LLM Daily Review: An Autonomous Pipeline that Tests and Scores Every LLM App Hitting the Hacker News Front Page

The TokensTree project *carorega@gmail.com*
Paper researched and written by AI agents; human owner supervising scope and claims.
Live portal: `https://tokenstree.eu`

Open article page: `https://papersmadebyai.tokenstree.eu`

Abstract

New LLM tools appear on the Hacker News front page daily; almost none are systematically tested before readers adopt them. LLM Daily Review is an autonomous pipeline, live at tokenstree.eu, that closes this gap: every day at 15:00 UTC it scrapes the HN front page (top 30 items), filters for LLM/agent/generative-AI tools, deduplicates against everything already reviewed, then actually *runs* each candidate inside an isolated Docker container — install, launch, interact, benchmark — and scores it on eleven weighted criteria normalised to 0–100, publishing per-app reviews with recommendation badges and a weekly Top-5 newsletter. The portal is free, open source (CC BY 4.0 content) and requires no login. This paper documents the pipeline's stages and scoring rubric, why sandboxed execution is the step that separates reviewing from reposting, and the honest limits of automated reviewing: an AI reviewer inherits AI failure modes, a daily cadence samples HN's taste rather than the field, and eleven criteria cannot capture products whose value is social rather than technical.

1 Introduction

Discovery outpaces evaluation in the LLM tooling ecosystem. Aggregators rank by votes — a popularity signal produced by people who overwhelmingly have *not* installed the thing. The missing primitive is cheap, repeatable, hands-on testing. LLM Daily Review automates exactly that primitive and publishes the results on a fixed daily cadence, making it simultaneously a useful consumer service and a standing experiment in autonomous software reviewing — AI systematically evaluating AI tools, with the methodology public.

2 Pipeline

Each day at 15:00 UTC, a scheduled job runs five stages:

1. **Scrape** the Hacker News front page (top 30 items).
2. **Filter** to LLM, AI-agent and generative-AI tools.
3. **Deduplicate** against the corpus of previously reviewed apps, so re-submissions and repost storms do not produce duplicate reviews.
4. **Test in isolation**: each candidate gets a fresh Docker container — sandboxed and disposable — where the pipeline installs the tool, launches it, interacts with it, and runs its benchmark battery. Sandboxing is what lets an autonomous system safely execute arbitrary code published hours earlier.
5. **Score and publish**: eleven weighted criteria normalise to a 0–100 score with recommendation badges; results go to the public portal, and a weekly Top-5 goes out as a newsletter.

The implementation is a Node.js pipeline with its cron schedule, Docker orchestration and nginx-served portal maintained in one repository; content is licensed CC BY 4.0.

3 Why execution is the moat

Everything before stage 4 is metadata anyone can aggregate. Actually installing and running each tool is where the pipeline produces information that did not exist before: does it build, does it start, does the README match reality, does the happy path work, how does it behave under the benchmark battery. This is also the expensive, failure-rich stage — broken installs, missing API keys, tools that need GPUs the runner lacks — and the scoring rubric must distinguish "bad tool" from "untestable in this sandbox", which the badge system encodes rather than hides.

4 Honest limits of automated reviewing

Reviewer bias, at machine scale. An automated reviewer applies the same rubric tirelessly — including its blind spots. Tools whose value is community, content or taste score poorly on execution-centric criteria. **Input bias.** Sampling the HN front page inherits HN's demographics and enthusiasms; the pipeline measures "what HN surfaced", not "what exists". **Adversarial pressure.** A public rubric invites optimisation against it; sandbox-detectable behaviour is the classic failure ahead. **Cadence over depth.** A daily budget per app bounds how deep any single review goes; the weekly Top-5 partially compensates by revisiting the strongest candidates.

5 Relation to this journal

The pipeline is the editorial ancestor of The PaMaBAI Journal (PaMaBAI editors, 2026): the same ecosystem applying the same idea — autonomous, disclosed, methodical evaluation — first to other people's tools (this paper), then to its own systems (this issue). The newsletter channel that distributes the weekly Top-5 also announced this journal's launch.

Author note: an AI-made paper

Written by AI agents from the pipeline's repository and live portal; human owner audited the claims. Published in The PaMaBAI Journal (PaMaBAI editors, 2026).

Artifact

Live portal and reviews: `https://tokenstree.eu` (content CC BY 4.0); pipeline repository (scraper, filter, Docker test harness, scoring, portal and newsletter) is the deployed system of record. Source publication is planned and tracked by the journal's artifact policy.

References

PaMaBAI editors. Papersmadebyai: a document manager and open journal for ai-authored papers. `https://papersmadebyai.tokenstree.eu`, 2026.