



UTD Hack 2025

Nemotron Prize Track

Ethan Einhorn

Developer Community

eeinhorn@nvidia.com



Challenge Statement:

In Addition to Using Nemotron, Your Project Must Include:

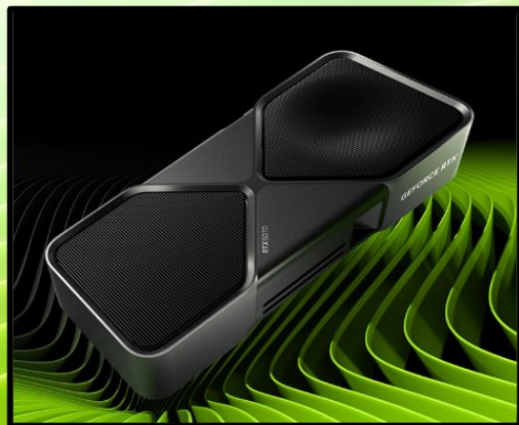
- ✓ Beyond a Chatbot - not just responding directly to prompts
- ✓ Multi-step workflows - agents that plan and execute complex tasks
- ✓ Tool integration - using external APIs/services intelligently
- ✓ Real-world applicability - solving actual problems

Prizing:

First Place:

3 RTX 5080 GPUs

3 NVIDIA Logo Hats



Second Place:

2 RTX 5080 GPUs

3 NVIDIA Backpacks

3 \$100 Brev Credits

Third Place:

3 NVIDIA Backpacks

3 \$200 Brev Credits

Essential Resources

Quick Start Workshops (self-paced, deploy in minutes):

[Report Generator Agent Tutorial](#) - Learn agent architecture with LangGraph

[Agentic RAG Tutorial](#) - Build smart retrieval systems

Refer to the get-started tutorials:

[Build a Report Generator AI Agent with NVIDIA Nemotron on OpenRouter](#)

[Build a Retrieval-Augmented Generation \(RAG\) Agent with NVIDIA Nemotron](#)

Access Nemotron:

build.nvidia.com - get API key

Via OpenRouter API Nemotron-nano-9b-v2

On [Hugging Face](#) (open weights)

Brev Getting Started:

<https://docs.nvidia.com/brev/latest/>

Getting Started - build.nvidia.com

The screenshot displays the NVIDIA AI platform interface. At the top, a navigation bar includes links for Explore, Models, Blueprints, Internal AI, GPUs, and Docs. A search bar is also present. The main content area shows a modal titled "Copy Code to Make an API Request" for the model "nvidia-nemotron-nano-9b-v2". The modal has tabs for Python, Node, and Shell, with Python selected. It includes an "Upgrade" button and a "Generate API Key" button. A code block shows the Python code for making an API request. A "View Code" button is visible in the background. Three green callout boxes provide step-by-step instructions: Step 1: Log in (pointing to the user icon), Step 2: Navigate to model and click view code (pointing to the "View Code" button), and Step 3: Generate API key (pointing to the "Generate API Key" button). A fourth callout box, "API key populates here", points to the placeholder text in the code block.

Step 1: Log in

Step 2: Navigate to model and click view code

Step 3: Generate API key

API key populates here

Copy Code to Make an API Request

Python Node Shell API Reference

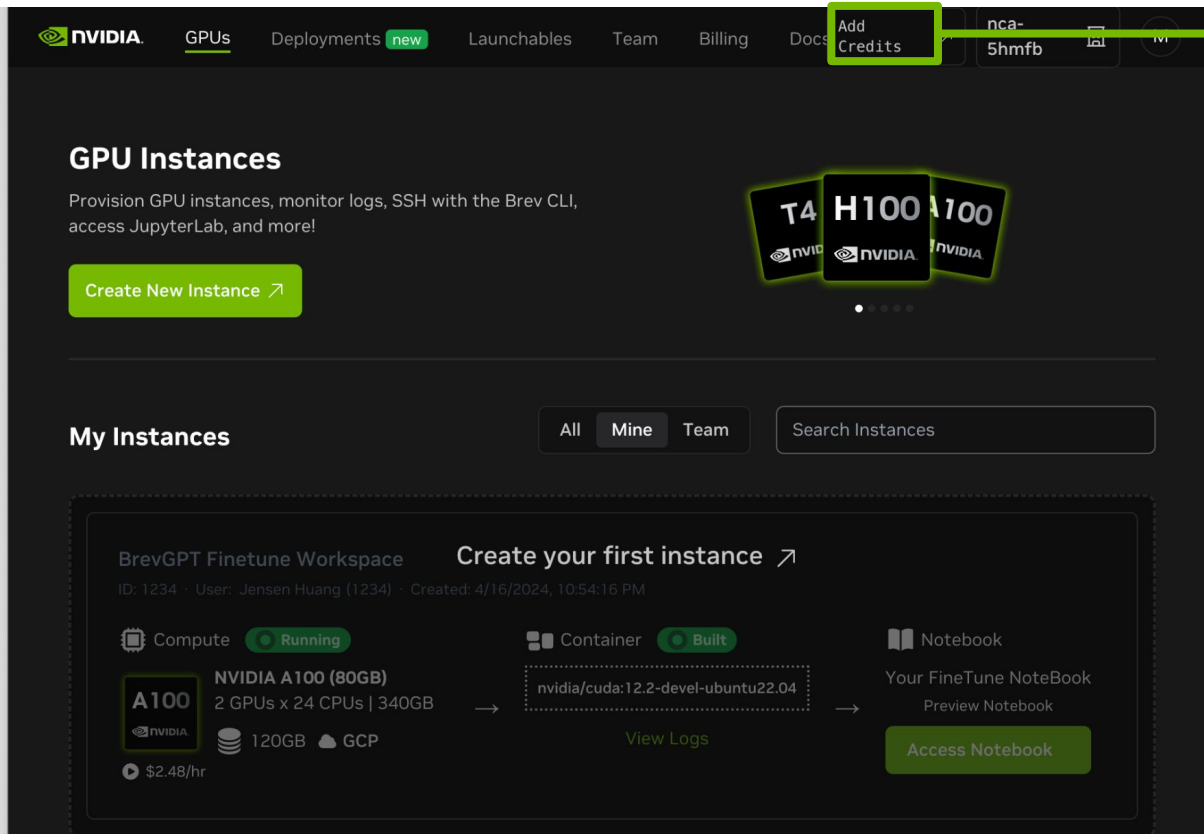
Using free API for development Upgrade Generate API Key Copy

```
from openai import OpenAI

client = OpenAI(
    base_url = "https://integrate.api.nvidia.com/v1",
    api_key = "$API_KEY_REQUIRED_IF_EXECUTING_OUTSIDE_NGC"
)

completion = client.chat.completions.create(
    model="nvidia/nvidia-nemotron-nano-9b-v2",
    messages=[{"role":"system","content":"/think"}],
    temperature=0.6,
    top_p=0.95,
    max_tokens=2048,
    frequency_penalty=0,
    presence_penalty=0,
    stream=True,
    extra_body={
```

Getting Started - brev.nvidia.com



The screenshot shows the NVIDIA Brev web interface. The top navigation bar includes links for GPU, Deployments, Launchables, Team, Billing, Docs, and Add Credits. The 'Add Credits' link is highlighted with a green box. A green arrow points from this box to the redeem code on the right. The main content area is titled 'GPU Instances' and features a 'Create New Instance' button. Below this, the 'My Instances' section shows a table of instances. The first instance is 'BrevGPT Finetune Workspace', which is in a 'Running' state. It is configured with 2 NVIDIA A100 GPUs, 24 CPUs, and 340GB of memory. The instance is running on a container with the image 'nvidia/cuda:12.2-devel-ubuntu22.04'. The cost is \$2.48/hr. A 'Create your first instance' link is also visible.

GPU Instances

Provision GPU instances, monitor logs, SSH with the Brev CLI, access JupyterLab, and more!

[Create New Instance](#)

My Instances

All Mine Team Search Instances

BrevGPT Finetune Workspace [Create your first instance](#)

ID: 1234 · User: Jensen Huang (1234) · Created: 4/16/2024, 10:54:16 PM

Compute **Running** Container **Built** Notebook

A100 **NVIDIA A100 (80GB)**
2 GPUs x 24 CPUs | 340GB
120GB 120GB GCP
\$2.48/hr

`nvidia/cuda:12.2-devel-ubuntu22.04`
[View Logs](#)

Your FineTune NoteBook
Preview Notebook
[Access Notebook](#)

\$60 Redeem code
utdhackathon-2025

Ideal Projects for This Track

Winning projects will showcase true agentic behavior:

Multi-Agent Systems: Build teams of specialized AI agents
(like Report Generator: Research Agent → Outline Agent → Writer Agent → Editor)

Agentic RAG: Systems that intelligently decide WHEN to retrieve information, not just HOW (perfect for domain-specific assistants)

ReAct Pattern Workflows: Agents that Reason → Act → Observe in loops to solve problems iteratively (like automated debugging or technical support)

Tool-Calling Applications: Leverage Nemotron's exceptional ability to use external APIs and tools (finance analysis, DevOps automation, content creation)

Multi-Modal Agents: Combine Nemotron reasoning with VLMs
(visual analysis + logical decision-making)

Agent Simulation & Evaluation: Use Nemotron to generate realistic test scenarios and evaluation pipelines

NVIDIA - NEMOTRON

“

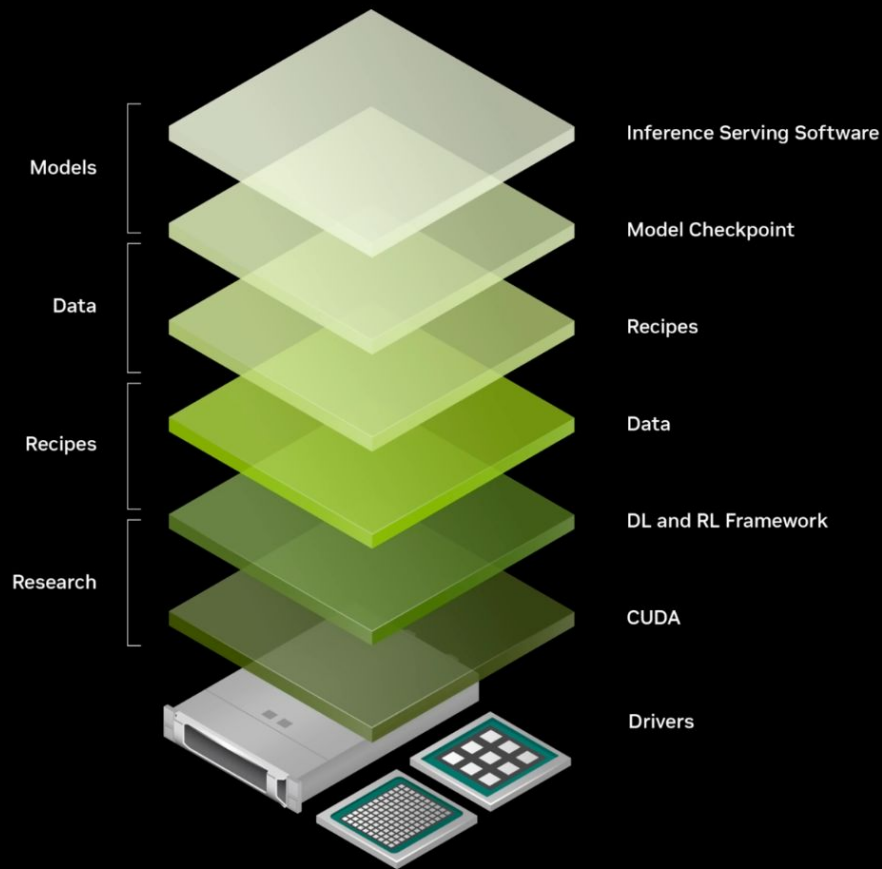
The most open approach to AI
development the world has
ever seen.

Bryan Catanzaro, VP of Applied Deep Learning Research. NVIDIA

”

Nemotron

Is more than models. It's a focus on optimizing for our AI ecosystem to provide the best quality of results across our technology stack.



What is Nemotron Best For?

NVIDIA Nemotron models are purpose-built for agentic AI - the next generation of intelligent systems that don't just answer questions, but reason, plan, and take action. Unlike general-purpose LLMs, Nemotron excels at:

- Advanced reasoning and multi-step problem-solving
- Function calling to interact with external tools and APIs
- Autonomous decision-making within agent workflows
- Retrieval-augmented generation (RAG) for knowledge-based tasks
- Multi-agent orchestration where specialized agents work together together

Models

Deploy and scale models on your GPU infrastructure of choice with NVIDIA NIM inference microservices

Filter by text

Sort By: Most Recent

Publisher

Use Case

NIM Type

Text: Nemotron

Clear Filters



nvidia

nvidia-nemotron-nano-9b-v2

High-efficiency LLM with hybrid Transformer-Mamba design, excelling in reasoning...

thinking budget chat reasoning +1



nvidia

llama-3.3-nemotron-super-49b-v1.5

High efficiency model with leading accuracy for reasoning, tool calling, chat, and instruction...

chat math advanced reasoning +3

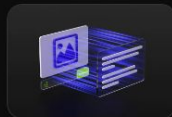


mistralai

mistral-nemotron

Built for agentic workflows, this model excels in coding, instruction following, and function calling

language generation chat +3



nvidia

llama-3.1-nemotron-nano-vl-8b-v1

Multi-modal vision-language model that understands text/img and creates informative...

doc intelligence +3



nvidia

llama-3.1-nemotron-nano-4b-v1.1

State-of-the-art open model for reasoning, code, math, and tool calling - suitable for edge agents

edge tool calling chat +3



nvidia

llama-3.1-nemotron-ultra-253b-v1

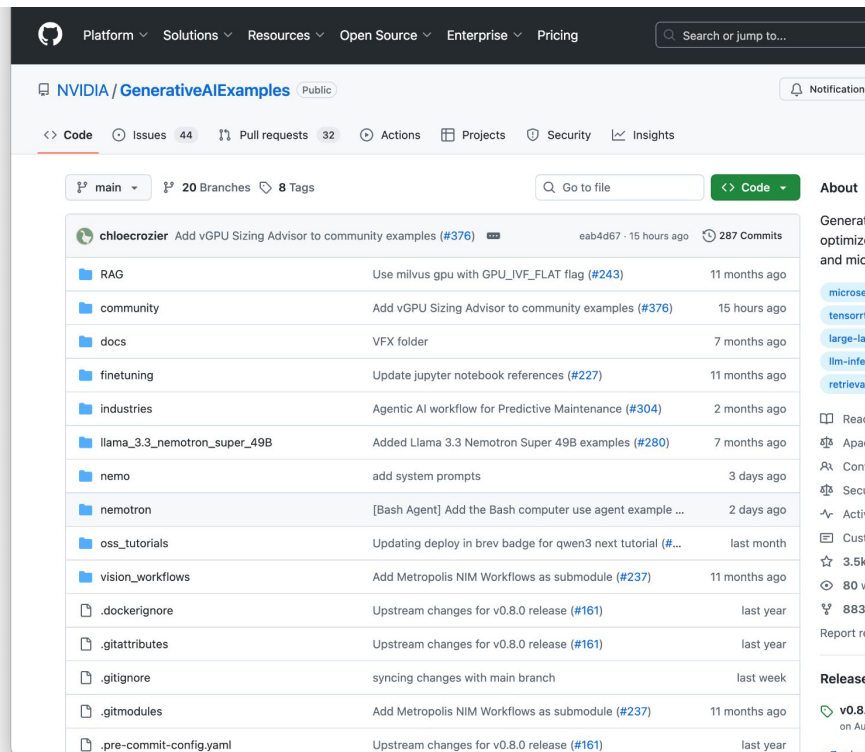
Superior inference efficiency with highest accuracy for scientific and complex math reasoning, coding...

chat math advanced reasoning +3

Learn from Past Winners

Agent examples [here](https://github.com/NVIDIA/GenerativeAIEExamples)
github.com/NVIDIA/GenerativeAIEExamples

- Route Optimization Agent (NVIDIA NeMo Hackathon): Multi-agent system coordinating logistics with specialized agents for constraint extraction and route planning
- Automated Bug Fixer (Dust Hackathon): Non-technical users describe bugs, agent autonomously diagnoses and fixes code
- Multi-Agent Research System (lablab.ai): Orchestrated deep research with autonomous agent collaboration





“

Happy Hacking!

”