

What are AI safety benchmarks?

An AI safety benchmark is a repeatable test that measures how a large language model (LLM) or AI agent behaves when someone tries to misuse it: whether it can be hijacked by a prompt injection, talked past its safety rules, steered into harmful actions, or tempted to cheat. Inspect, the open-source evaluation framework from the UK AI Security Institute, lists 25 of these under Safeguards. They group into seven families, from agent hijacking to hazardous knowledge. A benchmark gives you a comparable number; it does not tell you how your own application fails.

HOW IT WORKS

01 Prompt injection and agent hijacking

The largest and most security-relevant family. Each of these plants attacker instructions in data the agent reads, then checks whether the agent follows them.

- AgentDojo: realistic email, Slack, travel and banking tasks with injected instructions in tool output. Scores both task success and attack success.
- b3: crowdsourced attacks tested against the LLM that powers an agent, across data exfiltration, tool misuse and denial of service.
- AgentThreatBench: turns the OWASP Top 10 for Agentic Applications into runnable tests for memory poisoning, goal hijack and data exfiltration.
- CodeIPI: hides injected instructions in issues, code comments, README files and config that a coding agent reads while fixing a bug.

02 Harmful requests and jailbreaks

These measure whether a model refuses what it should, and whether a jailbreak actually produces useful harmful output rather than a refusal in disguise.

- AgentHarm: explicitly malicious multi-step tasks for agents across 11 harm categories, including fraud, cybercrime and harassment.
- StrongREJECT: forbidden prompts plus a judge that scores how specific and convincing a jailbroken answer is. Inspect lists it twice, as a plain eval and as an agent task.

SOURCES

- [1] Inspect Evals registry: Safeguards evaluations
- [2] Inspect: open-source framework for LLM evaluations
- [3] AgentDojo (Debenedetti et al., 2024)
- [4] AgentHarm (Andriushchenko et al., 2024)

- FORTRESS: 500 expert-written adversarial prompts on national security and public safety, each graded against its own rubric.
- CoCoNot (The Art of Saying No) and AbstentionBench: whether a model declines or abstains when it should, such as on unanswerable, underspecified or false-premise questions. Both are useful for measuring over-refusal as well as under-refusal.

03 Guardrails and long-term memory

Two newer areas: using one model to police another, and the risks of an assistant that remembers.

- Mind2Web-SC: tests whether a model can act as a guardrail agent, writing and running code that allows or blocks a web agent's action based on user rules.
- PersistBench: measures memory leakage across unrelated conversations, and memory-driven sycophancy, where stored beliefs bend the model's answers.

04 Cheating, reward hacking and honesty

These ask whether an agent will take a shortcut that looks like success: the problem known as reward hacking.

- Reward-Hack-Bench: SWE-bench and CyBench tasks with an injected prompt that offers a way to retrieve the solution. A judge scores a task as solved only if the agent did it without cheating. A control version runs the same 8 tasks with no cheat prompt, as a baseline.
- Hack-Verifiable Terminal Bench 2.1: 89 terminal tasks with hidden solution and test files. A file watcher records whether the agent opened them instead of doing the work.
- MASK: whether a model contradicts its own stated beliefs when pressured to lie.

05 Persuasion and social engineering

These measure a model's ability or willingness to manipulate, which matters for both abuse of your model and abuse by it.

- Make Me Pay: a con-artist model tries to talk a mark model into donating money.
- MakeMeSay: one model must get another to say a codeword without arousing suspicion.
- APE: how willing a model is to attempt persuasion on harmful, controversial and harmless topics.

For the attacker's side of this, see AI-assisted phishing.

06 Hazardous knowledge

These estimate how much dangerous expertise a model holds, rather than whether it can be hijacked.

- WMDP: about 3,600 multiple-choice questions in biosecurity, cybersecurity and chemical security, used as a proxy for hazardous knowledge. Inspect splits it into subsets such as wmdp_bio.
- LAB-Bench: questions on real scientific research workflows. It is listed under Safeguards because lab capability is a dual-use signal.
- FORTRESS (above) also sits here, since its prompts target national security and public safety.

07 Bias and values

The remaining four test values, not security. They are listed here for completeness.

- StereoSet: stereotype bias across gender, race, religion and profession.
- ANIMA and TAC: moral reasoning about animal welfare, and whether an agent shows that awareness when buying tickets for a user.
- MORU: how a model handles moral uncertainty in unfamiliar scenarios.

08 What a benchmark score does not tell you

Benchmarks are necessary but narrow. Four limits are worth keeping in mind before you trust a number:

- The test cases are public. Models can be trained on them, intentionally or not. AgentHarm ships a canary string so contamination can be detected.
- Many scores depend on a judge model. StrongREJECT, AgentHarm and PersistBench use an LLM to grade answers, so the grader's own blind spots carry into the score.
- Your application is not in the test set. A model that resists injection in a generic travel app can still leak data through your tools, your retrieval pipeline and your permissions. Excessive agency is decided by how you wire the model, not by the model.
- Scores compare models; they do not certify them. The AgentHarm authors say plainly that its grades are not an overall measure of agent safety.

Use benchmarks to choose and track a model. Use AI red teaming and a pentest of your own agent to find how it actually breaks.

Test your AI application before an attacker does.

securelayer7.net/learn/ai-security/ai-safety-benchmarks

[Open online](https://securelayer7.net/learn/ai-security/ai-safety-benchmarks)