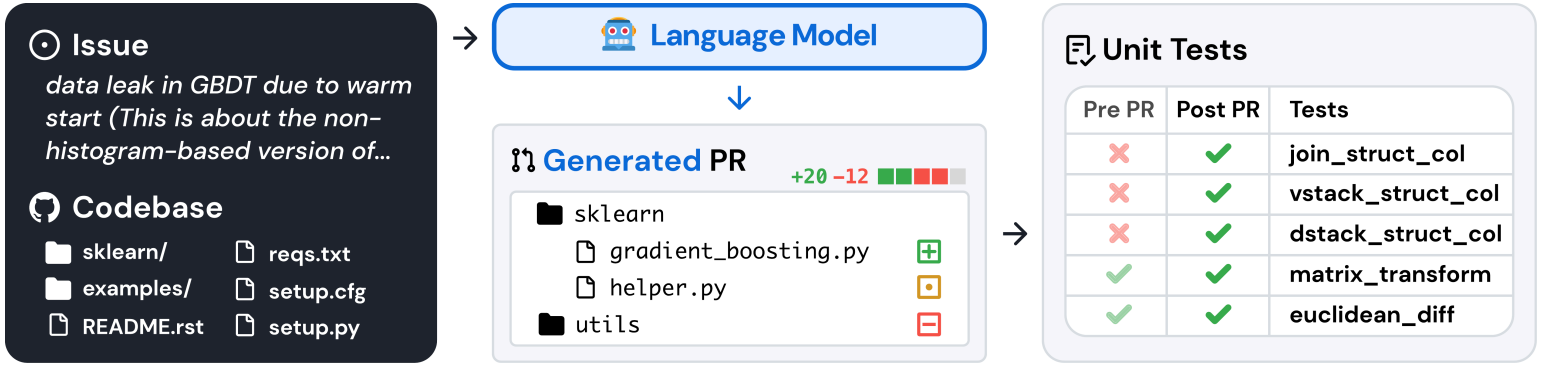
软件工程 Agent 的训练与验证

张恒睿 2601210115

SWE-Bench · SWE-Agent · OpenHands · SWE-Gym · R2E-Gym · SWE-RL · DeepSWE

SWE-Bench 的任务与评测

原始 SWE-Bench：12 个 Python 仓库，2,294 个已验证任务



Issue → 仓库修改 → 执行测试

- 输入：Issue 描述与修复前的代码库
- 输出：应用到旧代码库上的补丁

类型	修复前	候选修复后要求
F2P	失败	通过
P2P	通过	保持通过

参考文献：[1] Jimenez et al. SWE-bench: Can Language Models Resolve Real-World GitHub Issues? ICLR 2024.

模型如何完成一道 SWE-Bench 任务?

01 读 Issue
报错现象
期望行为

→ 02 搜索文件
查文件名
函数、类与调用

→ 03 定位与复现
阅读相关实现
运行复现示例

→ 04 编辑代码
修改关键逻辑
形成代码差异

→ 05 运行检查
复现脚本
仓库现有测试

→ 06 提交补丁
输出候选 diff
交给评测器

BM25：用 Issue 检索文件

以 Issue 为查询，按词项匹配给仓库文件打分，依排名与 token 预算选择文件。

oracle：已知参考修复涉及的文件

选取人类补丁修改文件的修复前内容，模型只看到旧代码，参考补丁不进入输入。

参考文献：[1] Jimenez et al. SWE-bench: Can Language Models Resolve Real-World GitHub Issues? ICLR 2024.
[2] Yang et al. SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering. NeurIPS 2024.

检索与上下文长度

模型	13k	27k	50k
Claude 2*	1.96	1.87	1.22
SWE-Llama 7B	0.7	0.31	0
SWE-Llama 13B	0.7	0.48	0

BM25 上下文长度实验：解决率（%）

* Claude 2：BM25 最佳 1.96%，oracle 文件设置 4.80%。两者的输入长度设置不同。



Figure 5：Claude 2，蓝色为解决，橙色为补丁应用

左图横轴：完整输入的 token 数（Issue、代码等）

右图横轴：仅 Issue 描述的 token 数

参考文献：[1] Jimenez et al. SWE-bench: Can Language Models Resolve Real-World GitHub Issues? ICLR 2024.

SWE-Agent 具体能做哪些动作?

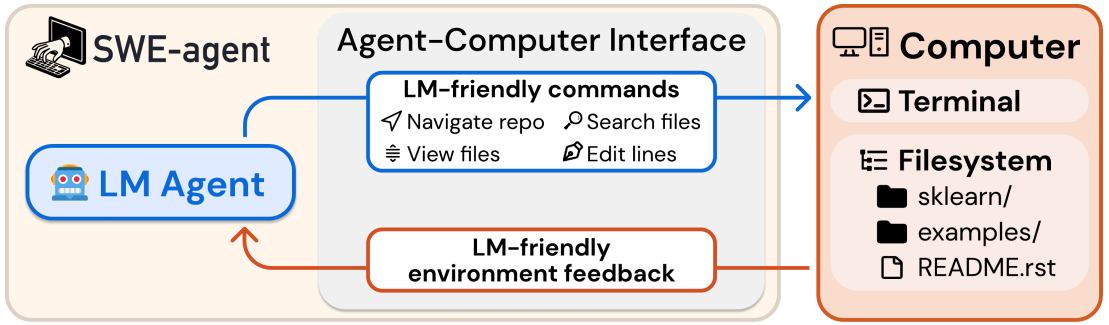
10 个专用 ACI 命令，每次动作的执行结果进入下一轮上下文

目的	动作	反馈
定位	find_file / search_dir / search_file	查文件名、目录内文本、文件内文本
查看	open / goto / scroll_up / scroll_down	返回带行号的代码窗口，默认 100 行
创建	create	创建并打开新文件，如复现脚本
编辑	edit n:m ... end_of_edit	显示新代码，引入 lint 错误时回退
执行（Bash）	python / python3 / pytest	返回程序输出、异常与测试结果
提交	submit	从当前改动生成补丁并结束轨迹

参考文献：[2] Yang et al. SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering. NeurIPS 2024.

SWE-Agent 的交互接口

- 模型读取历史，生成思考与动作
- ACI 设计搜索、查看、编辑与执行反馈
- 编辑后显示新内容，lint 检查新错误

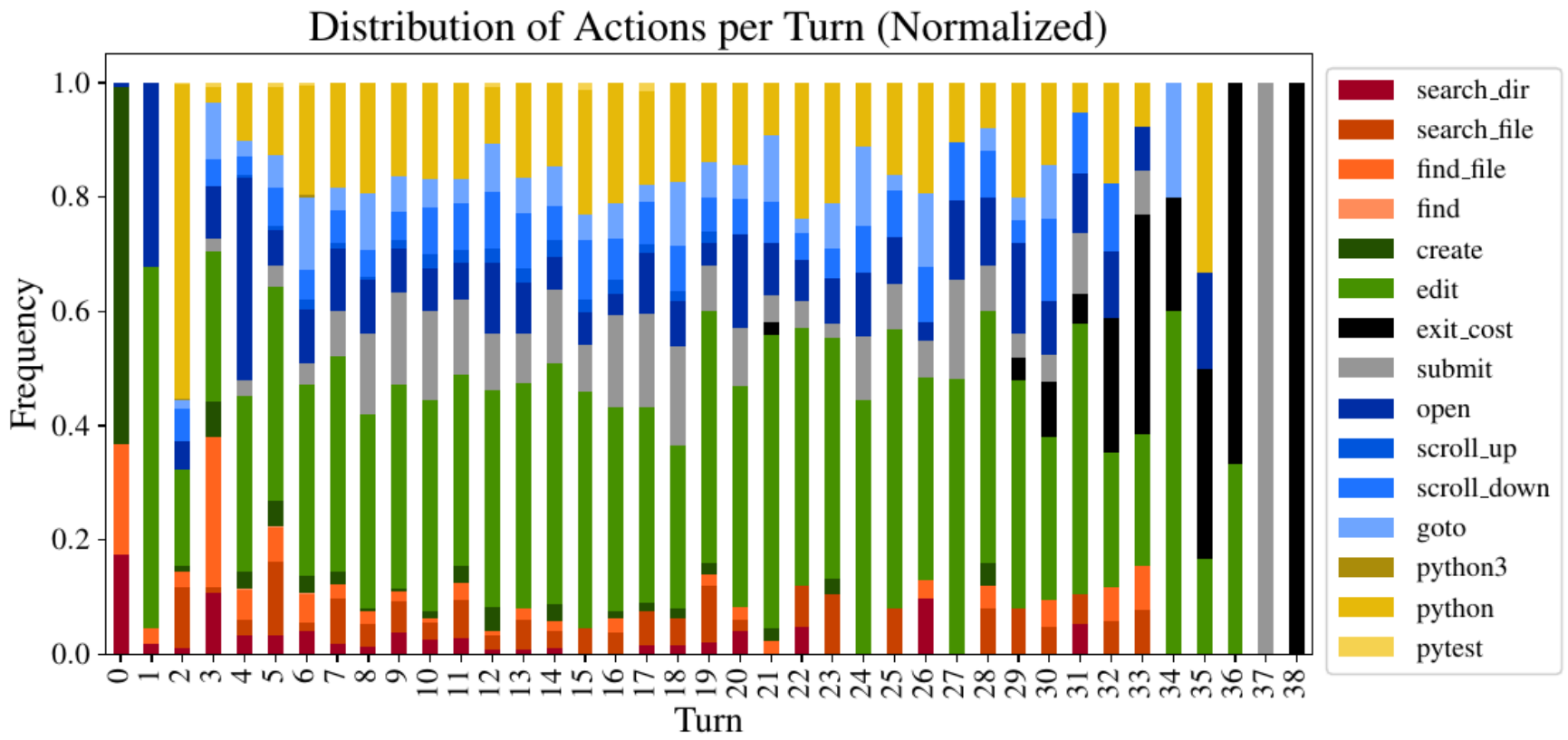


模型通过 ACI 控制搜索、编辑和执行

系统	解决率（%）	平均费用（美元）
RAG BM25 按 Issue 检索文件，模型一次生成补丁	2.67	0.13
Shell-only Agent 模型多轮调用标准 Bash，搜索、修改和执行代码	11	1.46
SWE-Agent 模型多轮使用专用 ACI，搜索、查看、编辑和执行代码	18	1.67

参考文献：[2] Yang et al. SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering. NeurIPS 2024.

成功轨迹中的动作比例



附录 Figure 18：横轴从 0 开始，分母是仍进行到该轮的成功轨迹数

参考文献：[2] Yang et al. SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering. NeurIPS 2024.

搜索、编辑与查看窗口消融

GPT-4 Turbo · SWE-bench Lite · 默认查看窗口 100 行

摘要式搜索

目录搜索列出命中文件及匹配数，文件内列出匹配行与行号。超过 50 条结果时提示缩小查询。

edit：指定行区间替换

edit n:m 用新文本替换当前文件第 n 到 m 行，并显示更新后的代码窗口。引入特定 lint 错误时回退。

消融改动	具体做法	解决率（%）	下降（百分点）
默认 ACI	**摘要式搜索，专用 edit 编辑命令**	**18**	**0**
逐项浏览搜索命中	一次显示一处，用 next / prev 翻页	12	6
移除 edit 命令	用 Shell 重定向覆写文件，或用 sed 修改	10.3	7.7
关闭编辑 lint 防护	不再拦截编辑引入的 lint 错误	15	3
查看窗口缩为 30 行	默认 100 行改为 30 行	14.3	3.7
查看窗口扩大为全文件	默认 100 行改为显示整个文件	12.7	5.3

参考文献：[2] Yang et al. SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering. NeurIPS 2024.

独立尝试与 Pass@k

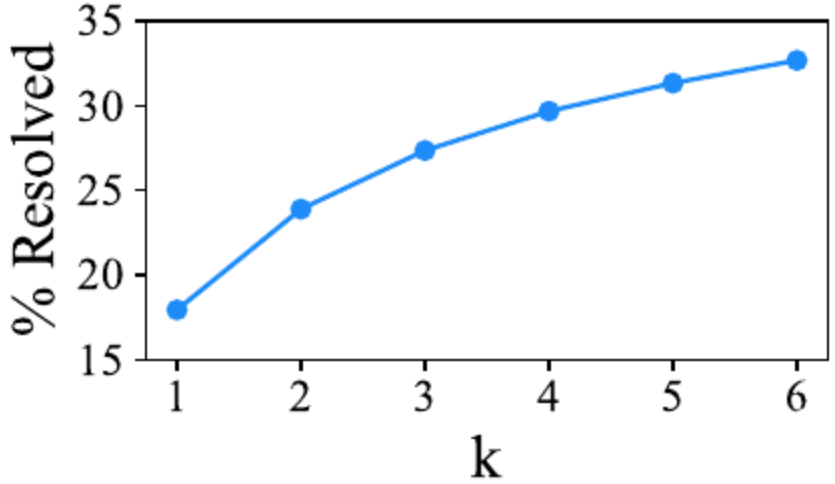
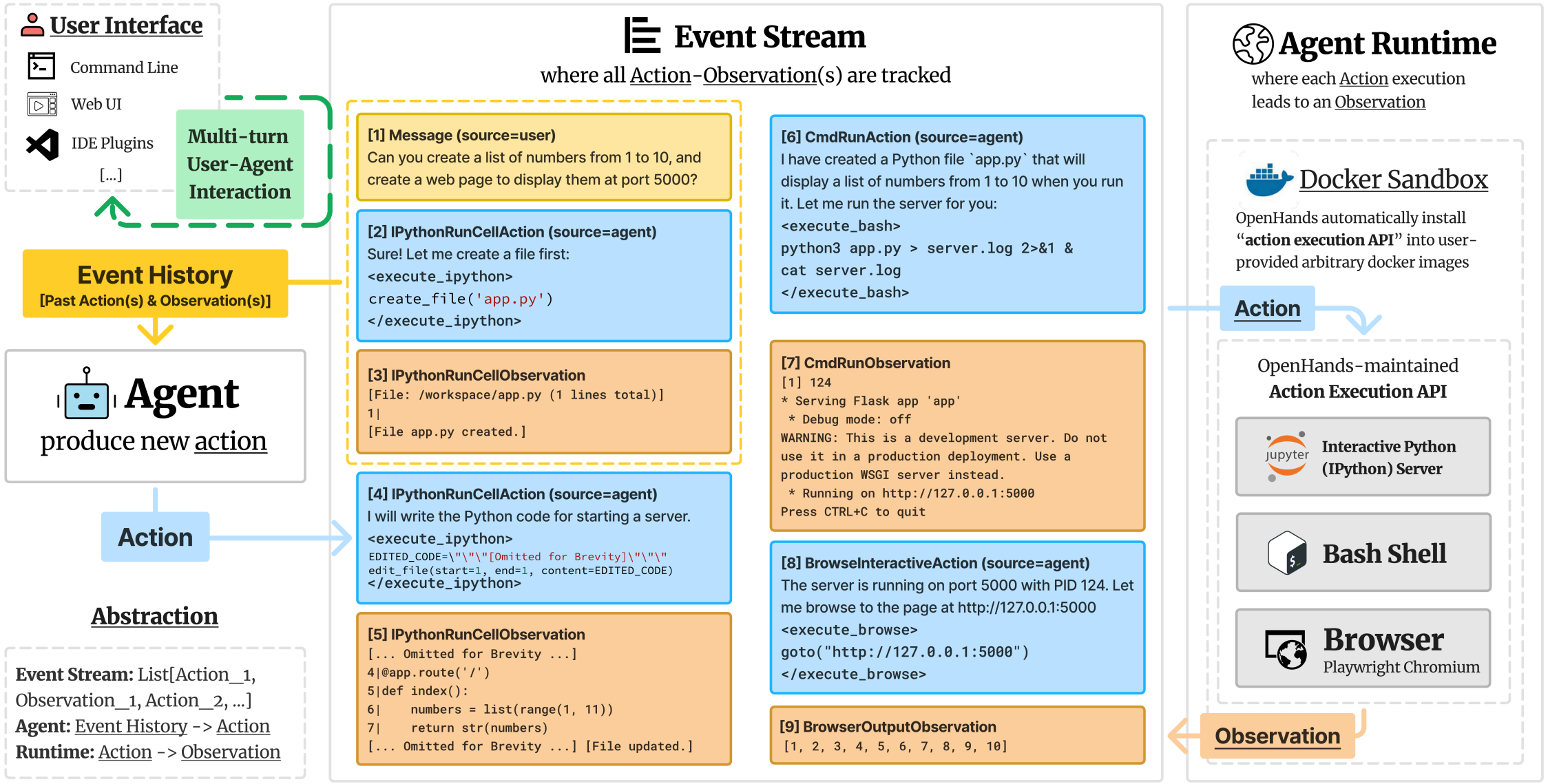


Figure 4：独立尝试的 Pass@k 曲线

- 6 次独立完整运行，每次从初始仓库开始
- Pass@1：17.94%，Pass@6：32.67%
- Pass@k：k 次中至少一次通过测试

参考文献：[2] Yang et al. SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering. NeurIPS 2024.



参考文献：[3] Wang et al. OpenHands: An Open Platform for AI Software Developers as Generalist Agents. ICLR 2025.

SWE-Gym 的可执行训练环境

训练环境与训练示范，是两个层次

层次	内容	规模
可执行任务	**Issue + 旧仓库 + 环境 + 测试**	**2,438 / 11 仓库**
教师示范	动作—观察交互，由测试筛选成功	491 条，平均约 19 轮

- GPT-4o / Claude 3.5 Sonnet + OpenHands CodeActAgent 2.1
- 保留通过测试的思考与动作，环境输出作为上下文
- 使用成功轨迹进行 SFT

参考文献：[4] Pan et al. Training Software Engineering Agents and Verifiers with SWE-Gym. ICML 2025.

成功轨迹 SFT 的训练收益

Qwen2.5-Coder-Instruct + OpenHands • SWE-bench Verified

基模	训练前（%）	轨迹 SFT 后（%）	变化（百分点）
7B	1.8	10.6	8.8
14B	4	16.4	12.4
32B	**7**	**20.6**	**13.6**

32B 行为指标	训练前	SFT 后
连续重复动作循环率	29.4%	23.8%
空补丁率	9.5%	13.8%

参考文献：[4] Pan et al. Training Software Engineering Agents and Verifiers with SWE-Gym. ICML 2025.

训练数据的数量与覆盖

Qwen 7B / 32B · Verified · 训练成功轨迹的子采样

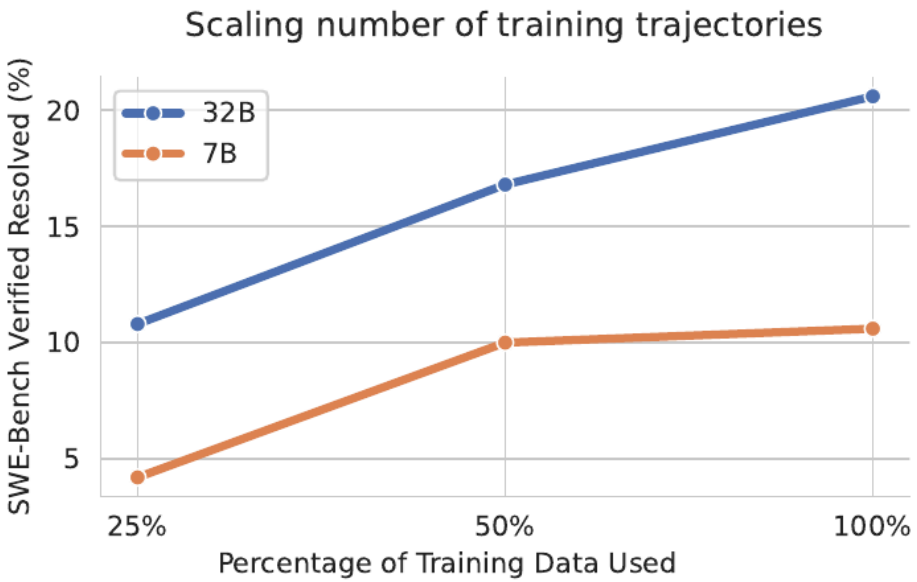


Figure 5：随机增加训练轨迹

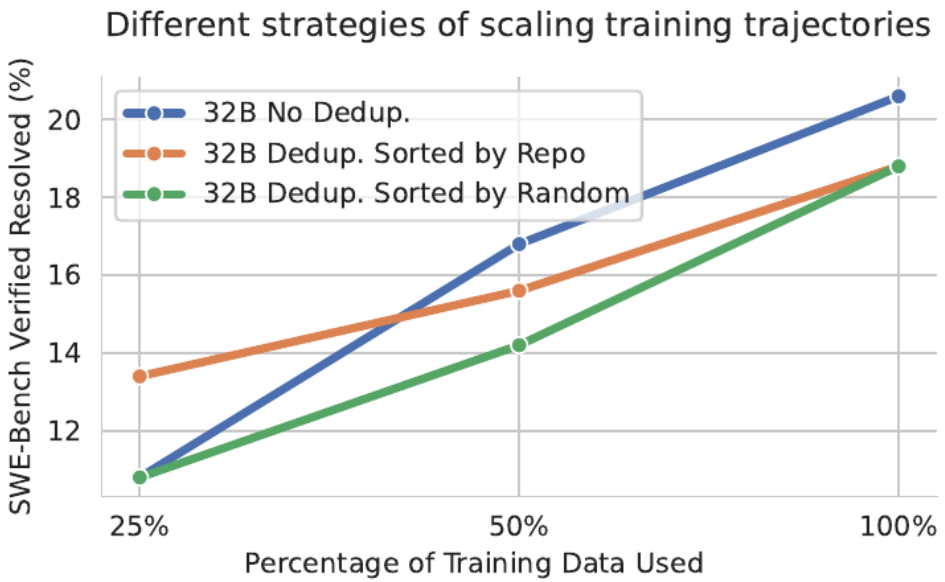


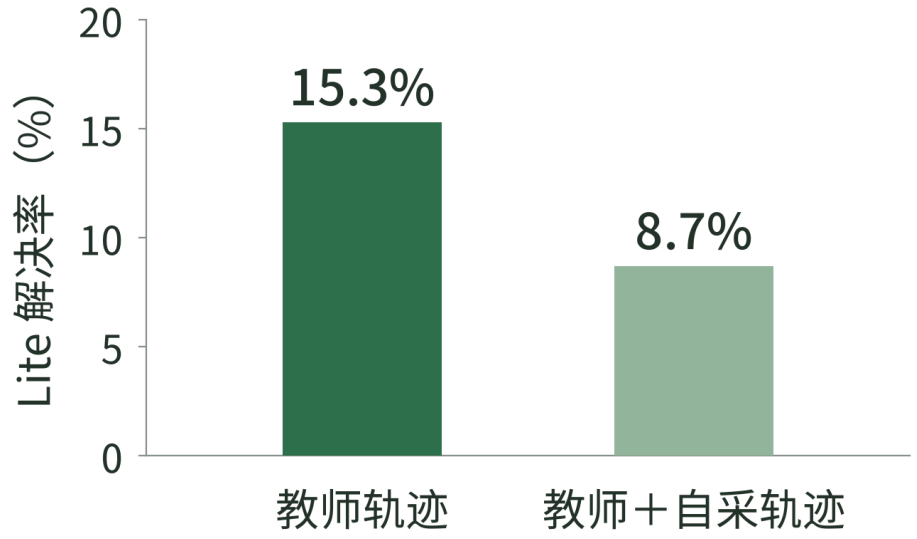
Figure 9：三种轨迹采样方式

右图 32B：蓝线不去重、随机抽轨迹，绿线每题一条后随机抽，橙线每题一条后按仓库排序抽。

参考文献：[4] Pan et al. Training Software Engineering Agents and Verifiers with SWE-Gym. ICML 2025.

自采成功轨迹的训练结果

Qwen2.5-Coder-Instruct-32B + OpenHands · SWE-bench Lite



根据 §4.2 报告结果绘制：491 条 vs 491 + 868 条

- 教师 SFT 后的 Agent 在每题采样 6 次
- 运行测试，筛出 868 条自采成功轨迹
- 同一 32B 基模做 SFT：仅教师 vs 教师 + 自采

参考文献：[4] Pan et al. Training Software Engineering Agents and Verifiers with SWE-Gym. ICML 2025.

候选覆盖与验证器选择

Qwen2.5-Coder-32B 策略 + 32B 验证器 · SWE-bench Verified

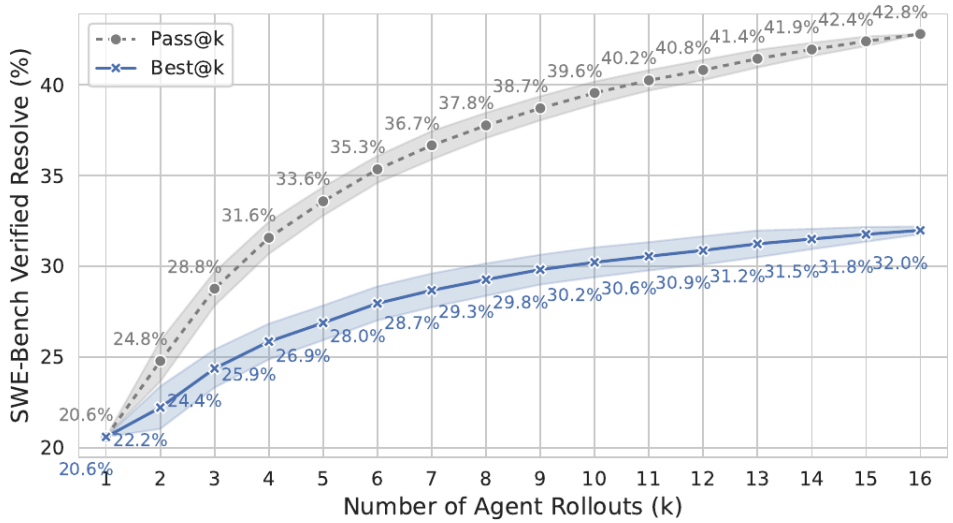


Figure 3：Verified 的 Pass@k 与 Best@k

- 验证器读取完整轨迹，预测成功概率
- Best@k：按预测分数选一条，再运行评测
- Pass@k：k 个候选中至少一条通过评测
- Best@16 32.0%，Pass@16 42.8%

参考文献：[4] Pan et al. Training Software Engineering Agents and Verifiers with SWE-Gym. ICML 2025.

SweGen 的任务构建

先验证真实代码修改，再根据失败行为反向生成 Issue

真实 commit
修复前与修复后代码
参考补丁、提交消息

→

测试与双版本验证
复用或生成测试
旧代码：Fail
修复后：Pass

→

反向生成 Issue
结合补丁、测试
与双版本执行日志
生成缺陷描述

→

可执行任务
旧代码环境
合成 Issue
测试与评价函数

规模	SWE-Gym	R2E-Gym
可执行环境／任务	2,438	完整集 8,135，训练子集 4,578
成功轨迹覆盖的任务	294	2,048
成功训练轨迹	491	3,321

参考文献：[4] Pan et al. Training Software Engineering Agents and Verifiers with SWE-Gym. ICML 2025.
[5] Jain et al. R2E-Gym: Procedural Environments and Hybrid Verifiers for Scaling Open-Weights SWE Agents. COLM 2025.

合成 Issue 的训练价值

训练问题来源	轨迹数	Pass@1 (%)
真实 GitHub Issue	400	28.0
SweGen 合成 Issue	400	27.8

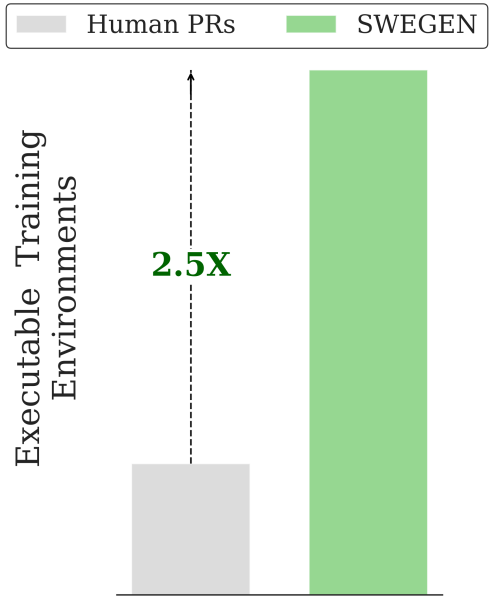


Figure 1a：作者报告环境规模约 2.5×

参考文献：[5] Jain et al. R2E-Gym: Procedural Environments and Hybrid Verifiers for Scaling Open-Weights SWE Agents. COLM 2025.

训练规模与模型容量

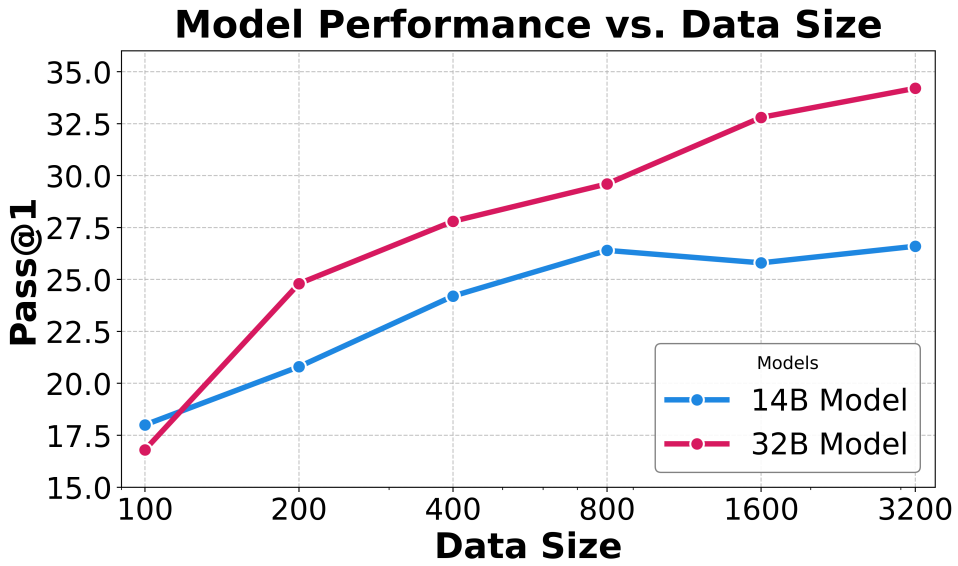


图 2：训练轨迹数与 SWE-bench Verified Pass@1，横轴为对数间隔。

消融	设置	Pass@1（%）
思考示范	保留	34.4
	移除	30.4

Figure 3：思考示范消融

参考文献：[5] Jain et al. R2E-Gym: Procedural Environments and Hybrid Verifiers for Scaling Open-Weights SWE Agents. COLM 2025.

候选里有正确解，为什么还选不出来?

SWE-bench Verified · 同一候选池 · K 为编辑候选数

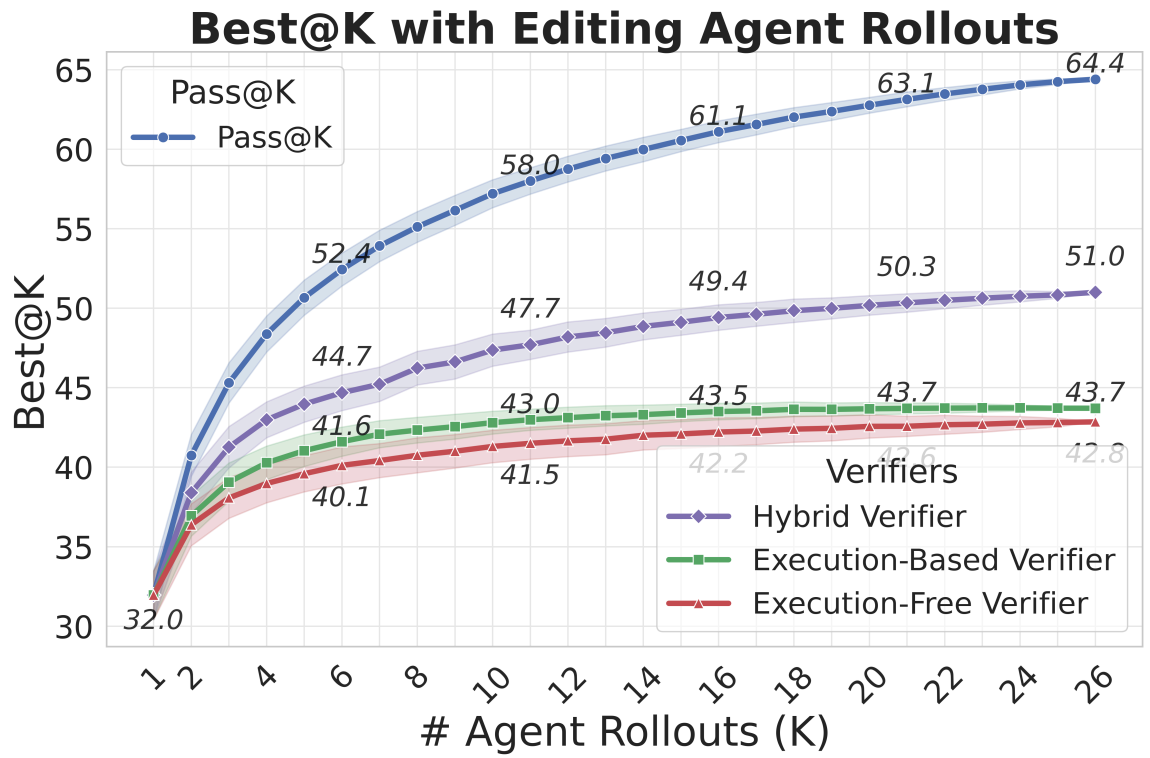


Figure 14a：候选覆盖与实际筛选

Pass@k：候选覆盖

k 个补丁中，至少一个正确

Pass@26 = 64.4%

Best@k：实际选中

验证器选一个补丁提交，统计选对的比例

单独 EF / EB 约 43%

参考文献：[5] Jain et al. R2E-Gym: Procedural Environments and Hybrid Verifiers for Scaling Open-Weights SWE Agents. COLM 2025.

两条验证路线：EF 阅读，EB 执行

EF：阅读判断

Qwen2.5-Coder-14B 验证器

1. 读取 Issue、完整轨迹和补丁
2. 预测 YES / NO 的概率
3. 得到 0–1 连续分数，对候选排序

EB：执行测试

Qwen2.5-Coder-32B 测试 Agent

1. 根据 Issue 和仓库生成复现测试
2. 结合已有回归测试，运行各个候选
3. 返回回归结果与生成测试通过数

参考文献：[5] Jain et al. R2E-Gym: Procedural Environments and Hybrid Verifiers for Scaling Open-Weights SWE Agents. COLM 2025.

为什么单独验证仍会选错?

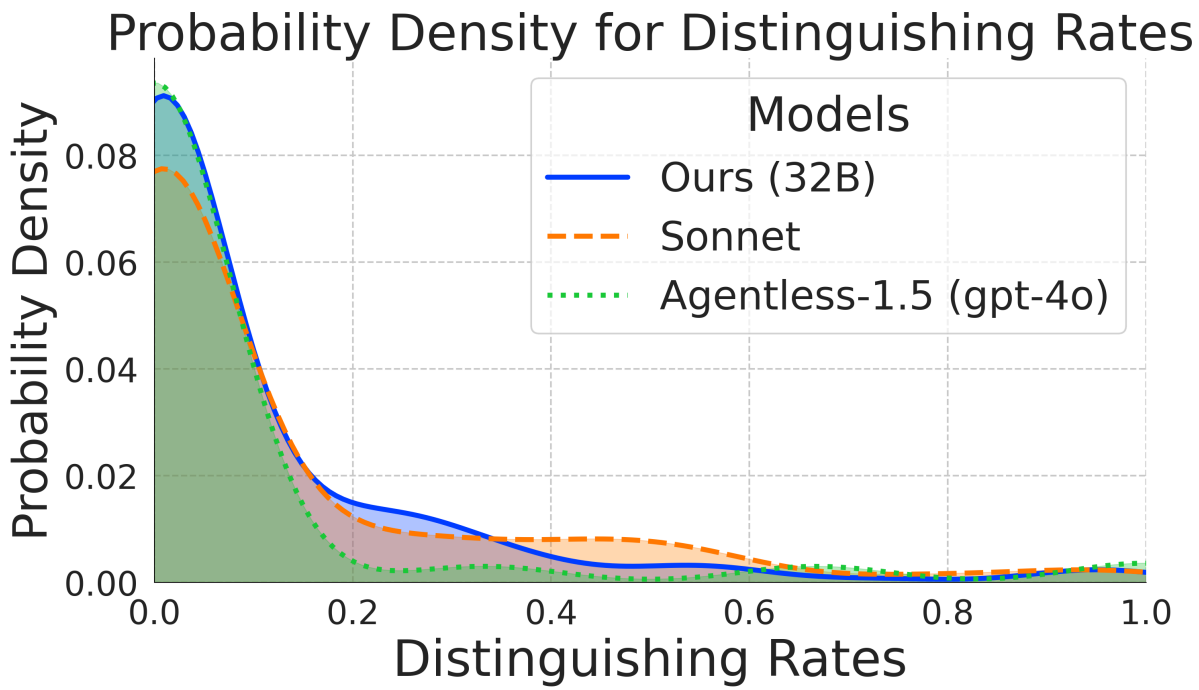


Figure 5 左：横轴为每题测试区分率

EB：测试可能让补丁同分

多数题目中，有区分信号的测试不足 20%

正确、错误补丁可能同时通过或同时失败

EF：判断可能被表述误导

> “太好了！修复有效。”

错误轨迹也可能获得高分

参考文献：[5] Jain et al. R2E-Gym: Procedural Environments and Hybrid Verifiers for Scaling Open-Weights SWE Agents. COLM 2025.

Hybrid：先筛选，再按执行结果排序

26 条编辑轨迹，另有 7 条测试生成轨迹

1. **EF 保留 Top-n 候选**
   按阅读分数缩小候选池
2. **回归测试过滤**
   保留回归得分最高的一组
3. **按生成测试通过数排序**
   通过更多测试的候选优先
4. **EF 分数打破平局**
   执行分数相同时，比较 EF 分数
5. **提交排名第一的补丁**

Hybrid Best@26：51.0%

距 Pass@26 覆盖上界仍差 13.4 个百分点

条件	Best@26 (%)
只有回归测试	47.4
采用 Agentless 生成测试	48.8
不做 EF Top-n 预筛	49.8
完整 Hybrid	**51.0**

参考文献：[5] Jain et al. R2E-Gym: Procedural Environments and Hybrid Verifiers for Scaling Open-Weights SWE Agents. COLM 2025.

SWE-RL 在什么数据上做 RL?

自采公开 GitHub PR 数据，排除 SWE-bench 使用的全部仓库

460 万仓库

GH Archive 与源码

2015.01 至 2024.08
克隆完整代码历史

→ **2,400 万 PR**

聚合已合并 PR

关联 Issue 与讨论
恢复代码与累计补丁

→ **约 1,100 万**

清洗后 PR

去机器人与空修改
过滤过大或低质变更

→ **27.3 万种子**

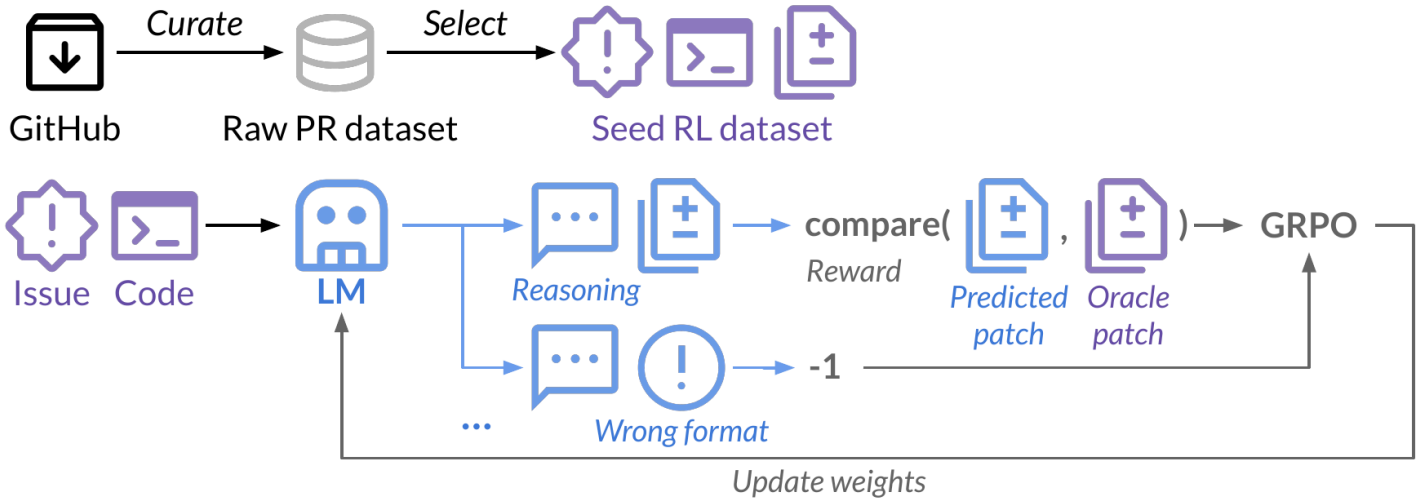
筛选 bug 修复任务

关联 Issue，修复请求
修改涉及程序文件

参考文献：[6] Wei et al. SWE-RL: Advancing LLM Reasoning via Reinforcement Learning on Open Software Evolution. NeurIPS 2025.

SWE-RL 的静态补丁训练

Llama-3.3-70B-Instruct · 27.3 万个 PR 种子 · 每题 16 个回答



Issue＋代码 → 推理与补丁 → 相似度奖励 → GRPO

- 输入 Issue 与已选文件，生成推理及 search/replace 编辑
- 格式错误得 −1，否则按补丁相似度给 0–1 分
- GRPO 更新模型，训练过程不执行工具或测试

参考文献：[6] Wei et al. SWE-RL: Advancing LLM Reasoning via Reinforcement Learning on Open Software Evolution. NeurIPS 2025.

RL 与 SFT 的修复能力

70B 模型	格式正确率	解决率（%）
Llama-3.3-Instruct	12.2%	5.4%
SWE-SFT	96.2%	29.6%
SWE-RL	**95.6%**	**34.8%**

参考文献：[6] Wei et al. SWE-RL: Advancing LLM Reasoning via Reinforcement Learning on Open Software Evolution. NeurIPS 2025.

奖励消融与正确性缺口

奖励	格式正确率	修复率
补丁完全匹配	94.2%	29.0%
补丁序列相似度	**95.6%**	**34.8%**

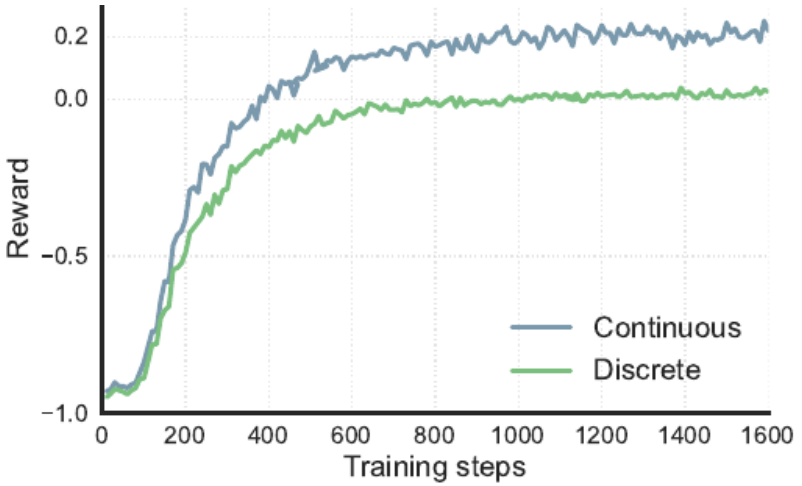


Figure 5：两种训练奖励的变化

> 参考：`if count >= limit:`
> 候选：`if count > limit:`

参考文献：[6] Wei et al. SWE-RL: Advancing LLM Reasoning via Reinforcement Learning on Open Software Evolution. NeurIPS 2025.

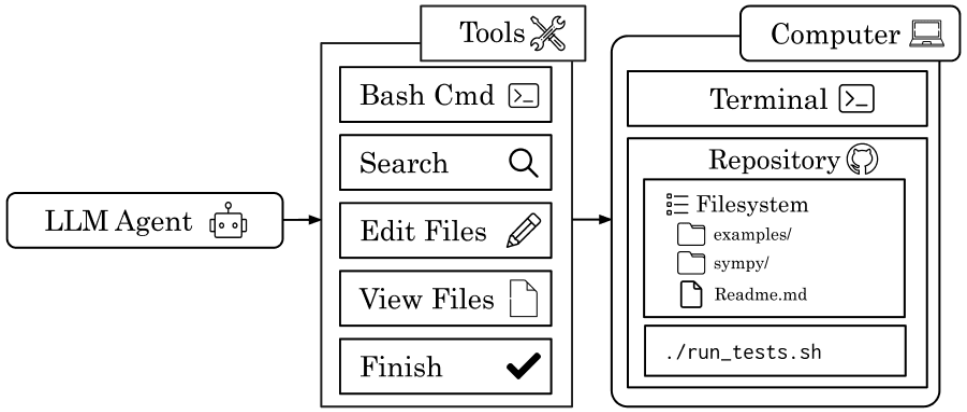
修复训练的领域外泛化

只做 Issue 修复 RL · 领域外评测 · 单位：%

基准	任务说明	基模	SFT	RL
HumanEval+	按函数签名与说明编写 Python 函数，以增强测试验证	76.2	73.2	79.9
BigCodeBench-Hard (I)	按自然语言指令编程，调用相关库（I = Instruct）	28.4	25.7	28.4
CRUXEval-I	给定代码与输出，反推一个有效输入	60.5	68.4	71.6
CRUXEval-O	给定代码与输入，推算输出	61.9	75.1	75.5
MATH（宽松）	解竞赛数学题，放宽最终答案提取格式	70.9	71.7	73.7
MMLU	57 个学科的知识与推理选择题	86.49	85.26	86.82

参考文献：[6] Wei et al. SWE-RL: Advancing LLM Reasoning via Reinforcement Learning on Open Software Evolution. NeurIPS 2025.

DeepSWE 的多轮交互 RL



- Qwen3-32B 启用思考，新增 SWE 训练只用 RL
- R2E-Gym 约 4,500 任务，64 张 H100、6 天
- 主动提交后，指定测试全部通过得 1，否则为 0
- 多轮 GRPO++，仅对模型输出计算损失

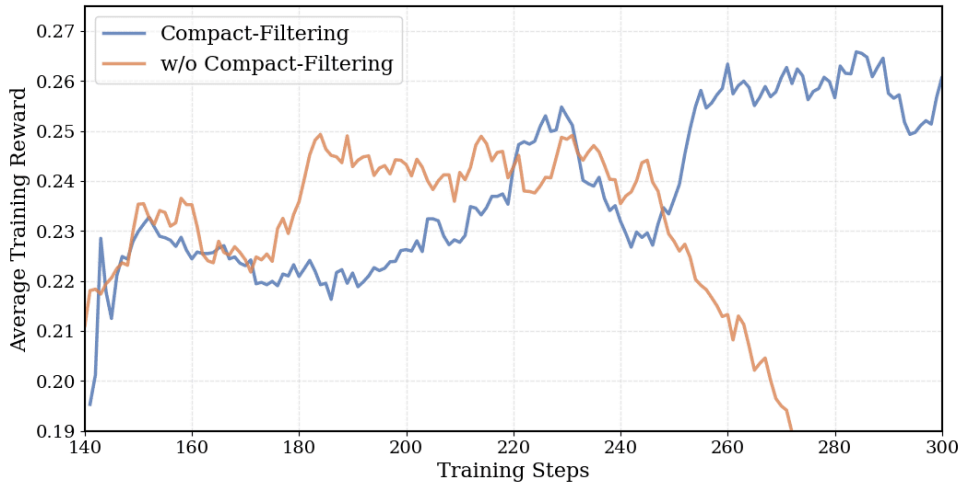
参考文献：[7] Luo et al. DeepSWE: Training a Fully Open-sourced, State-of-the-Art Coding Agent by Scaling RL. Together AI Blog, 2025.

SWE-RL 与 DeepSWE：训练方式对比

方面	SWE-RL	DeepSWE
基模	Llama-3.3-70B-Instruct	Qwen3-32B
训练任务	约 27.3 万条真实 GitHub Issue–PR 数据	约 4,500 个 R2E-Gym 可执行任务
模型输入	Issue＋已经选好的代码文件	Issue＋可操作的代码库环境
训练时的行为	一次生成推理和代码编辑	多轮搜索、查看、编辑、运行命令，最后提交
奖励	补丁相似度 0–1，格式错误 −1	主动提交后，指定 F2P、P2P 测试全通过得 1，否则得 0
RL 方法	GRPO	适配多轮交互的 GRPO++

参考文献：[6] Wei et al. SWE-RL: Advancing LLM Reasoning via Reinforcement Learning on Open Software Evolution. NeurIPS 2025.
[7] Luo et al. DeepSWE: Training a Fully Open-sourced, State-of-the-Art Coding Agent by Scaling RL. Together AI Blog, 2025.

轨迹过滤与训练稳定性

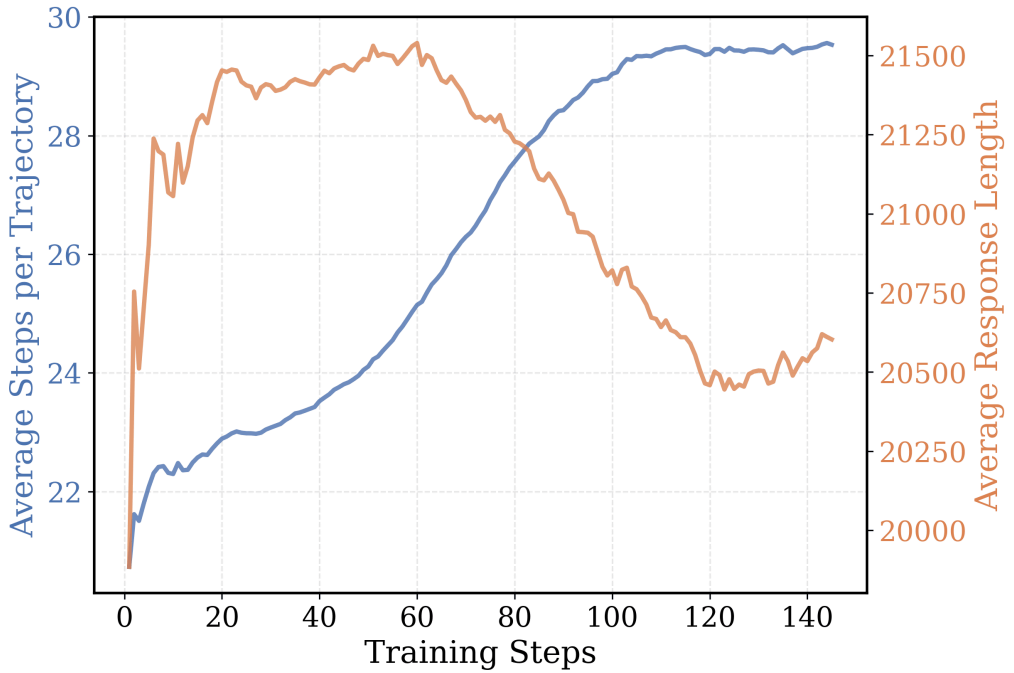


横轴训练步，纵轴平均训练奖励，蓝色过滤、橙色未过滤

- 过滤达到上下文、步数上限或超时的轨迹
- 未过滤组后期奖励下降，过滤组保持较高奖励
- 主动提交的轨迹才参与成功奖励更新
- 终止与截断规则，会改变模型学到的行为

参考文献：[7] Luo et al. DeepSWE: Training a Fully Open-sourced, State-of-the-Art Coding Agent by Scaling RL. Together AI Blog, 2025.

响应长度与环境步数



横轴训练步，左轴环境步数，右轴响应长度，分别读取两条曲线

- 蓝线：平均环境步数约 21 → 29，橙线：响应长度峰值后回落
- 作者解释：抑制单步过度思考，更多利用跨步反馈

参考文献：[7] Luo et al. DeepSWE: Training a Fully Open-sourced, State-of-the-Art Coding Agent by Scaling RL. Together AI Blog, 2025.

增加 token 与增加候选

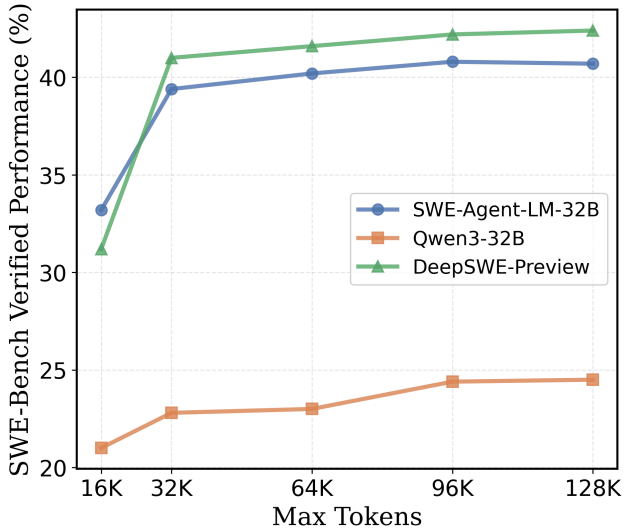


Figure 9：单条轨迹的最大 token 预算

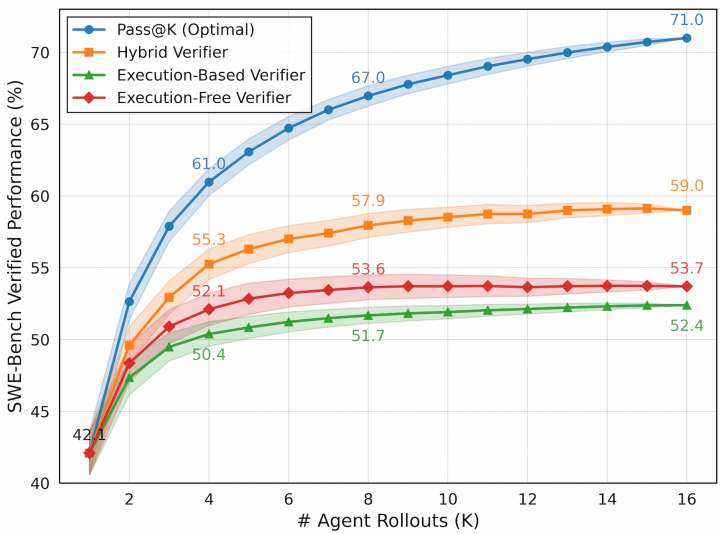


Figure 10：独立候选数与选择方法

单轨迹 Pass@1	Hybrid Best@16	Oracle Pass@16
42.2%	59.0%	71.0%

参考文献：[7] Luo et al. DeepSWE: Training a Fully Open-sourced, State-of-the-Art Coding Agent by Scaling RL. Together AI Blog, 2025.

任务、接口、训练与验证

论文	数据或训练信号	AI4Industry技术路径
SWE-Bench	Issue、仓库快照、评测测试	评测设施：任务与验收
SWE-Agent	模型 + 专用交互接口	资源算子化：工具接口
OpenHands	Agent、事件流、运行时	Agent-ready：运行与记录
SWE-Gym	真实任务环境 + 成功轨迹 SFT	工作流数据：成功轨迹
R2E-Gym	程序化环境 + 轨迹 SFT + Hybrid	工作流数据：成功轨迹
SWE-RL	静态补丁 + 相似度奖励 RL	-
DeepSWE	多轮交互 + 执行奖励 RL	-

参考文献：[1] Jimenez et al., SWE-bench, 2024. [2] Yang et al., SWE-agent, 2024. [3] Wang et al., OpenHands, 2025. [4] Pan et al., SWE-Gym, 2025. [5] Jain et al., R2E-Gym, 2025. [6] Wei et al., SWE-RL, 2025. [7] Luo et al., DeepSWE, 2025.