NVIDIA Nemotron 3.5 Lightning Does the Grunt Work of AI Agents, 74% Cheaper When Routed

Kabui, Charles

2026-08-24

Read at ToKnow.ai

NVIDIA released Nemotron 3.5 Lightning on August 11, a 30-billion-parameter model built for the high-volume execution layer inside always-on AI agents: making tool calls, checking tool outputs, handing subtasks to helper agents, the routine steps where a frontier model is

overkill. It is a mixture-of-experts design, so only 3 billion parameters run per token, which keeps it fast and cheap. NVIDIA reports 86% accuracy on PinchBench, an agent-efficiency benchmark, and says it finishes 10,000 tasks 30% faster than Qwen3.6 35B at similar accuracy, with up to 4x the output speed of comparable models. It runs from a desktop DGX Spark to a data center, and the weights are open under the OpenMDW-1.1 license.

The companion piece is NeMo Switchyard, an open-source router that sends each step of an agent's work to the model best suited to it, usually a cheap one. LangChain benchmarked it over 145 multi-turn agent tasks, routing between Nemotron 3.5 Lightning and Claude Opus 4.8. The small model handled 93% of all calls for 10.4% of the spend, and the total bill fell 74% against running Opus alone, from $11.45 to $3.00 per run, for a roughly six-point accuracy drop. For an always-on support agent, that turns a frontier-model bill into pocket change.

Most of what an agent does is mechanical, and routing exists so that work never touches a frontier model. Agent bills are now set by routing and small models, not the flagship in the news.

Read More: NVIDIA Nemotron 3 Nano Omni: One Open Model for Vision, Audio, and Text at 9x the Throughput

Sources:

- NVIDIA Nemotron 3.5 Lightning Delivers Fast, Accurate Specialized Task Execution for Long-Running Agents (NVIDIA Technical Blog)
- Route AI Agent Workloads Across Models with NVIDIA NeMo Switchyard (NVIDIA Technical Blog)
- How many of your agent's calls actually need a frontier model? (LangChain)
- NVIDIA Nemotron 3.5 Lightning model card (Hugging Face)
- NeMo Switchyard (GitHub)

Disclaimer: *For information only. Accuracy or completeness not guaranteed. Illegal use prohibited. Not professional advice or solicitation.* ***Read more:*** */terms-of-service*