30***	0.00	-0.00	-0.06**	-0.46***	-0.18***			
提交时年龄	0.02	0.02	-0.46***	0.39***	-0.33***	0.20***	-0.41***	0.02	-0.11***	-0.19***		
新增代码行数	0.73***	0.73***	-0.01	0.04	-0.05*	-0.02	0.07***	-0.05*	-0.01	-0.04*	0.04	
删除代码行数	-0.13***	-0.13***	-0.01	0.03	-0.03	0.01	0.03	-0.01	-0.01	-0.01	-0.01	0.39***

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$。未显示项目的虚拟变量。 $N = 2{,}145$。

研究了五因素模型人格特质，而其他人格特质所对应的语言学方法并不容易获得。

尽管“心理学中很少有发现比人格的稳定性更稳健” [67, p. 175], 这一观点在一些软件工程研究中受到了质疑，有研究指出，随着参与时间的增长，贡献者会变得更尽责、更外向，也更不随和 [38]。虽然这些研究中的效应量很小 [37],，但这可能会影响我们人格测量的效度。

此外，SonarQube 给出的技术债务估计并不完美，而且很可能可以改进 [81]。事实上，自动化检测技术债务可能普遍存在局限。显然，源代码分析工具无法捕捉先前研究所识别出的所有类型的技术债务 [8], [13], [82]。同样，不同的技术债务检测工具往往会对技术债务得出不同的评估 [83]。因此，使用其他工具提供的不同技术债务测量方法重复我们的分析，会很有意思。

2）内部效度：我们的研究的内部效度可能受到多种因素的限制。首先，开发者人格可能并非影响技术债务的唯一因素，而且其他相关变量也未必都得到了控制。这可能导致遗漏变量问题，进而使回归分析得出虚假结论。

其次，我们的研究结果可能受到选择效应的影响。由于所研究的软件项目中并非所有作者都拥有提交权限，因此实际进行且我们能够观察到的提交，可能与所有提议的代码更改存在系统性差异。例如，正如其他研究者所证明的那样，提交者的性格也会影响他们的行为 [36],[40]。因此，我们的数据可能经过了特定方式的筛选，从而使我们的分析和研究发现产生偏差。

3）外部效度：自然，在基于软件仓库数据开展的软件工程研究中，一个关键问题是研究发现能否推广到其他情境。就此而言，至少存在两种对效度的潜在威胁。

首先，鉴于近期关于样本代表性的讨论， [59], [60],我们强调，我们的样本很可能无法代表所有软件开发人员。具体而言，我们仅从规模相对较大的 Apache 项目中抽取主要以 Java 编写的开源贡献者作为样本，而且调查回复率也较低。

其次，技术债务数据集存在大量缺失数据。具体而言，许多提交的 SonarQube 分析并不完整。由于技术债务度量不仅要求每个焦点提交的分析完整，也要求其父提交的分析完整，因此因变量中出现了大量缺失值。这些数据是否为随机缺失，仍不明确。

4）可靠性：我们的研究应具有较高的可靠性。我们用于评估人格的所有量表均已在心理学文献中得到确立 [59]。研究过程中几乎不需要人工判断。对于必须作出判断的情况，我们在本文中记录了所遵循的指导原则。我们还提供了用于数据分析的所有脚本。因此，原则上，我们的研究应当能够被完整复现。然而，由于我们向接受调查的开发者作出了保密保证，无法分享人格数据，因此其他研究者当然无法直接重复我们的分析。不过，我们认为，后一个问题并不构成对本研究效度的威胁。

C. 对研究的启示

我们的研究对未来研究有几方面启示。首先，也是最重要的一点，它表明开发者人格对

表 III 面板回归

	诱导 TD	诱导 TD（广义）
外向性	-34.80 (45.98)	-36.55 (48.32)
宜人性	-0.93 (25.17)	-1.91 (26.32)
尽责性	-185.20** (60.77)	-191.80** (63.85)
情绪稳定性	-178.47*** (21.78)	-183.47*** (22.74)
经验开放性	-189.82*** (43.50)	-192.61*** (45.14)
促进焦点	-31.66 (28.94)	-31.41 (30.36)
预防焦点	-44.31*** (7.09)	-45.70*** (7.44)
自恋	19.77 (20.67)	22.23 (21.52)
提交时年龄	-17.77* (7.51)	-17.94* (7.78)
新增代码行数	0.62*** (0.14)	0.64*** (0.14)
删除代码行数	-0.71*** (0.14)	-0.73*** (0.14)
常数项	4500.48*** (1152.82)	4596.41*** (1203.54)
项目固定效应	Yes	Yes
观测值	2,145	2,145
聚类	19	19

因变量见首行。表中报告系数，括号内为聚类稳健标准误。* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

技术债务有着显著影响。因此，我们建议沿着这一方向开展进一步研究。具体而言，我们鼓励在其他代码库中复现本研究，并考虑采用其他技术债务衡量指标，例如自承技术债务。此外，研究我们所用指标可能未能涵盖的其他类型的技术债务，或许也会有所启发 [8], [13], [82]。此外，技术债务可能带来的后果，如软件缺陷或项目预算超支，也可能与人格相关，值得进一步研究。其次，我们的研究表明，五因素模型之外的人格特质也可能有助于解释和预测软件开发人员的行为。尽管在我们的研究中，只有预防焦点被证实具有显著影响，但我们建议进一步研究调节焦点（总体而言），甚至可能包括自恋，考察它们与技术债务问题之外的软件工程决策之间的关系。最后，未来研究可以将增加技术债务和减少技术债务视为两种不同的活动。因此，我们建议分析技术债务净增的提交和技术债务净减的提交 [29]，以了解它们是否可能具有不同的人格前因。

D. 实践启示

我们的研究表明，具有某些人格档案的开发者可能更容易或更不容易产生 TD。这带来了重要的实践启示。首先，项目经理（他们通常非常了解自己的团队，也无需开展可能存在法律风险的人格调查问卷）可以据此为软件开发项目配备人员。例如，他们可以有意将已有过多 TD 的项目制品分配给人格档案可能有助于减少 TD 的开发者，例如预防焦点或尽责性较高的开发者。同样，他们也可以在不同项目阶段调整团队组成。按照 Beck 的 3X 模型，例如可以设想，“探索”阶段允许开发者产生更多 TD，而“扩展”阶段则需要更谨慎的开发者，“提取”阶段则最好由厌恶 TD 的开发者负责 [84]。其次，经理可以为密切跟踪人格档案容易导致 TD 的程序员所开发制品中的 TD，投入更多时间和资源。第三，在采用结对编程等技术时，项目经理可以考虑将人格作为组建搭档的标准。例如，可以让人格档案互补的程序员结对，以主动管理项目中的 TD。最后，我们的研究或许能帮助开发者自身更加了解与生俱来的倾向。本研究使用的所有人格测量工具通常都可以获取，也可用于培训项目，让开发者有机会评估自己的人格，并了解其行为对 TD 可能产生的影响。

VI. 结论

总之，开发者人格似乎会影响技术债务的产生与消除。我们希望本研究能引发对这一有趣问题的进一步探究。

致谢

我们谨此致谢 Davide Taibi 提供的有关技术债务数据集的有用信息。

参考文献

[1] W. Cunningham，《wycash投资组合管理系统》，载于*OOPSLA 1992*年会议论文集增补，1992年，第29–30页。[2] R. Ramaˇc、V. Mandi´c、N. Tauˇsan、N. Rios、S. Freire、B. P´erez、C. Castellanos、D. Correal、A. Pacheco、G. Lopez、C. Izurieta、C. Seaman和R. Spinola，《技术债务的普遍性、常见原因及影响：一组针对IT行业的调查结果》，《系统与软件杂志》，第184卷，第111114页，2022年。[3] P. Avgeriou、P. Kruchten、I. Ozkaya和C. Seaman，《软件工程中的技术债务管理（Dagstuhl研讨会16162）》，*Dagstuhl* 报告，第6卷，第4期，第110–138页，2016年。[4] T. Besker、A. Martini和J. Bosch，《技术债务的昂贵账单：何时以及由谁偿还？》，载于*ICSME 2017*，I. I. C. o. S. M. a. Evolution编，皮斯卡特维，新泽西州：IEEE，2017年，第13–23页。[5] H. Ghanbari、T. Besker、A. Martini和J. Bosch，《寻求内心平静？管理你的（技术）债务：一项探索性实地研究》，载于*2017*年 *ACM/IEEE* 实证软件工程与度量国际研讨会（*ESEM*）。IEEE，2017年，第384–393页。[6] T. Besker、H. Ghanbari、A. Martini和J. Bosch，《技术债务对软件开发者士气的影响》，《系统与软件杂志》，第167卷，第110586页，2020年。

[7] J. Olsson、E. Risfelt、T. Besker、A. Martini和R. Torkar，《从技术债务中度量情感状态》，《实证软件工程》，第26卷，第5期，2021年。[8] E. Tom、A. Aurum和R. Vidgen，《技术债务探索》，《系统与软件杂志》，第86卷，第6期，第1498–1516页，2013年。[9] F. Shull，《有限资源世界中的完美主义者》，《*IEEE*软件》，第28卷，第2期，第4–6页，2011年。[10] K. Schmid，《关于技术债务隐喻的局限：超越隐喻的一些指导》，载于*2013*年第*4*届技术债务管理国际研讨会（*MTD 2013*），P. Kruchten编。皮斯卡特维，新泽西州：IEEE，2013年，第63–66页。[11] P. Kruchten、R. L. Nord、I. Ozkaya和D. Falessi，《技术债务：迈向更明确的定义》，*ACM SIGSOFT*《软件工程札记》，第38卷，第5期，第51–54页，2013年。[12] E. Allman，《技术债务管理》，《*ACM*通讯》，第55卷，第5期，第50–55页，2012年。[13] N. S. Alves、T. S. Mendes、M. G. de Mendonc¸a、R. O. Sp´ınola、F. Shull和C. Seaman，《技术债务的识别与管理：一项系统性映射研究》，《信息与软件技术》，第70卷，第100–121页，2016年。[14] P. C. Avgeriou、D. Taibi、A. Ampatzoglou、F. Arcelli Fontana、T. Besker、A. Chatzigeorgiou、V. Lenarduzzi、A. Martini、A. Moschou、I. Pigazzini、N. Saarimaki、D. D. Sas、S. S. de Toledo和A. A. Tsintzira，《技术债务度量工具概览与比较》，《*IEEE*软件》，第38卷，第3期，第61–71页，2021年。[15] F. Gilson、M. Morales-Trujillo和M. Mathews，《初级开发者如何应对其技术债务？》，载于第*3*届技术债务国际会议论文集，ACM数字图书馆丛书，C. Izurieta编。纽约州纽约：计算机协会，2020年，第51–61页。[16] B. Curtis、J. Sappidi和A. Szynkarski，《估算应用程序技术债务的本金》，《*IEEE*软件》，第29卷，42，2012年6，34第，期。页– 。[17] G. Digkas、M. Lungu、A. Chatzigeorgiou和P. Avgeriou，《Apache生态系统中技术债务的演化》，载于软件架构，LNCS子库：SL2——编程与软件工程，A. Lopes和R. de Lemos编。瑞士Cham：Springer，2017年，第10475卷，第51–66页。[18] G. Digkas、M. Lungu、P. Avgeriou、A. Chatzigeorgiou和A. Am-patzoglou，《开发者如何修复问题并偿还Apache生态系统中的技术债务？》，载于第*25*届*IEEE*软件分析、演化与再工程国际会议，E. IEEE International Conference on Software Analysis and Reengineering编。皮斯卡特维，新泽西州：IEEE，2018年，第153–163页。[19] R. Maipradit、C. Treude、H. Hata和K. Matsumoto，《敬请期待：识别“搁置”的自承技术债务》，《实证软件工程》，第25卷，第5期，第3770–3798页，2020年。[20] R. Alfayez、W. Alwehaibi、R. Winn、E. Venson和B. Boehm，《技术债务优先级排序的系统性文献综述》，载于第*3*届技术债务国际会议论文集，ACM数字图书馆丛书，C. Izurieta编。纽约州纽约：计算机协会，2020年，第1–10页。[21] V. Lenarduzzi、T. Besker、D. Taibi、A. Martini和F. Arcelli Fontana，《关于技术债务优先级排序的系统性文献综述：策略、流程、因素和工具》，《系统与软件杂志》，第171卷，第110827页，2021年。[22] J. Yli-Huumo、T. Rissanen、A. Maglyas、K. Smolander和L.-M. Sainio，《商业模式实验与技术债务之间的关系》，载于软件业务，《商业信息处理讲义》丛书，J. M. Fernandes、R. J. Machado和K. Wnuk编。纽约，纽约州：Springer Berlin Heidelberg，2015年，第210卷，第17–29页。[23] J. Bedi和K. Kaur，《理解影响技术债务的因素》，《国际信息技术杂志》，第14卷，第2期，第1051–1060页，2022年。[24] T. Besker、A. Martini和J. Bosch，《利用激励措施促进技术债务管理》，《信息与软件技术》，第142卷，第106740页，2022年。[25] N. Rios、R. Oliveira Spinola、M. G. de Mendonca Neto和C. Seaman，《导致开发团队在软件项目中承担技术债务的因素研究》，载于*2018*年第*44*届*Euromicro*软件工程与高级应用会议（*SEAA 2018*）。皮斯卡特维，新泽西州：IEEE，2018年，第429–436页。[26] N. Rios, R. O. Sp´ınola, M. Mendonc¸a, 和 C. Seaman，《从业者对技术债务概念及其成因与后果的看法：面向全球工业调查系列的设计及其首批巴西调查结果》，《实证软件工程》，第25卷，第5期，第3216–3287页，2020年。[27] S. Freire, N. Rios, B. Perez, C. Castellanos, D. Correal, R. Ramac, V. Mandic, N. Tausan, G. Lopez, A. Pacheco, D. Falessi, M. Mendonca, C. Izurieta, C. Seaman, 和 R. Spinola，《经验如何影响从业者对技术债务成因与影响的认知》，载于2021年IEEE/ACM第13届软件工程协作与人文方面国际研讨会。新泽西州皮斯卡特维：IEEE，2021年，第21–30页。[28] T. Amanatidis, A. Chatzigeorgiou, A. Ampatzoglou, 和 I. Stamelos，《谁产生了更多技术债务？》，载于《XP2017科学研讨会论文集》，R. Tonelli 编。美国纽约州纽约：ACM，2017年，第1–8页。[29] R. Alfayez, P. Behnamghader, K. Srisopha, 和 B. Boehm，《开发者对技术债务影响的探索性研究》，载于《2018年技术债务国际会议论文集》，ACM会议系列，R. L. Nord 编。纽约：ACM，2018年，第1–10页。[30] Z. Codabux 和 C. Dutchyn，《透过技术债务的视角为开发者画像》，载于《第14届ACM/IEEE实证软件工程与测量国际研讨会论文集》（ESEM）。美国纽约州纽约：ACM，2020年，第1–6页。[31] O. P. John 和 R. W. Robins 编，《人格手册：理论与研究》，第4版。纽约：吉尔福德出版社，2021年。[32] G. Matthews, I. J. Deary, 和 M. C. Whiteman，《人格特质》，第2版。剑桥：剑桥大学出版社，2003年。[33] R. R. McCrae 和 O. P. John，《五因素模型及其应用简介》，《人格杂志》，第60卷，215，1992 2，第175页。[34] O. P. John，《大五人格特质分类体系的历史、测量与概念拓展：范式走向成熟》，载于《人格手册》，O. P. John 和 R. W. Robins 编。纽约：吉尔福德出版社，2021年，第35–82页。[35] C. M. Ching, A. T. Church, M. S. Katigbak, J. A. S. Reyes, J. Tanaka-Matsumi, S. Takaoka, H. Zhang, J. Shen, R. M. Arias, B. C. Rincon, 和 F. A. Ortiz，《日常行为与情感中的特质表现：五种文化研究》，《人格研究杂志》，第48卷，第1–16页，2014年。[36] R. N. Iyer, S. A. Yun, M. Nagappan, 和 J. Hoey，《人格特质对拉取请求接受的影响》，《IEEE软件工程汇刊》，第47卷，第11期，第2632–2643页，2021年。[37] F. Calefato, F. Lanubile, 和 B. Vasilescu，《对Apache生态系统中开发者人格的大规模深入分析》，《信息与软件技术》，第114卷，第1–20页，2019年。[38] A. Rastogi 和 N. Nagappan，《关于GitHub贡献者的人格特质》，载于《2016年IEEE第27届软件可靠性工程国际研讨会》（ISSRE）。IEEE，2016年，第77–86页。[39] Z. Karimi, A. Baraani-Dastjerdi, N. Ghasem-Aghaee, 和 S. Wagner，《计算机编程中的人格、风格与绩效之间的关联》，《系统与软件杂志》，第111卷，241，2016 228，第。[40] O. H. Paruma-Pab´on, F. A. Gonz´alez, J. Aponte, J. E. Camargo, 和 F. Restrepo-Calle，《通过挖掘开发者电子邮件发现社会技术因素与人格特质之间的关系》，载于《第9届软件工程协作与人文方面国际研讨会论文集》。美国纽约州纽约：ACM，2016年，第8–14页。[41] S. D. Gosling, P. J. Rentfrow, 和 W. B. Swann，《大五人格领域的一个极简测量工具》，《人格研究杂志》，第37卷，第6期，第504–528页，2003年。[42] N. Nicholson, E. Soane, M. Fenton-O’Creevy, 和 P. Willman，《人格与领域特定风险承担》，《风险研究杂志》，第 8 卷，第2期，第157–176页，2005年。[43] T. A. Judge, J. B. Rodell, R. L. Klinger, L. S. Simon, 和 E. R. Crawford，《用于预测工作绩效的人格五因素模型层级表征：整合三种组织框架与两种理论视角》，《应用心理学杂志》，第98卷，第6期，第875–925页，2013年。[44] J. B. Hirsh 和 M. Inzlicht，《熟悉的魔鬼：神经质可预测对不确定性的神经反应》，《心理科学》，第19卷，第10期，第962–967页，2008年。

[45] M. J. Lommen, I. M. Engelhard, 和 M. A. van den Hout, 《神经质与回避模糊刺激：宁可谨慎，也不愿后悔？》，《人格与个体差异》，第49卷，第8期，第1001–1006页，2010年。[46] G. L. Flett, P. L. Hewitt, 和 D. G. Dyck, 《自我导向完美主义、神经质与焦虑》，《人格与个体差异》，第10卷，第7期，第731–735页，1989年。[47] T. C. Marshall, K. Lefringhausen, 和 N. Ferenczi, 《大五人格、自尊与自恋能否预测人们在Facebook状态更新中撰写的主题》，《人格与个体差异》，第85卷，第35–40页，2015年。[48] E. T. Higgins, 《超越快乐与痛苦》，《美国心理学家》，第52卷，第12期，第1280–1300页，1997年。[49] K. Lanaj, C.-H. D. Chang, 和 R. E. Johnson, 《调节焦点与工作相关结果：综述与荟萃分析》，《心理学通报》，第138卷，第5期，第998–1034页，2012年。[50] M. R. Hamstra, J. W. Bolderdijk, 和 J. L. Veldstra, 《日常风险承担与调节焦点的关系》，《人格研究杂志》，第45卷，第1期，第134–137页，2011年。[51] M. B. Donnellan, R. A. Ackerman, 和 A. G. Wright, 《当代人格心理学中的自恋》，载于《人格手册》，O. P. John 和 R. W. Robins 编。纽约：吉尔福德出版社，2021年，第625–641页。[52] D. R. Ames, P. Rose, 和 C. P. Anderson, 《作为自恋简短测量工具的NPI-16量表》，《人格研究杂志》，第40卷，第4期，第440–450页，2006年。[53] V. Zeigler-Hill, G. A. McCabe, J. K. Vrabel, C. M. Raby, 和 S. Cronin, 《对地位的自恋式追求》，载于《特质自恋手册》，A. D. Hermann, A. B. Brunell, 和 J. D. Foster 编。Cham：施普林格国际出版公司，2018年，第299–306页。[54] C. C. Morf 和 F. Rhodewalt, 《揭开自恋的悖论：动态自我调节加工模型》，《心理学探究》，第12卷，第4期，第177–196页，2001年。[55] M. T. Buelow 和 A. B. Brunell, 《自恋与参与风险承担行为》，载于《特质自恋手册》，A. D. Hermann, A. B. Brunell, 和 J. D. Foster 编。Cham：施普林格国际出版公司，2018年，第233–242页。[56] V. Lenarduzzi, N. Saarim¨aki, 和 D. Taibi, 《技术债务数据集》，载于《第十五届软件工程预测模型与数据分析国际会议论文集》，L. Minku, F. Khomh, 和 J. Petri´c 编。美国纽约州纽约：ACM，2019年，第2–11页。[57] S. T. Certo, J. R. Busenbark, M. Kalm, 和 J. A. LePine, 《分则俱败：比率如何削弱战略管理研究》，《组织研究方法》，第23卷，第2期，第211–237页，2020年。[58] D. Spadini, M. Aniche, 和 A. Bacchelli, 《PyDriller：用于挖掘软件仓库的Python框架》，载于《2018年第26届ACM欧洲软件工程会议与软件工程基础研讨会联合会议论文集》，G. T. Leavens, A. Garcia, 和 C. S. P˘as˘areanu 编。美国纽约州纽约：ACM，2018年，第908–911页。[59] S. Wagner, D. Mendez, M. Felderer, D. Graziotin, 和 M. Kalinowski, 《问卷调查研究中的挑战》，载于《软件工程中的当代实证方法》，M. Felderer 和 G. H. Travassos 编。Cham：施普林格国际出版公司，2020年，第93–125页。[60] S. Baltes 和 P. Ralph, 《软件工程研究中的抽样：批判性综述与指南》，《实证软件工程》，第27卷，第4期，2022年。[61]D. Graziotin, F. Fagerholm, X. Wang, 和 P. Abrahamsson, 《软件开发人员的不快乐》，载于《第21届软件工程评估与评价国际会议论文集》，E. Mendes 编。纽约：ACM，2017年，第324–333页。[62] S. D. Gosling, 《关于TIPI中α信度与因子结构的说明》。 [在线]。网址：https://gosling.psy.ut
exas.edu/scales-weve-developed/ten-item-personality-measure-tipi/a-note-on-alpha-reliability-and-factor-structure-in-the-tipi/ [63] K. L. Haws, U. M. Dholakia, 和 W. O. Bearden, 《慢性调节焦点测量的评估》，《营销研究杂志》，第47卷，第5期，第967–982页，2010年。[64] J. M. Cortina, 《什么是α系数？对理论与应用的考察》，《应用心理学杂志》，第78卷，104，1993 1，第98页。[65] J. C. Nunnally, 《心理测量理论》，第2版，麦格劳-希尔心理学丛书。纽约和伦敦：麦格劳-希尔，1978年。

[66] P. T. Costa 和 R. R. McCrae, 《成年期人格：基于NEO人格量表自我报告和配偶评价的六年纵向研究》，《人格与社会心理学杂志》，第54卷，第5期，第853–863页，1988年。[67] R. R. McCrae 和 P. T. Costa, 《人格的稳定性：观测值与评估》，《心理科学当前方向》，第3卷，第6期，第173–175页，1994年。[68] F. Calefato 和 F. Lanubile, 《在分布式团队中建立基于个人信任的联系》，《互联网技术快报》，第1卷，第4期，e6页，2018年。[69] P. C. Rigby 和 A. E. Hassan, 《开源软件邮件列表能告诉我们什么？对Apache开发者邮件列表的初步心理测量文本分析》，载于第四届软件仓库挖掘国际研讨会（MSR'07：ICSE 2007研讨会）。IEEE，2007年，第23页。[70] B. A. Visser, 《自恋与人格大五人格/HEXACO人格模型》，载于《特质自恋手册》，A. D. Hermann、A. B. Brunell 和 J. D. Foster 编。Cham：施普林格国际出版公司，2018年，第205–212页。[71] K. J. Rothman, 《多重比较无需调整》，《流行病学》，第1卷，第1期，第43–46页，1990年。[72] R. Bender 和 S. Lange, 《多重检验的调整——何时以及如何调整？》，《临床流行病学杂志》，第54卷，349，2001年4，第343–页。[73]D. J. O'Keefe, 《讨论：是否应调整家族错误率α？反对家族错误率α调整》，《人类传播研究》，第29卷，第3期，第431–447页，2003年。[74] M. L. Anderson, 《多重推断与早期干预效果中的性别差异：对Abecedarian、Perry Preschool和早期训练项目的重新评估》，《美国统计协会杂志》，第103卷，第484期，第1481–1495页，2008年。[75] A. M. Isen, 《积极情感对复杂情境中决策的影响：具有实践意义的理论问题》，《消费者心理学杂志》，第11卷，第2期，第75–85页，2001年。[76] L. Gren, 《行为软件工程研究中的效度标准与标准的效度》，载于第12届ACM/IEEE实证软件工程与测量国际研讨会论文集，M. Oivo、D. M´endez 和 A. Mockus 编。美国纽约州纽约市：ACM，2018年，第1–4页。[77] F. L. Schmidt 和 J. E. Hunter, 《心理学研究中的测量误差：来自26个研究情境的经验》，《心理学方法》，第1卷，第2期，第199–223页，1996年。[78] D. L. Paulhus 和 S. Vazire, 《自我报告法》，载于《人格心理学研究方法手册》，R. W. Robins、R. C. Fraley 和 R. F. Krueger 编。纽约州纽约市：吉尔福德出版社，2007年，第224–239页。[79] F. C. van Mil、A. Rastogi 和 A. Zaidman, 《在GitHub上推断人格的前景与风险》，载于第15届ACM / IEEE实证软件工程与测量国际研讨会（ESEM）论文集，F. Lanubile 编。美国纽约州纽约市：ACM，2021年，第1–11页。[80] F. Calefato 和 F. Lanubile, 《将人格检测工具用于软件工程研究：我们能走多远？》，《ACM软件工程与方法学汇刊》，第31卷，48，2022年3，1期，第–页。[81] V. Lenarduzzi、A. Martini、D. Taibi 和 D. A. Tamburri, 《迈向外科手术般精确的技术债务估算：早期结果与研究路线图》，载于第三届ACM SIGSOFT软件质量评估机器学习技术国际研讨会论文集——MaLTeSQuE 2019，F. A. Fontana、B. Walter、A. Ampatzoglou、F. Palomba、G. Perrouin、M. Acher、M. Cordy 和 X. Devroey 编。美国纽约州纽约市：ACM出版社，2019年，第37–42页。[82] N. Rios、M. G. de Mendonc¸a Neto 和 R. O. Sp´ınola, 《关于技术债务的三次研究：类型、管理策略、研究趋势以及面向从业者的基础信息》，《信息与软件技术》，第102卷，第117–145页，2018年。[83] J. Lefever、Y. Cai、H. Cervantes、R. Kazman 和 H. Fang, 《关于技术债务检测工具之间缺乏共识》，载于2021年第43届IEEE/ACM国际软件工程大会：软件工程实践（ICSE-SEIP）。IEEE，2021年，第121–130页。[84] K. Beck, 《产品开发铁人三项》，2018年。[在线]。可获取自：https://medium.com/@kentbeck 7670/the-product-development-triathlon-6464e2763c46