SWE-ABS

Adversarial Benchmark Strengthening Exposes Inflated Success Rates on Test-based Benchmarks

Boxi Yu*, Yang Cao*, Yuzhong Zhang, Liting Lin, Junjielong Xu, Zhiqing Zhong,
Qinghua Xu, Guancheng Wang, Jialun Cao, Shing-Chi Cheung, Pinjia He†, Lionel Briand

Lero @ University of Limerick | *CUHK–Shenzhen* | *HKUST* | *University of Ottawa*

* Equal contribution † Corresponding author

Code & Strengthened Tests: **github.com/OpenAgentEval/SWE-ABS**

The SWE-Bench leaderboard is misleading

19.78%

of "solved" patches
are semantically wrong

78.8% → 62.2%

top agent's resolve rate
(−16.6 pp)

#1 → #5

top agent's rank
after strengthening

Across **11,041 patches** from the top-30 SWE-Bench Verified agents, our strengthened test suites reject **2,184 (19.78%)** that the original tests accepted.

Why? PR tests verify, they don't discriminate

Two systematic weaknesses in PR-derived tests:

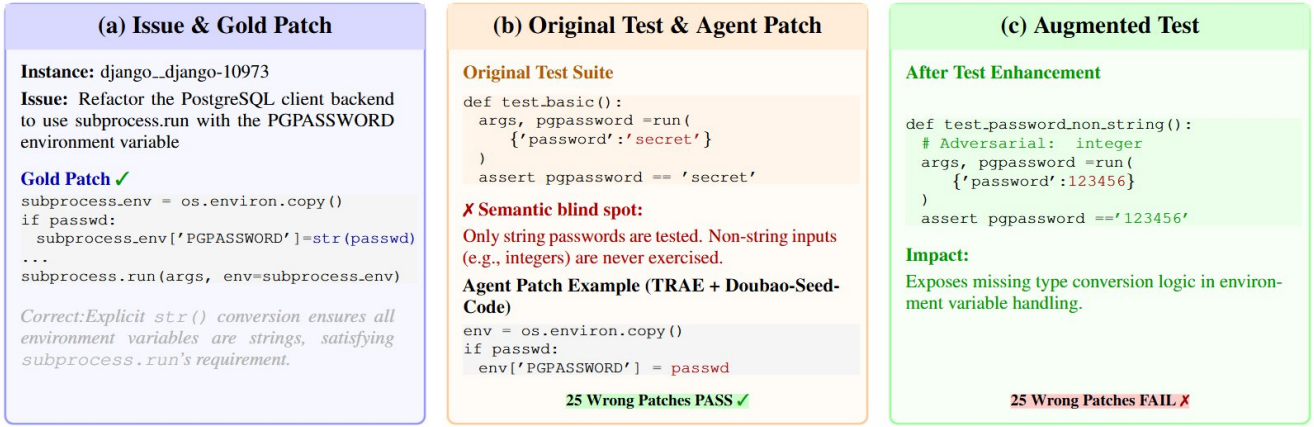
- **Coverage gaps**

tests miss patch-affected code entirely.

- **Semantic blind spots**

tests accept superficially correct behavior.

Example :A Django issue requires str(passwd) when passing PGPASSWORD to subprocess.run. The original test only checks string passwords — **25 wrong patches** omit the conversion and silently pass.



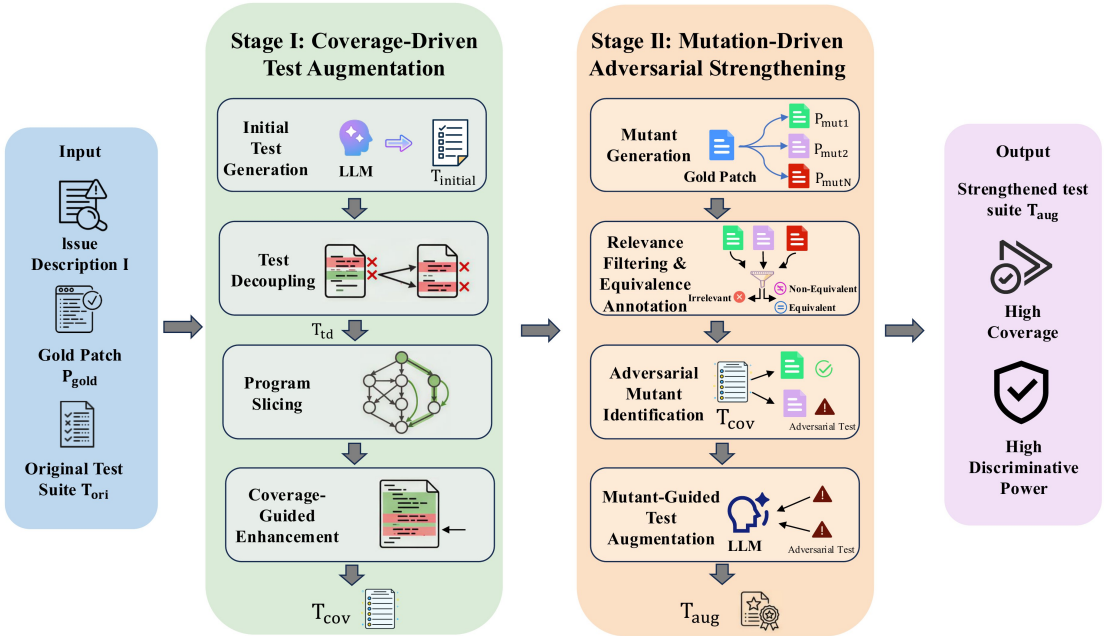
SWE-ABS: a two-stage adversarial framework

Stage I — Coverage-Driven Augmentation

Program slicing finds patch-relevant code; LLM generates tests for uncovered lines. A test-decoupling step prevents overfitting to the gold patch.

Stage II — Mutation-Driven Adversarial Strengthening

Synthesize plausible-but-incorrect mutant patches that pass current tests, then generate targeted tests to kill them. Mirrors red-team / blue-team dynamics.



Results: 25.1× stronger than prior work

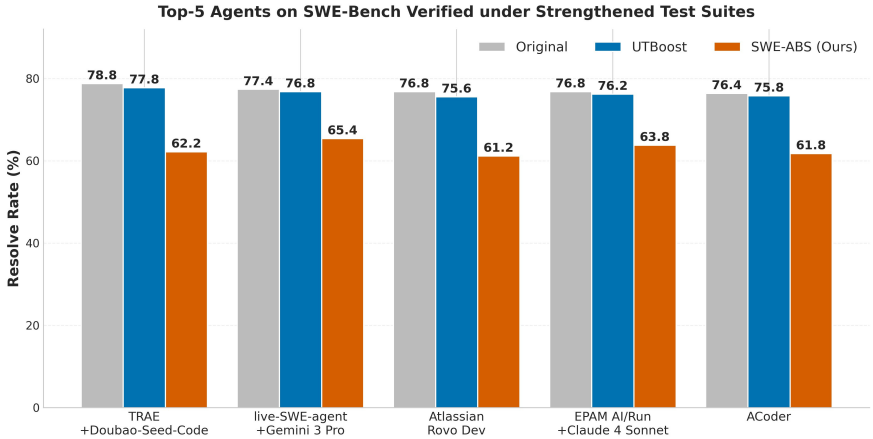
Benchmark-level (SWE-Bench Verified, 500 instances)

Augmentation	Str.	Avg. Drop	Patch Kill	# Rank Changes	Spearman ρ
UTBoost	10 / 500	0.70	105/11,041	24	0.98
SWE-ABS*	206 / 500	11.22	1683/11,041	25	0.86
SWE-ABS	251 / 500	14.56	2184/11,041	30	0.82

SWE-ABS strengthens **251 / 500 (50.2%)** instances — **25.1×** the rate of UTBoost — and rejects **2,184 / 11,041** patches.

Avg. resolve-rate drop: **14.56 pp** (vs. 0.70 pp for UTBoost).

Agent-level (Top-5)



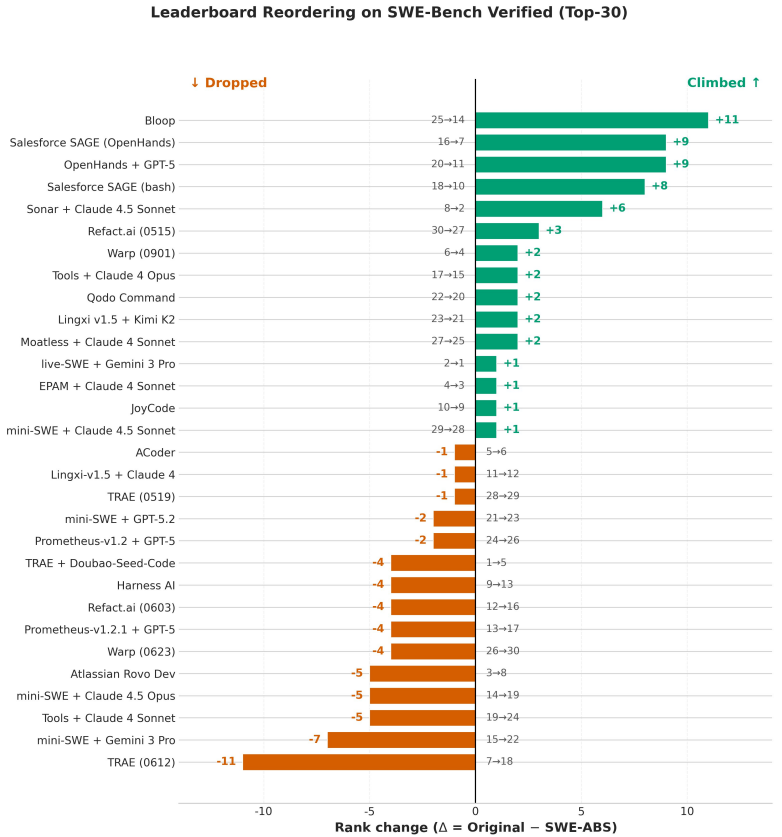
#1->#5 #2->#1 #3->#8 #4->#3 #5->#6

Strengthening reshuffles the entire Top-30

Among the top-30 agents:

- **30 / 30 agents** change rank — the leaderboard is not stable.
- **TRAE ↓ 4 ranks** (#1 → #5) despite the highest original score.
- **Bloop ↑ 11 ranks** (#25 → #14) — most robust patches.
- **Spearman ρ** drops from **0.98** (UTBoost) to **0.82**.

Current rankings may not reflect true agent capability.



Generalizes to harder, multi-language benchmarks

	Verified	Pro (subset)
Instances	500	150
Top agent score	78.80%	45.89%
Strengthened	251 (50.2%)	**97 (64.7%)**
Avg. drop	14.56 pp	**16.46 pp**
Languages	Python	Py / JS / Go / TS

Task difficulty ≠ test strength

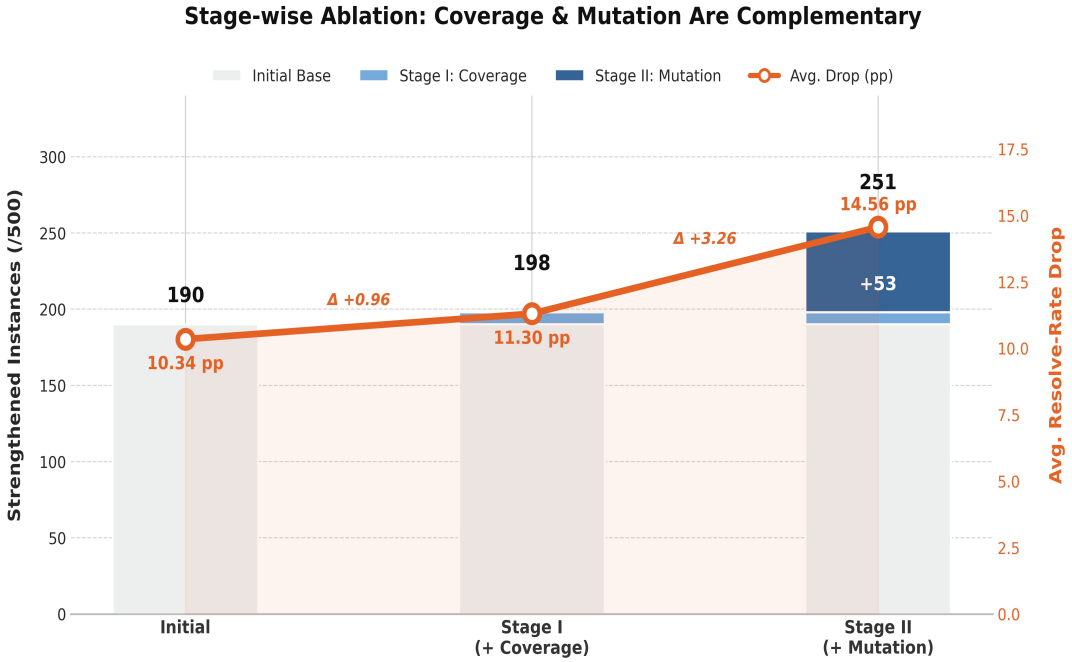
SWE-Bench Pro is **33 pp harder** for top agents, yet SWE-ABS strengthens it at a **higher** rate (64.7% vs. 50.2%).

⇒ Test inadequacy persists across difficulty levels.

⇒ Holds on contamination-resistant, multi-language code.

⇒ Method is not a Verified-specific artifact.

Ablation: coverage and mutation are complementary



Stage I alone — modest gain:
190 → 198 strengthened (+8 instances).

+ Stage II — substantial gain:
198 → **251** (+53 instances).

Coverage enforces structural adequacy; mutation exposes semantic blind spots that survive coverage.
Both stages are needed.

What kinds of bugs does SWE-ABS catch?

Analysis of **100 sampled patches** *rejected by SWE-ABS but accepted by original tests* (Cohen's κ = 0.67).

46%

Logic errors

Patches implement fundamentally wrong logic — pass on golden cases, fail elsewhere.

29%

Incomplete fixes

Primary case patched, related call sites or branches missed — "shallow" solutions.

13%

Type mismatches

String / int / None confusion at API boundaries.

12%

Boundary violations

Off-by-one, edge cases, null / empty input handling.

75% are semantic errors (logic + incomplete) — modern LLMs solved surface correctness; the gap is semantic reasoning.

Takeaways

1. **Weak tests inflate scores.** 1 in 5 "solved" SWE-Bench patches are semantically wrong — strengthen test suites before trusting a leaderboard.

2. **Task difficulty ≠ test strength.** SWE-ABS achieves a 64.7% strengthening rate on SWE-Bench Pro — comparable to Verified — even though the top agent is 33 pp weaker there.

3. **Adversarial generation beats LLM-only generation.** Mutation-driven probing adds +53 instances over coverage alone — it's what finds semantic blind spots.

Strengthened test suites & code: github.com/OpenAgentEval/SWE-ABS