APISENS- 面向 API 的众包知识情感评分工具

Kisub Kim新加坡管理大学
新加坡
kisubkim@smu.edu.sg

Ferdian Thung新加坡管理大学新加坡
ferdianthung@smu.edu.sg

张婷新加坡管理大学新加坡
t
ingzhang.2019
@phdcs.smu.edu.sg

伊瓦娜·克莱琳·伊尔桑
新加坡管理大学
新加坡
ivanairsan@smu.edu.sg

拉特纳迪拉·维迪亚萨里
新加坡管理大学
新加坡
ratnadiraw.2020@phdcs.smu.edu.sg

周阳
新加坡管理大学
新加坡
zyang@smu.edu.sg

David Lo新加坡管理大学
新加坡 davidlo@smu.edu.sg

arXiv:2304.05121v1 [cs.SE] 11 Apr 2023

摘要— **利用现有的软件工件（如库和应用程序编程接口（API））对于提高软件开发效率至关重要。然而，提供相似功能的软件工件数量众多，可能会让开发者感到困惑，进而难以正确地进行选择和实施。通过初步调查，我们发现，利用群体的集体知识可以极大地帮助开发者全面、深入地了解软件开发过程中涉及的复杂问题。尤其是情感作为人类本性中不可分割的一部分，会影响开发者的活动。鉴于此，我们尝试构建一种工具，用于检索软件 API 的情感信息，帮助开发者确定适用于其任务的 API。我们采用最受欢迎平台（即 Twitter 和 YouTube）上的数据集来构建研究原型。源代码、工具和演示视频可在 GitHub 上获取：https://github.com/FalconLK/APISens。**

索引词—**API 情感分析、API 理解、预训练模型**

I. 引言

过去几十年来，开源软件（OSS）迅速发展。这一现象自然推动了软件工件（例如库或框架）的日益广泛复用。由于社区对这类工件的需求极为旺盛，工件推荐系统受到了广泛关注。例如，开发者经常搜索现有源代码工件 [1], [2] ，以获取实现所需功能或维护软件 [3]所需的资源。因此，人们提出了许多自动化工件推荐系统 [4]–[9] ，以帮助开发者高效地选择工件。

对所设计的软件工件进行复用会导致另一个问题，即存在大量实现相同或相似功能的工件，这可能会使开发人员感到困惑。例如，两个 API：getData 和 retrieveInformation 能够执行相同的功能，即“请求并获取必要的数据”。作为一个更复杂的例子，decrypt 和 unscramble 的名称完全不同，但它们都用于“解码/破解某个对象以获取初始状态”。面对多个选项，开发者需要选择最合适的选项。一项研究 [10]表明，软件复用主要是由于缺乏相应工件的基础知识。这表明，开发者复用工件，是因为他们不知道选择最合适工件所需了解的信息。他们还发现，当工件需要进一步增强、依赖关系过于复杂或工件已被弃用时，就会进行重新实现。此外，许多开发者正在软件开发社区[1] 及其网站[2][3]上积极讨论如何选择合适的工件。

为了帮助开发者在选择软件制品时全面了解相关信息，众包知识被证明很有帮助 [11], [12]。尤其是情绪作为人性中不可分割的一部分，也会影响开发者的活动 [13]。软件工程领域的多项研究发现，开发者会对库 [10], 、API [14],、提交消息 [15], 、项目制品 [16], 等表达情感。然而，大多数现有推荐系统 [4]–[6], [17], [18] 在检索多个制品时，并未考虑开发者情感所反映的制品质量，只有少数方法 [5], [19] 考虑了来自 Stack Overflow 的讨论数据 [4]。

[1] https://stackoverflow.com/questions/488348/what-are-your-criteria-for-choosing-a-framework-or-library[2] https://www.lagarsoft.com/blog/8-tips-for-choosing-the-right-library[3] https://www.theserverside.com/tip/7-tips-to-choose-the-right-Java-library[4]https://stackoverflow.com/

这些发现促使我们开发了一款名为 APISENS 的工具，它是一款利用来自多种资源的众包知识（即在线开发者情感）为软件 API 进行情感评分的工具。APISENS 会检索 API 的情感分数，使用户能够了解其受欢迎程度、认知度，或公众对它的反应。

为构建该工具，我们采用了一个预训练的 Transformer 模型，即一种基于注意力机制的深度神经网络架构——BERT（来自 Transformer 的双向编码器表示） [20]。一项实证研究 [12]已证明，BERT 在情感分析方面成效显著。APISENS 由两个基于 BERT-base 的模型组成。第一个模型是识别模型，用于区分输入讨论是否处于软件制品语境，而非普通语境（例如，decrypt 和 unscramble 在任何地方都可能用于普通语境）。我们从多种在线资源（即 Twitter 和 YouTube）收集大量讨论文本数据，并使用第一个模型进行筛选。请注意，处于普通语境的讨论是训练 API 情感分析模型时的噪声。此外，我们认为，能够识别与软件制品相关讨论的识别模型，在遇到包含 API 相关内容的讨论时，也能够提取出与 API 相关的讨论。我们将第一个模型的筛选结果称为与 API 相关的讨论。第二个模型会为输入的 API 推断分数。该模型使用与 API 相关的讨论进行微调，通过分析相关情感来获取 API 的分数。简而言之，APISENS 将 API 名称作为文本输入，并为该 API 返回一个综合的归一化情感分数。

总而言之，本文的贡献如下：

- 用于微调 BERT 模型的资源多样化（即 Twitter 和 YouTube），以推断情感分数，为用户提供全面的 API 评分。
- 构建了一种工具，可为软件 API 提供包括情感和受欢迎程度在内的多项评分，帮助开发者为其任务确定更合适的 API。

II. 相关工作

A. 软件工程中的情感分析

软件工程中的情感分析是对开发者针对各种软件工件的不同观点进行的计算研究。其中大多数研究将情感分析视为极性分类。给定一段文本，研究目标是预测其情感倾向属于正面、中性或负面。我们介绍该主题的代表性方法。

Stanford CoreNLP [21]旨在进行单句分类；它使用斯坦福情感树库 [21]进行训练，训练模型为递归神经张量网络 [22] 。SentiStrength [23] 是一种基于词典的方法，使用多个词典，其中包括正式和非正式术语，同时每个术语都标注了情感强度。SentiStrength- SE [24] 包含一个领域专用词典，该词典基于对前一种方法结果的深入调查构建而成。SentiCR [25]旨在用于代码审查评论数。结果显示，GradientBoost- ingTree [26] 是最适合其数据的模型。Senti4SD [27] 是一个监督模型，利用了基于以下三种内容的特征：(1) 通用情感词典 [23]；(2) 关键词（出现次数）；(3)在 Stack Overflow 数据上专门训练的分布语义模型（DSM）中的词表示。最近，Zhang 等人 [12]开展了一项研究，用于评估这些情感分析工具以及在软件工程领域经常使用的大型预训练模型的性能。

B. 基于深度学习的 *API* 推荐

近期的 API 推荐方法利用神经网络学习并识别 API 使用和文档中的模式。这些系统可以使用包含过往 API 使用情况和文档的数据集进行训练，从而为未来的 API 选择提供有依据的推荐。

DeepAPI [6] 是首个用于 API 推荐的深度学习模型，实现了端到端的 API 序列生成。该方法将 API 推荐视为机器翻译问题，并使用循环神经网络（RNN）编码器-解码器模型将查询编码为上下文向量。随后，它根据查询的上下文向量推荐 API 序列。API2Vec [28] 利用无监督学习和词嵌入技术生成嵌入，并根据 API 与给定上下文之间的语义相似性推荐 API。Huang 等人 [19] 提出了 BIKER，通过利用词袋嵌入优化 API 选择，并根据候选 API 与 Stack Overflow 帖子的匹配程度筛选 API。CLEAR [5] 也利用 Stack Overflow 帖子和 BERT 句子嵌入，以保留查询和帖子中的语义信息。此外，它还采用对比学习 [29] 来区分语义上不相似但词汇上相似的查询。尽管 BIKER 和 CLEAR 以利用 Stack Overflow 帖子的方式考虑了众包知识，但它们只采用了与目标 API 相关的术语。因此，它们仍未考虑开发者情感。

III. APISENS 详解

APISENS 接受 API 名称，并检索两类情感分数：一类来自讨论，另一类来自视频。如图 1 所示，该工具由以下三个组件组成：(1) APIRecognizer、(2) DiscussionAnalyzer，以及 (3) 视频分析器。假设在线平台的数据是通过相应的 API（例如 GitHub REST API[5]）抓取的，APISENS 必须以简单的 API 名称形式作为查询。给定 API 名称作为查询后，APISENS 经过预训练的 API 识别器会识别与 API 相关的讨论（即该工具中的推文）。检测到相关讨论后，讨论分析器将其作为输入，以检索情感分数。与此同时，视频分析器从多个方面（即“点赞数”、“评论数”和“观看次数”）分析与目标 API 对应的视频中的情感。

[5]https://docs.github.com/en/rest?apiVersion=2022-11-28

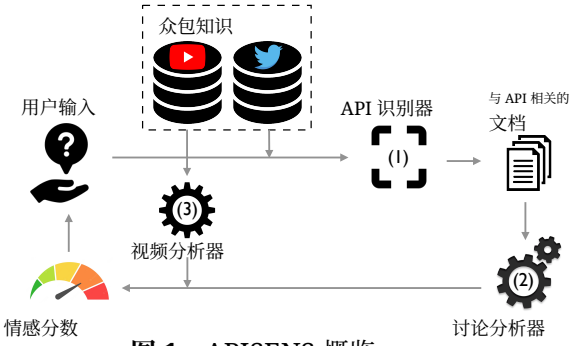


图 1： APISENS 概览。

最后，用户界面会显示结果。图 1 展示了分数检索过程中展开的各个步骤。

A. 数据集（众包知识）

我们从两个大型平台（即 Twitter 和 YouTube）收集数据集。Twitter 是最大的社交网络平台之一，情感分析工具可以从多个方面受益。Twitter 拥有庞大的用户群，其中包括开发者，能够广泛反映人们对软件 API 的情感；同时，其内容丰富多样的讨论也能确保该工具处理各种情感。Twitter 上的数据持续更新，有助于从业者及时掌握最新信息。YouTube 是最大的视频分享平台，内容涵盖众多不同主题，因此是情感分析的宝贵数据来源。许多人依赖 YouTube 获取有关产品、服务乃至软件 API 的信息和观点。

为了从 Twitter 和 YouTube 获取 API 数据集，我们同时从这两个平台收集数据。由于推文可能使用各种语言书写，并可能包含各种形式的噪声，例如转推、提及其他用户、URL 和表情符号，因此在收集之前会对数据进行预处理。预处理步骤会将表情符号转换为相应的文本表示，并通过计算每个 API 的平均推文数来筛选 API 及其对应的推文（即，相关推文数量低于平均值的 API 会被剔除）。这有助于防止局部泛化，因为纳入仅有一条推文的 API 的推文可能会导致偏差。随后，使用 Python 库（即 langdetect）移除重复推文以及非英语推文。通过选取概率最高的语言即可进行语言检测。视频数据集包含标题、描述、“点赞数”、“评论数”和“观看次数”。部分视频可能不包含情感分析所需的信息统计数据，因此也会作为进一步剔除的候选项。我们的研究原型最初以 1409 个 JDK API 为目标，并收集了 56,646 条推文。数据处理后，推文数量为 28,278 条，对应 476 个 API。虽然爬虫可以实时运行，APISENS 也可以分析动态数据，但我们的研究原型假定数据集已收集完毕。

B. API 识别器

我们的 API 识别器由预训练的 BERT [20]之上的一个全连接层组成，因为事实证明它在推文 [30]中的软件库识别任务上非常有效。我们在每条推文前加上 [CLS] 标记，作为输入传递给BERT。 [CLS] 的最终隐藏状态被视为该推文的聚合表示。我们使用从 [CLS]输出的嵌入作为分类器的输入。我们的 API 识别器使用 AdamW 优化器 [31] 进行训练，并使用线性学习率调度器。

所采用的模型用于预测推文是否与软件工件相关。该模型最初使用库数据集进行训练（即包含 23 个库的 4,456 条推文）。不过，这类识别模型所学习的关键特征（即与软件工件相关的词元）对 API 而言并无不同。此外，文献 [30] 表明，混合设置（即使用涵盖不同库的混合数据集）效果最佳，F1 分数达到 90%。因此，采用相同设置的模型适合用于分类与我们的目标 API 相关的讨论。根据预测结果，作者手动确认标签，以提高准确性。

此外，我们对视频数据集进行分类，因为其中可能存在噪声，例如，与 unscramble 等普通词语相关的视频可能涉及任何主题。由于我们有视频标题和描述，因此采用相同的流程来过滤噪声。经过视频筛选，目标 API 的数量从 476 个减少到 235 个，相关推文数量也降至 1,606 条。

C. 讨论分析器

给定与 API 相关的讨论数据集，该组件会执行情感分析，以获取用于检索的分数。我们再次采用同一个预训练 BERT 模型，该模型也应用于一种最先进的情感分析方法中 [12]。文献中的实验结果表明，BERT 的表现优于其他专门针对软件工程下游任务中的情感分析而设计的先进技术（即 BERT 的 Micro-avg 达到 89%，比排名第二的模型高出 7 个百分点 [12]）。用于训练的 API 数据集包含 4,522 条与软件 API 相关的 Stack Overflow 帖子。由于模型会在正面、中性和负面中预测类别，我们分别将其转换为整数 10、5 和 0，以计算最高为 10 的平均分。最后，我们整合结果，并计算输入 API 对应的每条推文的平均情感分数，从而提供全面而简洁的评分。

D. 视频分析器

APISENS 根据由 API 识别器通过标题和描述筛选出的与 API 相关的视频列表，提取相关统计数据，以更全面地辅助情感判断。这些统计数据包括每个视频的点赞数、评论数和观看次数，并用于确定内容（即输入的 API）的情感倾向。这些视频数据可反映公众对该

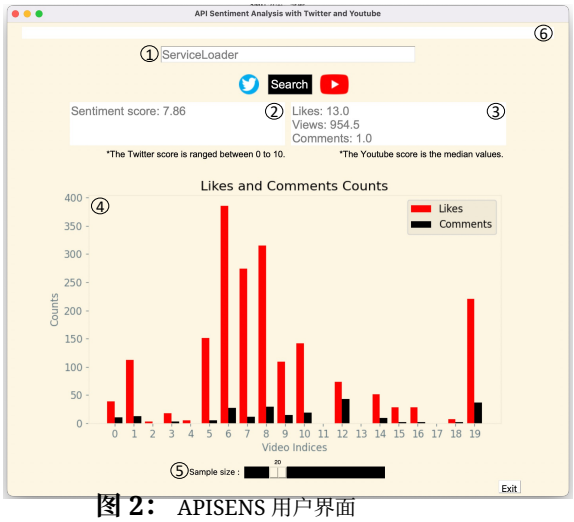


图 2： APISENS 用户界面

视频的情绪。点赞数和观看次数较高，意味着视频受欢迎且制作精良；评论数较高，则能让我们了解视频引发的讨论类型和反应。我们认为这些数据有助于情感分析，因为它们直接反映了受众的观点，而这些观点可能带有情绪倾向。为了更准确地反映结果集的集中趋势，同时避免离群值的影响，我们决定在用户界面中提供各项指标的中位数。

IV. 原型说明与评估 ION

A. 原型说明

APISENS 的用户界面设计直观且易于使用。图 2 展示了该用户界面。主窗口分为三个主要区域：顶部的查询面板、中央的结果展示面板，以及底部的图表面板。查询面板（①）将 API 名称作为用户查询。点击搜索 按钮后，系统会分析讨论情绪，获取评分并显示在结果展示面板（②）中。同时，系统还会提供 ③ 中相关视频的情绪信息。如前所述，APISENS 会计算与输入 API 相关的已收集视频的中位数。此外，图表面板中的柱状图（④）会展示一定数量的与相应 API 相关视频的点赞数和评论数统计信息。这有助于用户了解相关视频点赞数和评论数的实际变化趋势。用户可以通过滑动条（⑤）重新采样视频数量，以获得更明显的变化。顶部的进度条（⑥）用于显示进度。

B. 基于用户体验的可能应用场景

为了支持从业者，我们还开展了一项用户研究/讨论，以评估 APISENS 的实用性。我们邀请了2名研究科学家和 2名软件工程博士生，他们都有超过 5 年的编程经验。为了获取他们的意见，我们首先向参与者提供 APISENS以及 20个 API 样本，并让参与者进行评分（即从 1 到 5，其中 1 表示强烈不同意，5 表示强烈（同意）其可用性和实用性，并讨论其潜力。总体而言，参与者认为 APISENS 易于使用（满分 5 分，得 5 分）且实用（满分 5 分，得 4.05 分）。我们通过用户讨论确定并确立了这些有前景的用例。

一眼了解 API 的受欢迎程度。APISENS 的主要应用场景确实是获取 API 的情感分数，让用户了解该 API 是否得到广泛使用和良好支持。此外，它还能帮助用户了解该 API 的需求程度，并在比较不同 API 或考虑潜在替代方案时提供有价值的背景信息和视角。**将其集成到 IDE 中**。情感信息可用于根据个别开发者的偏好和需求，为其提供个性化的 API 推荐。因此，这种集成可以提供更多选择。情感信息还可作为优先级排序的依据，帮助 IDE 实时提供 API 建议，从而改善代码补全功能。

与序列推荐技术结合。将 APISENS 与 API 序列推荐相结合后，可以发挥协同效应，例如优先推荐或重新排序情感分数为正面的 API。这有望带来更好的用户体验。

情感信息作为学习特征。情感信息可以作为机器学习模型中的特征，以提高 API推荐的准确性。例如，由于 API情感与 API 的受欢迎程度或使用情况之间存在很强的相关性，将此类信息作为模型中的特征可能会提升性能。

V. 进一步构想

基于我们的观察以及用户的早期反馈，我们认为 APISENS 有望成为一个实用的 API 理解工具。以下是一些进一步改进的设想：**支持并发处理多个 API**。开发者往往会比较多个 API，以便为特定任务找到最佳 API，或及时了解行业趋势。这有助于他们掌握新的 API 和最佳实践，并确保使用最受欢迎、最有效的 API。**使用更大的数据集训练模型**。为了最大限度地提升工具性能，可以使用更大的数据集对预训练模型进行微调。更大、更优质的数据集能够提升模型的统计效力和泛化能力，因此可以进一步微调 APISENS，以获得更可靠的结果。此外，涵盖更多 API 也有助于开发者更广泛地使用我们的工具。**进一步支持功能相同的相似 API**。 APISENS 可以加入一项功能，支持查询相似的 API，并提供与所查询 API 相对应的情感信息。这可以进一步提高效率，并简化 API 的比较和筛选过程，帮助用户确定更合适的 API。

参考文献

[1] S. E. Sim、M. Umarji、S. Ratanotayanon 和 C. V. Lopes，“搜索引擎在支持网络代码检索方面的表现如何？”计算机协会会刊

软件工程与方法论，第 21 卷，第 25 期，2011 年，第 1–4 页。[2] R. E. Gallardo-Valencia 和 S. Elliott Sim，“互联网规模的代码搜索”，载于 *2009* 年 *ICSE* 搜索驱动开发研讨会论文集：用户、基础设施、工具与评估。电气与电子工程师学会计算机协会，2009 年，第 49–52 页。[3] X. Xia、L. Bao、D. Lo、P. S. Kochhar、A. E. Hassan 和 Z. Xing，“开发者在网络上搜索什么？”《实证软件工程》，第 22 卷，第 6 期，第 3149–3185 页，2017 年。[4] F. Thung、D. Lo 和 J. Lawall，“自动化库推荐”，载于*2013* 年第 *20* 届逆向工程工作会议（*WCRE*）。电气与电子工程师学会，2013 年，第 182–191 页。[5] M. Wei、N. S. Harzevili、Y. Huang、J. Wang 和 S. Wang，“CLEAR：面向 API 推荐的对比学习”，载于第*44*届国际软件工程会议论文集，2022 年，第 376–387 页。[6] X. Gu、H. Zhang、D. Zhang 和 S. Kim，“深度 API 学习”，载于*2016* 年第 *24* 届计算机协会软件工程基金会国际研讨会论文集，2016 年，第 631–642 页。[7] Q. Huang、X. Xia、Z. Xing、D. Lo 和 X. Wang，“无需担心任务与 API 知识鸿沟的 API 方法推荐”，载于*2018* 年第 *33* 届电气与电子工程师学会/计算机协会自动化软件工程国际会议（*ASE*）。电气与电子工程师学会，2018 年，第 293–304 页。[8] X. Liu、L. Huang 和 V. Ng，“无需历史软件仓库的有效 API 推荐”，载于第*33*届计算机协会/电气与电子工程师学会自动化软件工程国际会议论文集，2018 年，第 282–292 页。[9] A. T. Nguyen、M. Hilton、M. Codoban、H. A. Nguyen、L. Mast、E. Rademacher、T. N. Nguyen 和 D. Dig，“利用细粒度变更中的统计学习进行 API 代码推荐”，载于*2016* 年第 *24* 届计算机协会软件工程基金会国际研讨会论文集，2016 年，第 511–522 页。[10] B. Xu、L. An、F. Thung、F. Khomh 和 D. Lo，“为何重复造轮子？关于库复用与重新实现的实证研究”，《实证软件工程》，第 25 卷，第 1 期，第 755–789 页，2020 年。[11] E. Biswas、M. E. Karabulut、L. Pollock 和 K. Vijay-Shanker，“使用 BERT 在软件工程领域实现可靠的情感分析”，载于*2020* 年电气与电子工程师学会国际软件维护与演化会议（*ICSME*）。电气与电子工程师学会，2020 年，第 162–173 页。[12] T. Zhang、B. Xu、F. Thung、S. A. Haryono、D. Lo 和 L. Jiang，“软件工程情感分析：预训练 Transformer 模型能走多远？”，载于*2020* 年电气与电子工程师学会国际软件维护与演化会议（*ICSME*）。电气与电子工程师学会，2020 年，第 70–80 页。[13] M. De Choudhury 和 S. Counts，“通过社交媒体理解职场中的情感”，载于*2013* 年计算机支持协同工作会议论文集，2013 年，第 303–316 页。[14] G. Uddin、O. Baysal、L. Guerrouj 和 F. Khomh，“理解开发者如何以及为何寻求并分析与 API 相关的观点”，电气与电子工程师学会软件工程汇刊，第 47 卷，第 4 期，第 694–735 页，2019 年。[15] E. Guzman、D. Az´ocar 和 Y. Li，“GitHub 提交评论的情感分析：一项实证研究”，载于第*11*届软件仓库挖掘工作会议论文集，2014 年，第 352–355 页。[16] A. Murgia、P. Tourani、B. Adams 和 M. Ortu，“开发者会感受到情绪吗？对软件工件中情绪的探索性分析”，

发表于<style id='1">第11届软件仓库挖掘工作会议论文集，2014年，第262–271页。[17] X. Gu、H. Zhang 和 S. Kim，“深度代码搜索”，发表于*2018*年电气与电子工程师学会/计算机协会第*40*届国际软件工程会议（*ICSE*）。电气与电子工程师学会，2018年，第933–944页。[18] K. Kim、D. Kim、T. F. Bissyand´e、E. Choi、L. Li、J. Klein 和 Y. L. Traon，“Facoy：一种代码到代码搜索引擎”，发表于第*40*届国际软件工程会议论文集，2018年，第946–957页。[19] L. Cai、H. Wang、Q. Huang、X. Xia、Z. Xing 和 D. Lo，“Biker：一种基于双信息源的 API 方法推荐工具”，发表于*2019*年第*27*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，2019年，第1075–1079页。[20] J. Devlin、M.-W. Chang、K. Lee 和 K. Toutanova，“Bert：用于语言理解的深度双向变换器预训练”，*arXiv* 预印本 *arXiv:1810.04805*，2018年。[21] R. Socher、A. Perelygin、J. Wu、J. Chuang、C. D. Manning、A. Y. Ng 和 C. Potts，“情感树库上的语义组合性递归深度模型”，发表于*2013*年自然语言处理实证方法会议论文集，2013年，第1631–1642页。[22] R. K. Bakshi、N. Kaur、R. Kaur 和 G. Kaur，“意见挖掘与情感分析”，发表于*2016*年第*3*届可持续全球发展计算国际会议（*INDIACom*）。电气与电子工程师学会，2016年，第452–455页。[23] M. Thelwall、K. Buckley、G. Paltoglou、D. Cai 和 A. Kappas，“简短非正式文本中的情感强度检测”，《美国信息科学与技术学会杂志》，第61卷，第12期，第2544–2558页，2010年。[24] M. R. Islam 和 M. F. Zibran，“在软件工程中利用自动化情感分析”，发表于*2017*年电气与电子工程师学会/计算机协会第*14*届国际软件仓库挖掘会议（*MSR*）。电气与电子工程师学会，2017年，第203–214页。[25] T. Ahmed、A. Bosu、A. Iqbal 和 S. Rahimi，“Senticr：用于代码审查交互的定制情感分析工具”，发表于*2017*年第*32*届电气与电子工程师学会/计算机协会国际自动化软件工程会议（*ASE*）。电气与电子工程师学会，2017年，第106–111页。[26] M. Pennacchiotti 和 A.-M. Popescu，“民主党人、共和党人和星巴克爱好者：Twitter 上的用户分类”，发表于第*17*届 *ACM* 数据挖掘特别兴趣小组知识发现与数据挖掘国际会议论文集，2011年，第430–438页。[27] F. Calefato、F. Lanubile、F. Maiorano 和 N. Novielli，“软件开发中的情感极性检测”，《实证软件工程》，第23卷，第3期，第1352–1382页，2018年。[28] T. D. Nguyen、A. T. Nguyen、H. D. Phan 和 T. N. Nguyen，“探索用于 API 用法及应用的 API 嵌入”，发表于*2017*年电气与电子工程师学会/计算机协会第*39*届国际软件工程会议（*ICSE*）。电气与电子工程师学会，2017年，第438–449页。[29] T. Chen、S. Kornblith、M. Norouzi 和 G. Hinton，“用于视觉表征对比学习的简单框架”，发表于国际机器学习会议。机器学习研究论文集，2020年，第1597–1607页。[30] T. Zhang、D. P. Chandrasekaran、F. Thung 和 D. Lo，“推文中的库识别基准测试”，发表于*2022*年电气与电子工程师学会/计算机协会第*30*届国际程序理解会议（*ICPC*），2022年，第343–353页。[31] I. Loshchilov 和 F. Hutter，“解耦权重衰减正则化”，*arXiv* 预印本 *arXiv:1711.05101*，2017年。