LLM開発を支えるエヌビディアの生成AIエコシステム

Mana Murakami, Solution Architecture and Engineering, NVIDIA | Sep 11st 2025

AGENDA

- エージェンティックAIとエヌビディアのエコシステム
- AIエージェントの為のエヌビディアモデル
- エヌビディアの生成AI向けソリューション
- まとめ

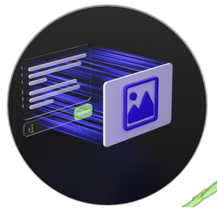
AI の進化

エージェント AI が、より強力な AI アプリケーションを可能に

PHYSICAL AI

AGENTIC AI

コーディング アシスタント
カスタマー サービス
患者ケア



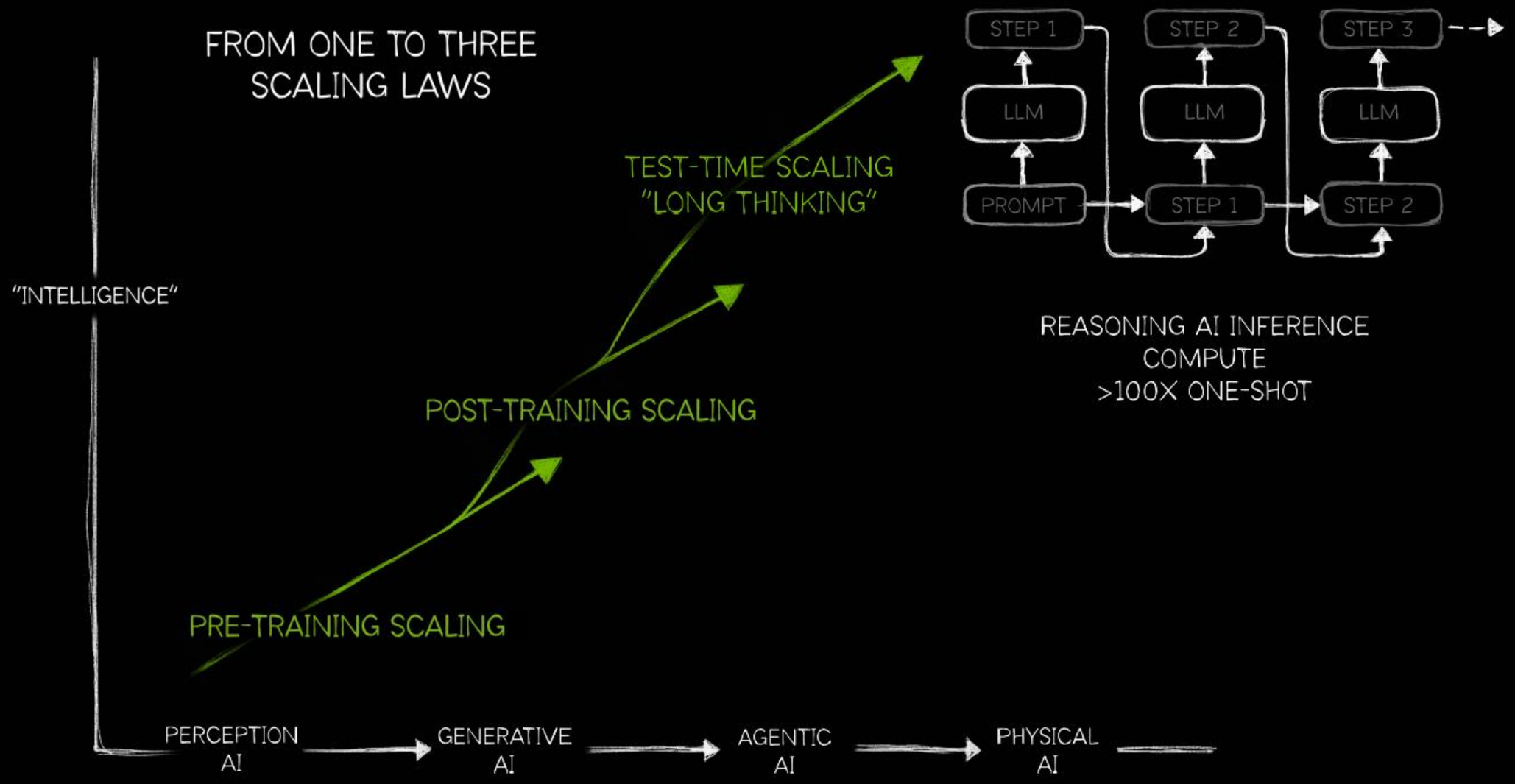
GENERATIVE AI

デジタル マーケティング
コンテンツ作成

PERCEPTION AI

音声認識
ディープ レコメンドシステム
医用画像処理

2012 ALEXNET


FROM ONE TO THREE
SCALING LAWS
"INTELLIGENCE"
PRE-TRAINING SCALING
POST-TRAINING SCALING
TEST-TIME SCALING
"LONG THINKING"
PERCEPTION AI
GENERATIVE AI
AGENTIC AI
PHYSICAL AI
STEP 1
STEP 2
STEP 3
LLM
LLM
LLM
PROMPT
STEP 1
STEP 2
REASONING AI INFERENCE
COMPUTE
>100X ONE-SHOT

NVIDIA AI: フルスタックアプローチ

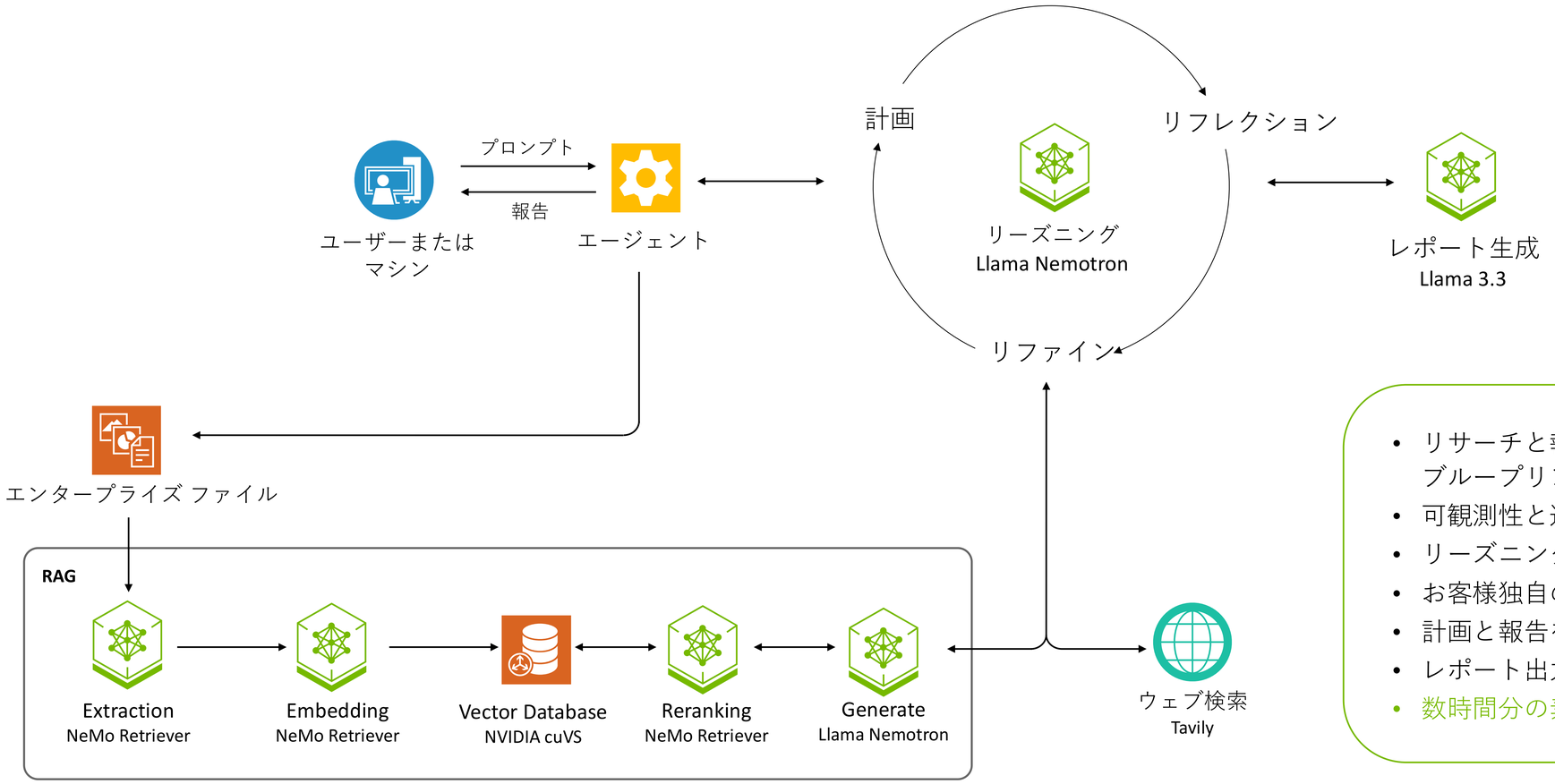
NVIDIA AI Enterprise - エンドツーエンド生成 AI パイプラインの開発とデプロイを効率化

NVIDIA AI: フルスタックアプローチ

NVIDIA AI Enterprise - エンドツーエンド生成 AI パイプラインの開発とデプロイを効率化

NVIDIA AI-Q AI Research Assistant Blueprint

エージェンティックAI の例: リーズニングモデルを使用してAIエージェント、データ、ツールを接続する



- リサーチと報告のためのオープンソースのブループリント
- 可観測性と透明性を提供
- リーズニングによりエージェントの精度を向上
- お客様独自のデータソースをアップロード
- 計画と報告をリアルタイムでガイド
- レポート出力を動的に変更
- 数時間分の素材を数分で合成

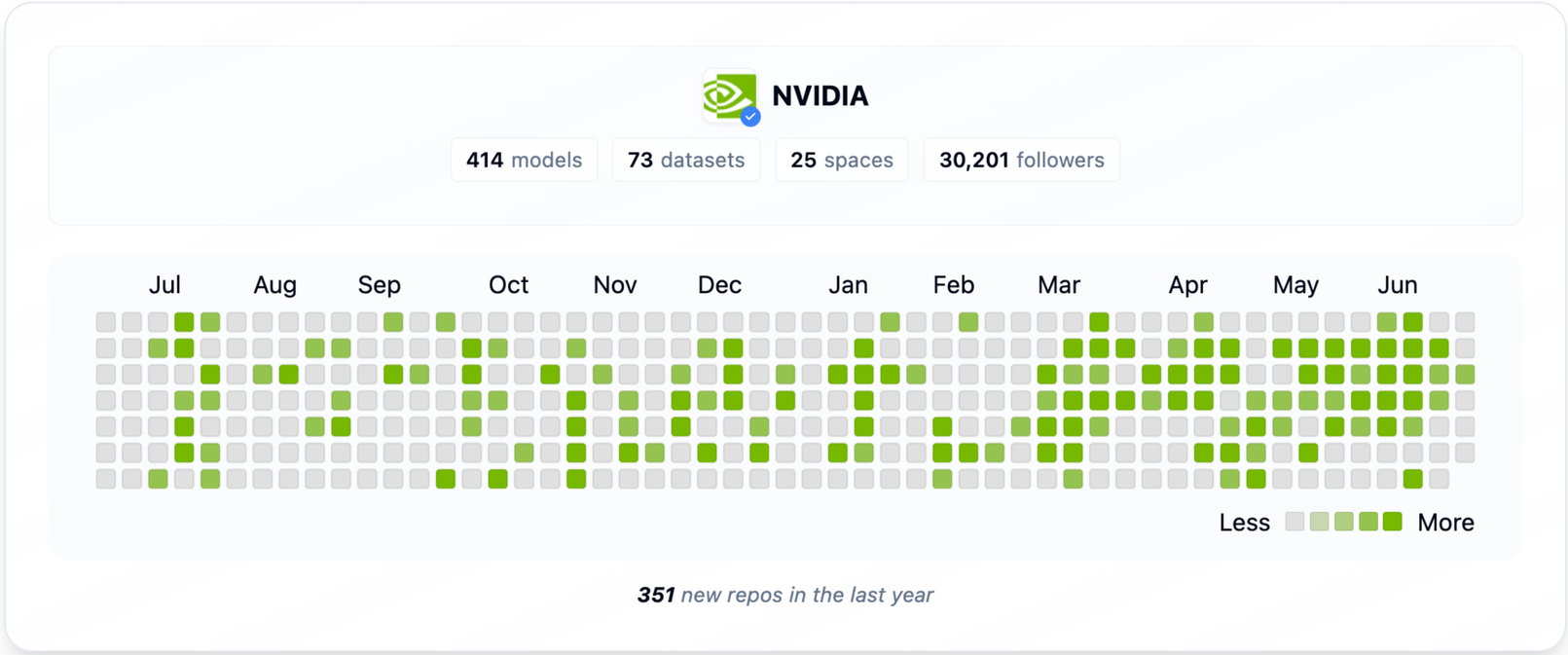
https://github.com/NVIDIA-AI-Blueprints/aiq-research-assistant

NVIDIA AI: フルスタックアプローチ

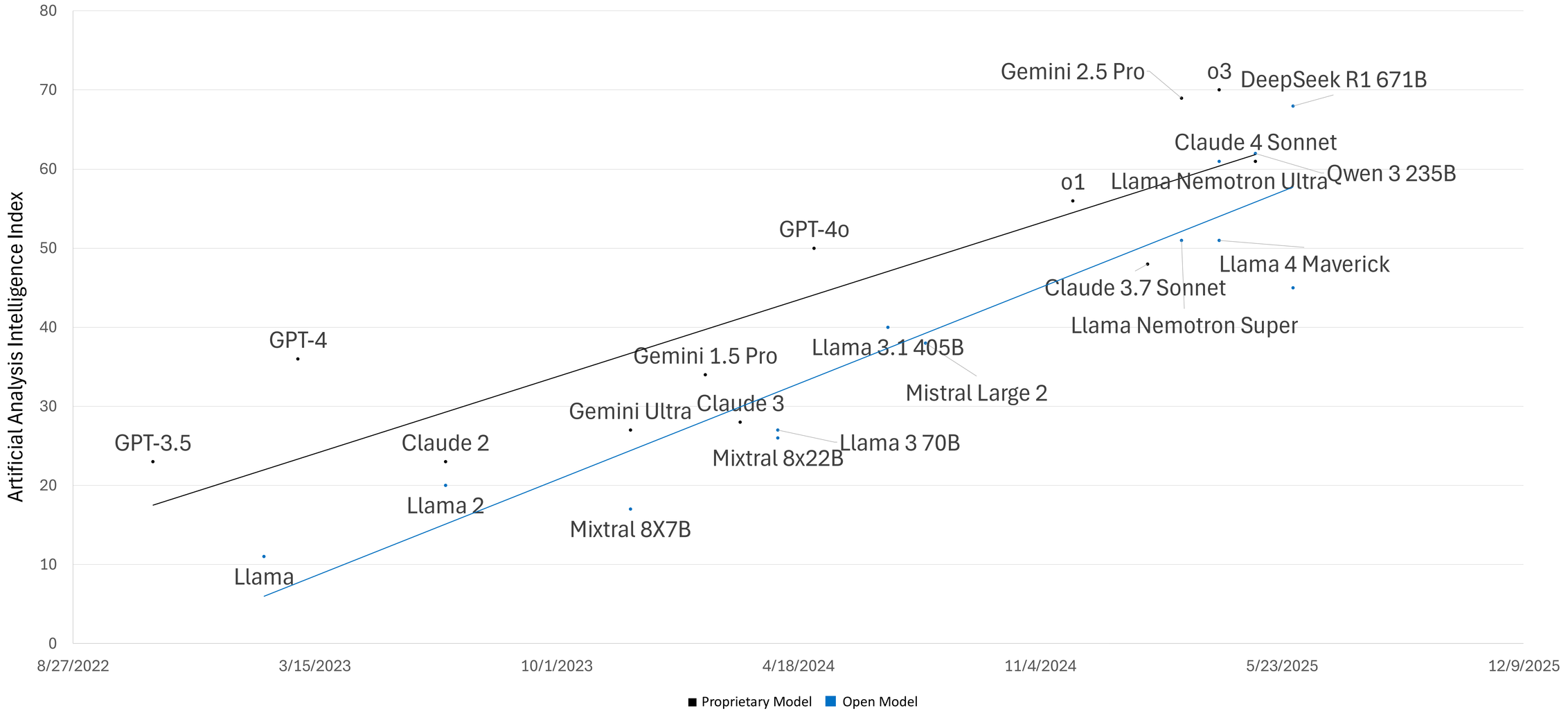
NVIDIA AI Enterprise - エンドツーエンド生成 AI パイプラインの開発とデプロイを効率化

オープンモデルとデータセットでAIコミュニティをサポート

#1 Ai2 #2 NVIDIA #3 Meta Llama #4 Hugging Face #5 Qwen #6 Google #7 Microsoft #8 IBM Granite #9 ByteDance #10 DeepSeek #11 Apple #12 Stability AI #13 Cohere Labs #14 Mistral AI_ #15 OpenAI #16 Anthropic



オープンソースのAIモデルの急速な進歩がAIエージェントの進化を加速

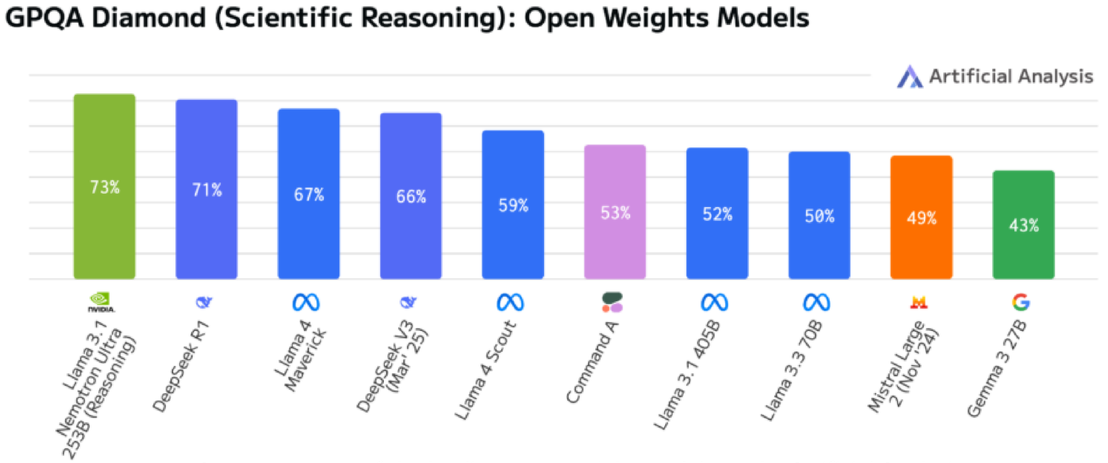


Intelligence Index: MMLU-Pro, GPQA Diamond, Humanity's Last Exam, LiveCodeBench, SciCode, AIME, MATH-500.
Source: Artificial Analysis

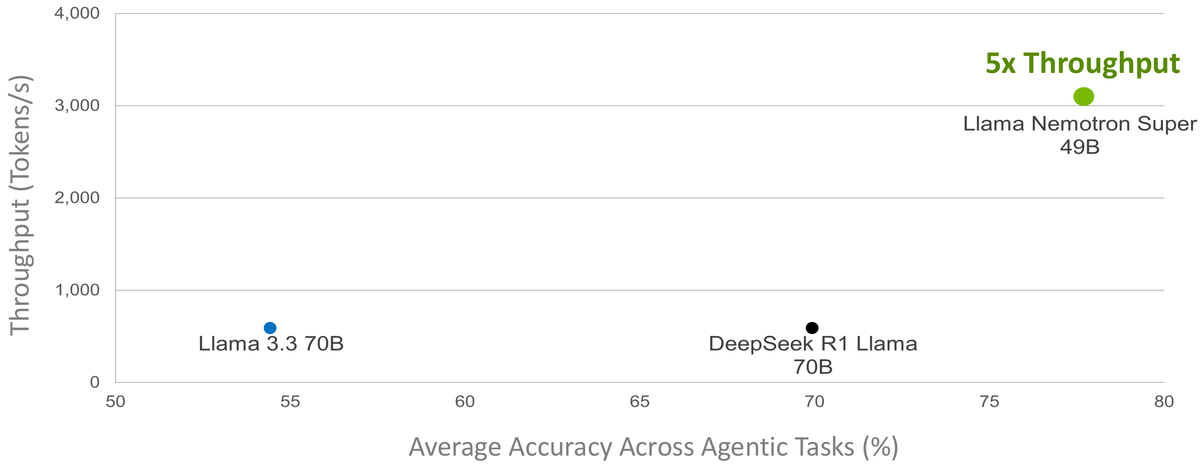
NVIDIA Llama Nemotron

Llama-Nemotron: Efficient Reasoning Models, Akhiad Bercovich et al. 2025

- **Llama Nemotron Ultra:**
  - 253B パラメータ
  - Llama 3.1 405B から蒸留
  - 最高精度のエージェント向けモデル
- **Llama Nemotron Super:**
  - 49B パラメータ
  - Llama 3.3 70B から蒸留
  - 1枚のデータセンター GPU で推論可能かつ最高のスループットと最高精度
- **Llama Nemotron Nano:**
  - 8B パラメータ
  - Llama 3.1 8B ベース
  - PC およびエッジで最高精度

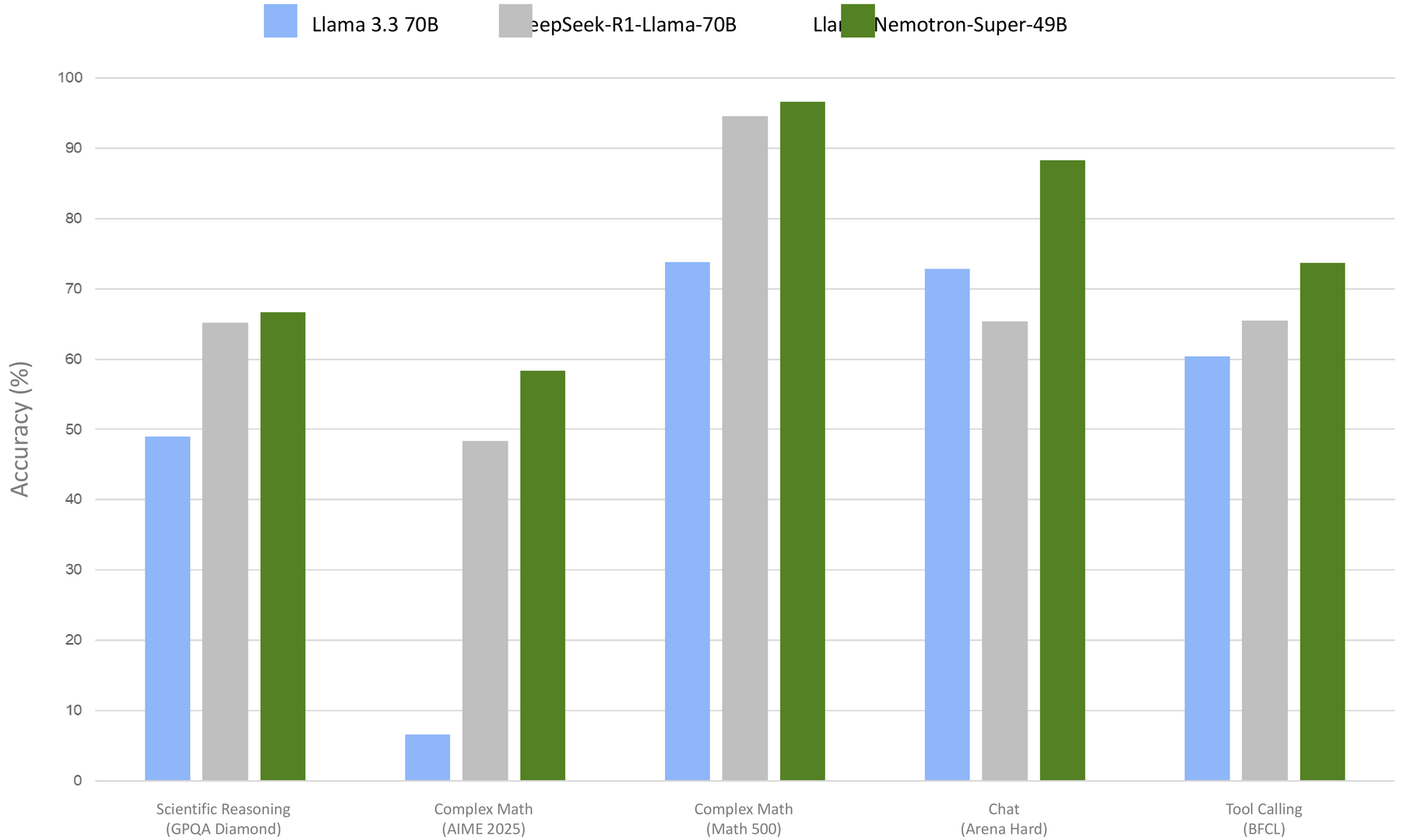


Accuracy scores of the leading open-weight models on the Artificial Analysis – GPQA benchmark for evaluating scientific reasoning



Llama Nemotron Super

オープンな70Bモデルの中でトップクラスのパフォーマンスを実現



Reasoning
GPQA Diamond, AIME, MATH

Chat
Arena Hard

Tool Calling
BFCL

Nemotron-H

Nemotron-H: A Family of Accurate and Efficient Hybrid Mamba-Transformer Models, Aaron Blakeman et al. 2025

Hybrid Mamba-Transformerモデルアーキテクチャ

Physical AI向けVLM、Cosmos-Reason 1のバックボーンとして採用

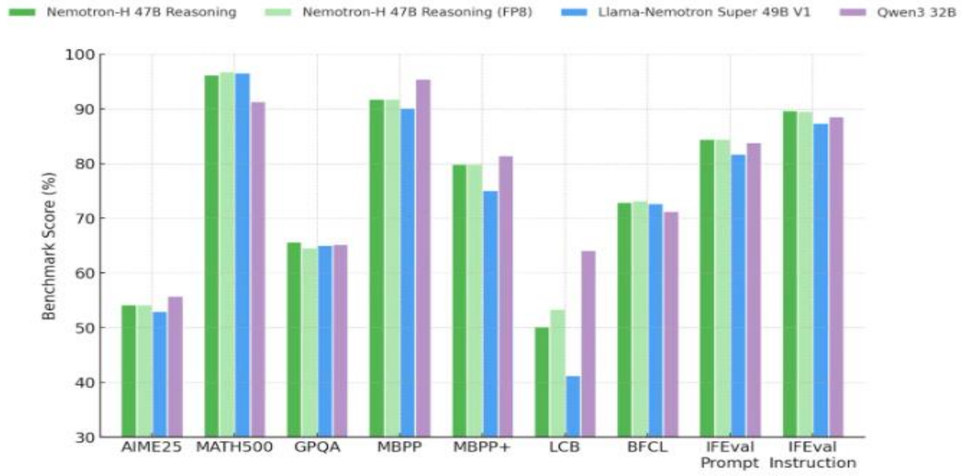
Llama-Nemotron Super 49B V1.0と比較して4倍近くのスループット

128Kトークンコンテキストをサポート

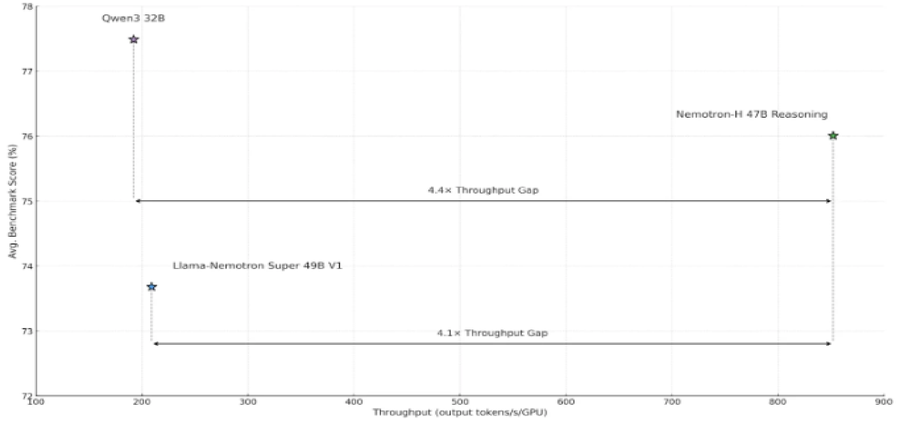
Current ScalingによるFP8学習の採用

NVIDIA Resiliency Extentions の使用

- **Nemotron-H-56B reasoning**
  - 56B パラメータ
  - 最大規模・最高精度
- **Nemotron-H-47B reasoning**
  - 47B パラメータ
  - 56Bからの蒸留モデル (MiniPuzzle)
  - 56Bとほぼ同等の精度を維持しつつ、推論速度やメモリ効率を重視
- **Nemotron-H-8B reasoning**
  - 8B パラメータ
  - 精度よりスループットやコストを重視
  - エッジデバイスや小規模サーバ等コスト制約がある環境向け



Benchmark scores for Nemotron-H 47B Reasoning, Llama-Nemotron Super V1, and Qwen3 32B. All models were evaluated using our internal evaluation pipeline to ensure consistency



Benchmark scores compared to throughput for Nemotron-H 47B Reasoning and competing models.

NVIDIA Resiliency Extensions

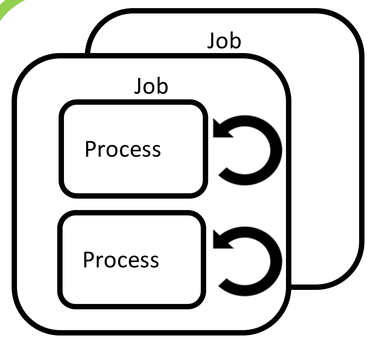
レジリエンス機能を構築するためのフレームワーク用の Python パッケージ

NVDIA Resiliency Extension (NVRx)とは?

- フレームワークとアプリケーションをレジリエンス機能で拡張するための **Python パッケージ**
  - pip install nvidia-resiliency-ext
- Github上でオープンソースとして公開
  - https://github.com/NVIDIA/nvidia-resiliency-ext

どのような問題の解決可能か?

- 実行時に障害を迅速に検出し、トレーニングを再開することで、グッドプットを最大化
- 高速かつ頻繁なチェックポイント作成を可能にすることで、作業のロスを削減
- さまざまなコンポーネントの健全性チェックを支援



In-Process
In-Job
…

自動再起動

Local Mem
Global
Distributed
Async
…

階層的チェックポイント

Hang
Straggler
User Error
…

フォルトディテクション

GPU
NvLink
NIC
…

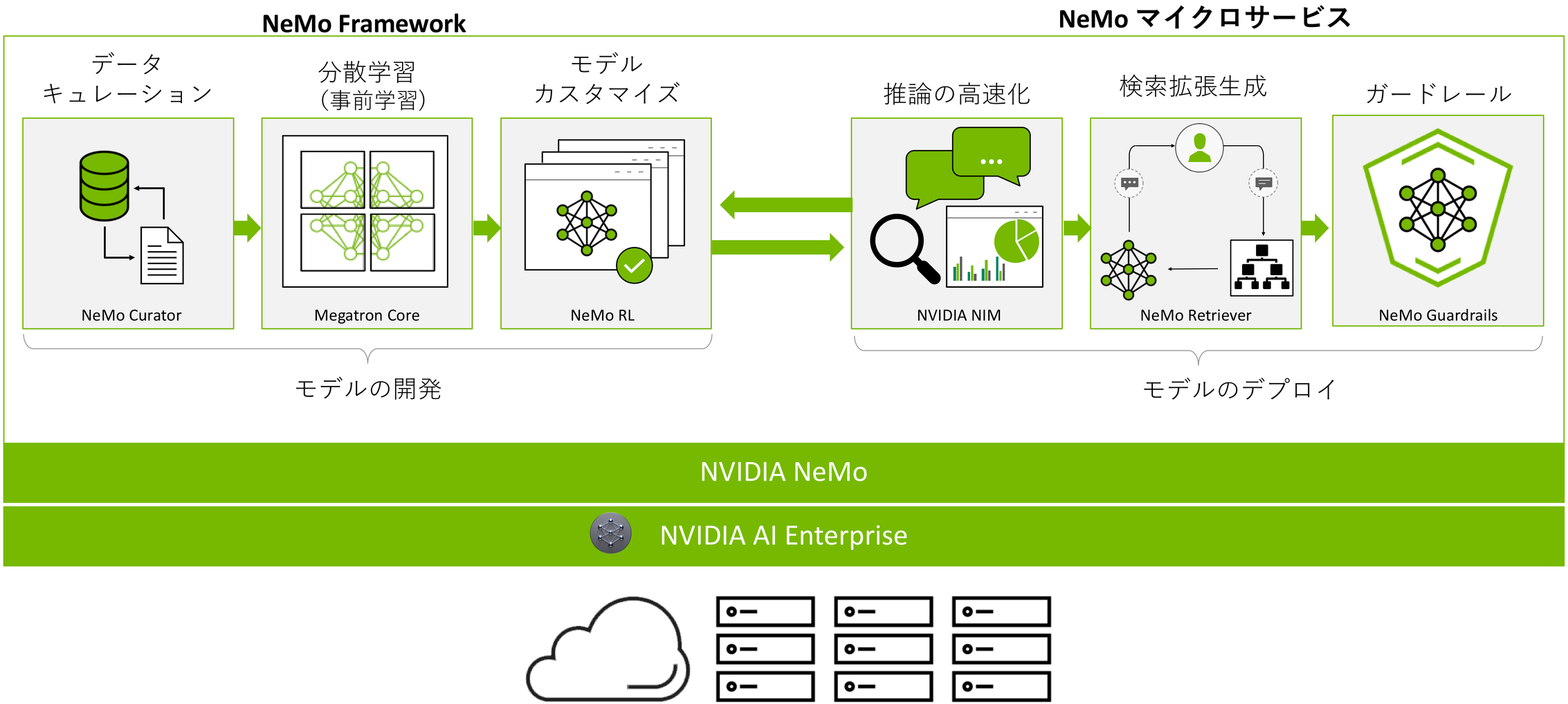
ヘルスチェック

NVIDIA AI: フルスタックアプローチ

NVIDIA AI Enterprise - エンドツーエンド生成 AI パイプラインの開発とデプロイを効率化

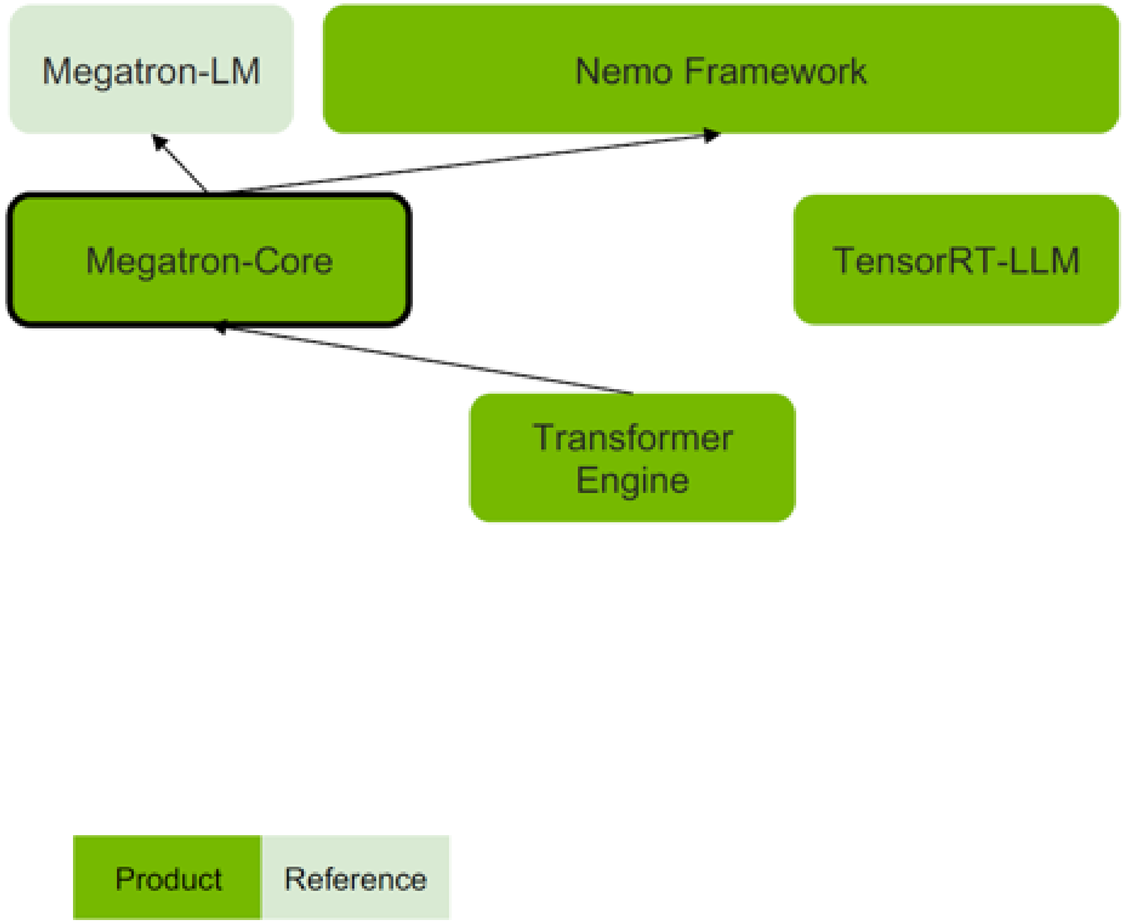
企業向け生成 AI アプリケーションの構築

NVIDIA NeMo による生成 AI モデルの構築、カスタマイズ、デプロイ



生成AIの大規模学習 大規模推論

NeMo Framework / Megatron-LM / Megatron-Core



- **NeMo Framework**：

エンタープライズユーザーが実験、学習、デプロイを行える、大規模なモデルコレクションを備えた使いやすいOOTB FW。

- **Megatron-LM**：

Megatron-Coreを使用して独自のLLMフレームワークを構築する為の軽量リファレンスフレームワーク

- **Megatron-Core**：

大規模なTransformerモデルをトレーニングするためのGPU最適化技術用ライブラリ。

- **Transformer Engine**：

Hopper, Ada, Blackwell GPUの為のTransformerモデルのアクセラレーションライブラリ。

- **TensorRT-LLM**：

NVIDIA GPU上の最新の大規模言語モデルで最適な推論パフォーマンスを実現

生成AI推論の最適化の為のTensorRT-LLM

NVIDIA GPU向け大規模言語モデルおよび VLMの為の推論ライブラリ

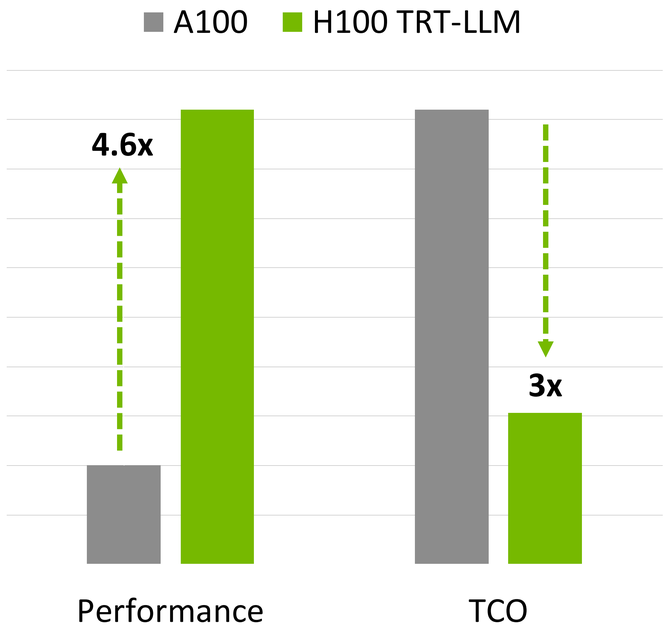
LLMの性能最適化は、リアルタイムでコスト効率の高い本番環境へのデプロイに不可欠。LLMエコシステムは急速に進化し、新しいモデルや手法が定期的にリリースされるため、モデルを最適化するための高性能で柔軟なソリューションが必要

TensorRT-LLM は、NVIDIA GPU向けの最新の大規模言語モデルおよびVLMの推論性能を最適化するオープンソースライブラリ

FasterTransformerとTensorRTを基盤とし、シンプルなPython APIを使用して、本番環境における推論用LLMの定義、最適化、実行が可能

最高峰の性能

Leverage TensorRT compilation & kernels from FasterTransformers, CUTLASS, OAI Triton, ++



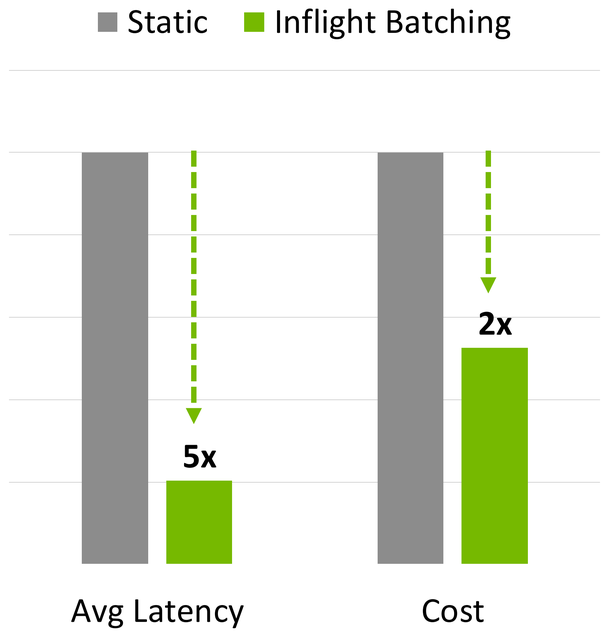
簡単に拡張

Add new operators or models in Python to quickly support new LLMs with optimized performance

```
# define a new activation
def silu(input: Tensor) → Tensor:
    return input * sigmoid(input)

#implement models like in DL FWs
class LlamaModel(Module)
  def __init__(…)
    self.layers = ModuleList([…])

  def forward (…)
    hidden = self.embedding(…)

    for layer in self.layers:
      hidden_states = layer(hidden)

    return hidden
```

Dynamo Triton対応

Maximize throughput and GPU utilization through new scheduling techniques for LLMs

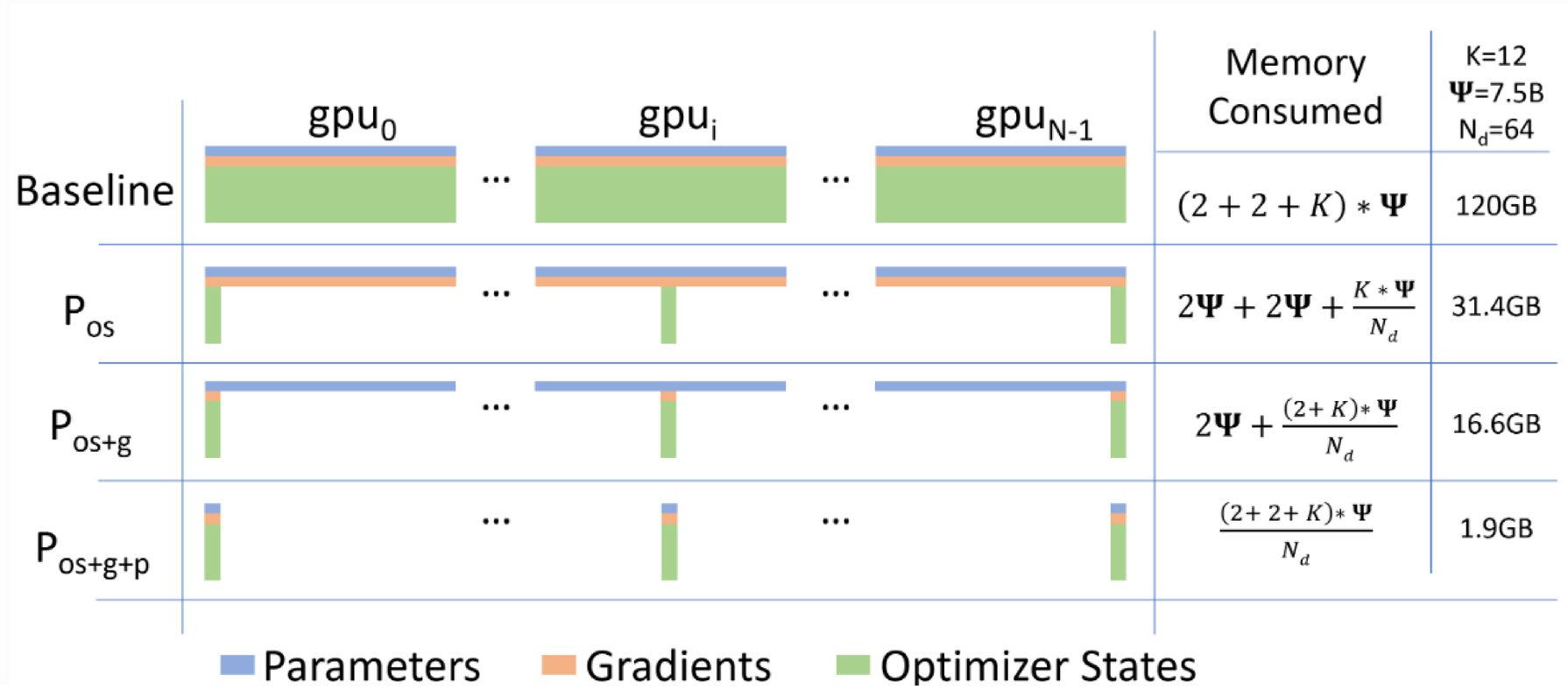


Numbers are preliminary based on internal evaluation on Llama 7B on H100

FSDP (Fully Shared Data Parallel)

代表的なモデル並列手法

- Optimizer States, Gradients, Parameterも各GPUに複製される為、各GPUのメモリ使用量を抑える効果がある
- その一方、 Optimizer States, Gradients, Parameterを各GPUにばらまく為、パラメータ数が多いモデルの場合、大規模通信が発生

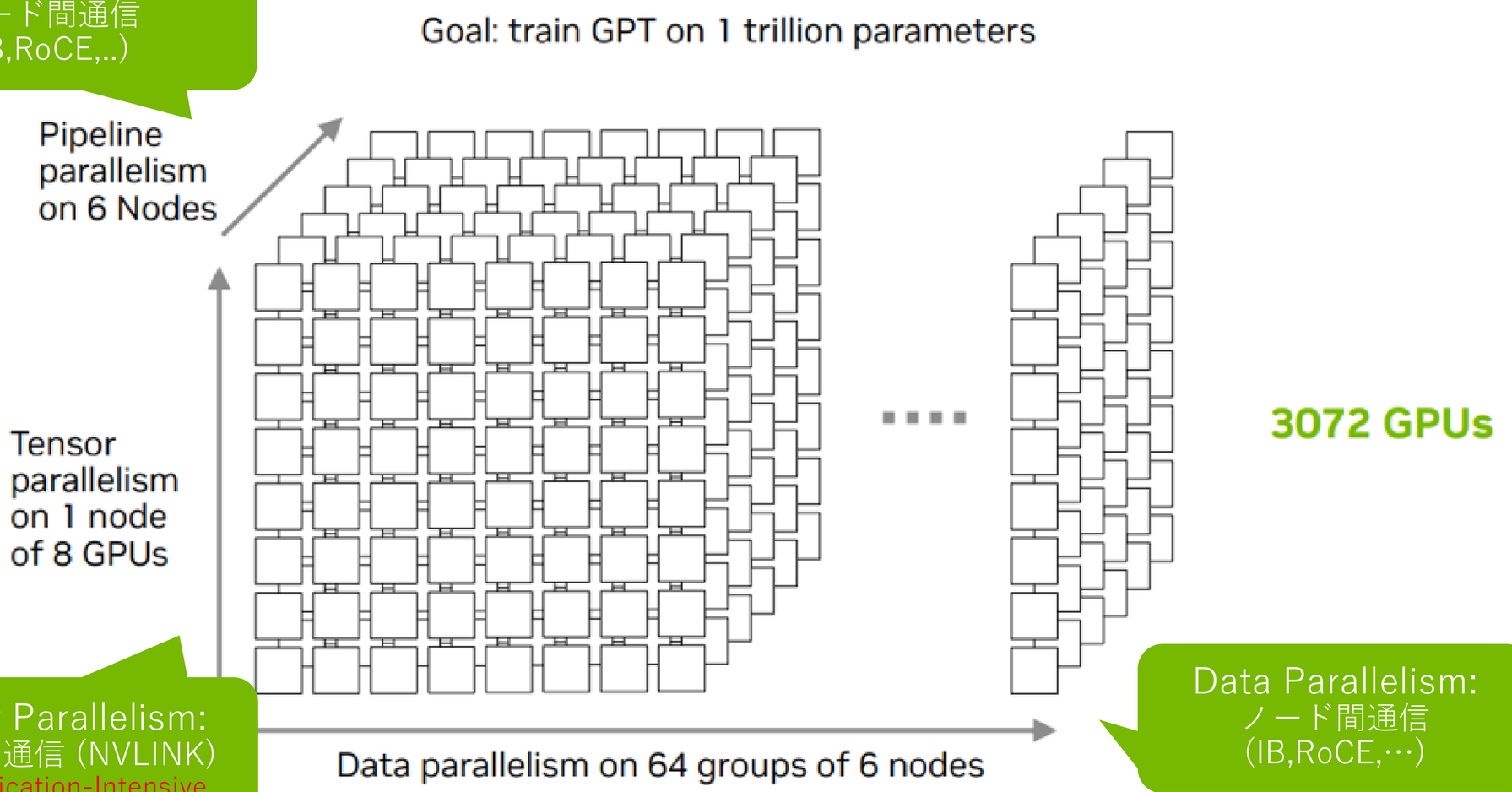


https://uvadlc-notebooks.readthedocs.io/en/latest/tutorial_notebooks/scaling/JAX/data_parallel_fsdp.html より引用

3D Parallelismによる分散学習 (NVIDIA Megatron)

Efficient Large-Scale Language Model Training on GPU Clusters, Deepak Narayanan et al., 2021

Pipeline Parallelism:
ノード間通信
(IB,RoCE,..)



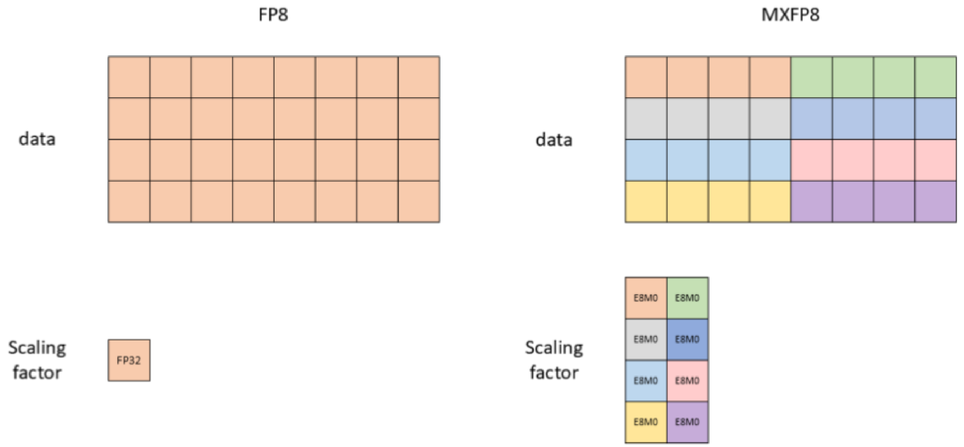
Tensor Parallelism:
ノード内通信 (NVLINK)
Communication-Intensive

Data Parallelism:
ノード間通信
(IB,RoCE,…)

Transformer Engine: FP8学習向けの対応状況

https://github.com/NVIDIA/TransformerEngine

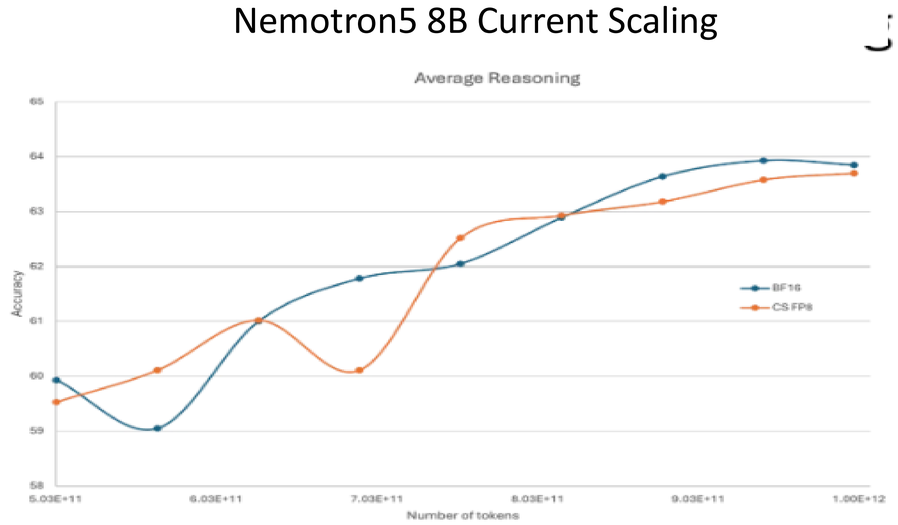
- Blackwell 向けに MXFP8形式をサポート (v2.0)
  - テンソル内の32個の値の各ブロックに異なるスケーリング係数を適用



FP8 and MXFP8 scaling factors

https://developer.nvidia.com/blog/floating-point-8-an-introduction-to-efficient-lower-precision-ai-training

- FP8 Block Scaling、MXFP8 Block Scalingのサポート (v2.3)
  - Hopper GPU向け、Deepseek v3 論文で提案
- FP8 Current Scalingのサポート (v2.4) 注1:



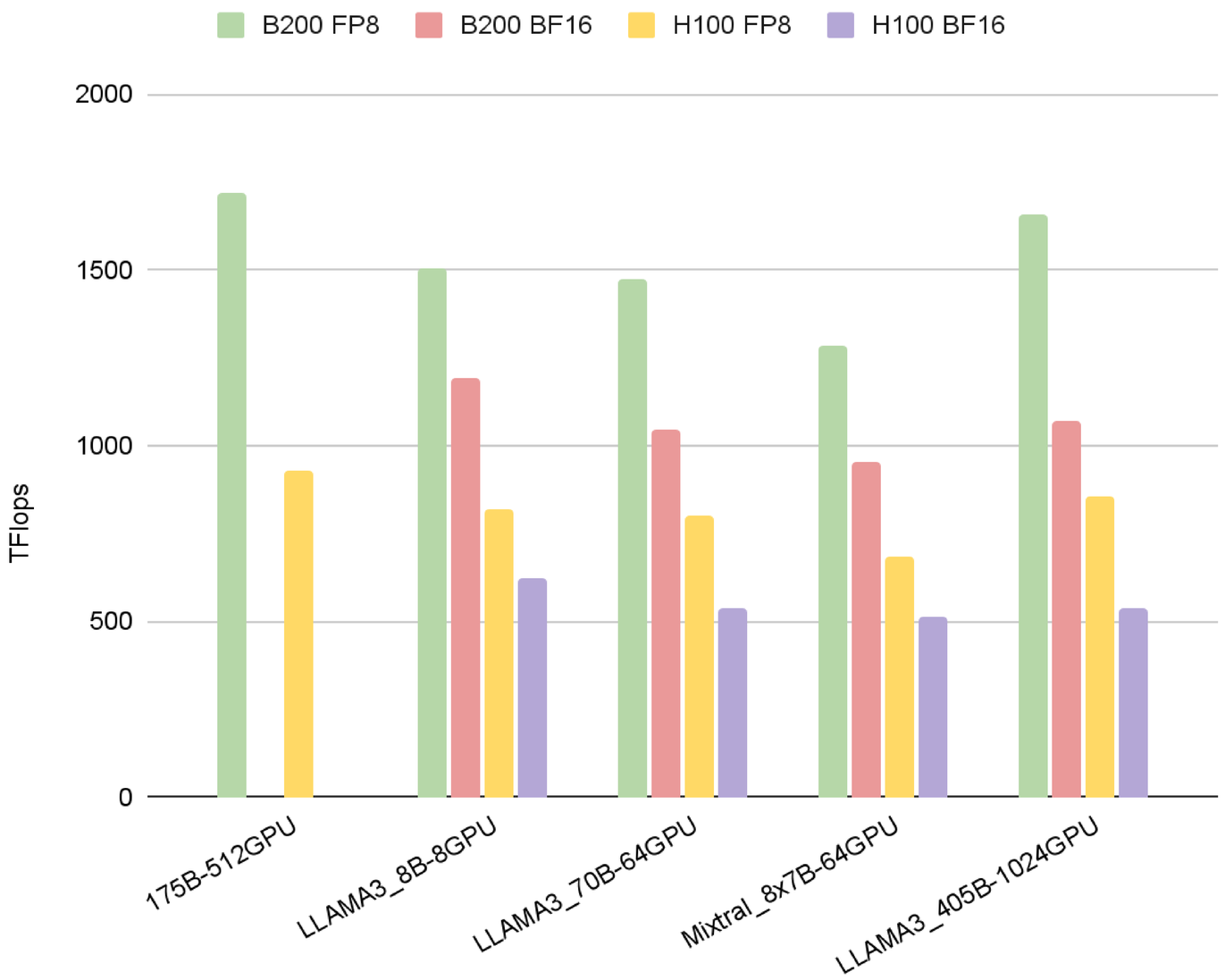
Massive Multitask Language Understanding (MMLU) and average reasoning evaluation of Nemotron5 8B BF16 compared to FP8

https://developer.nvidia.com/blog/per-tensor-and-per-block-scaling-strategies-for-effective-fp8-training/

注1: Current Scalingアルゴリズム自体はTE v2.0から対応

Pre-Training Benchmarks for Hopper and Blackwell

NeMo Framework (Megatron-Core) ベンチマーク



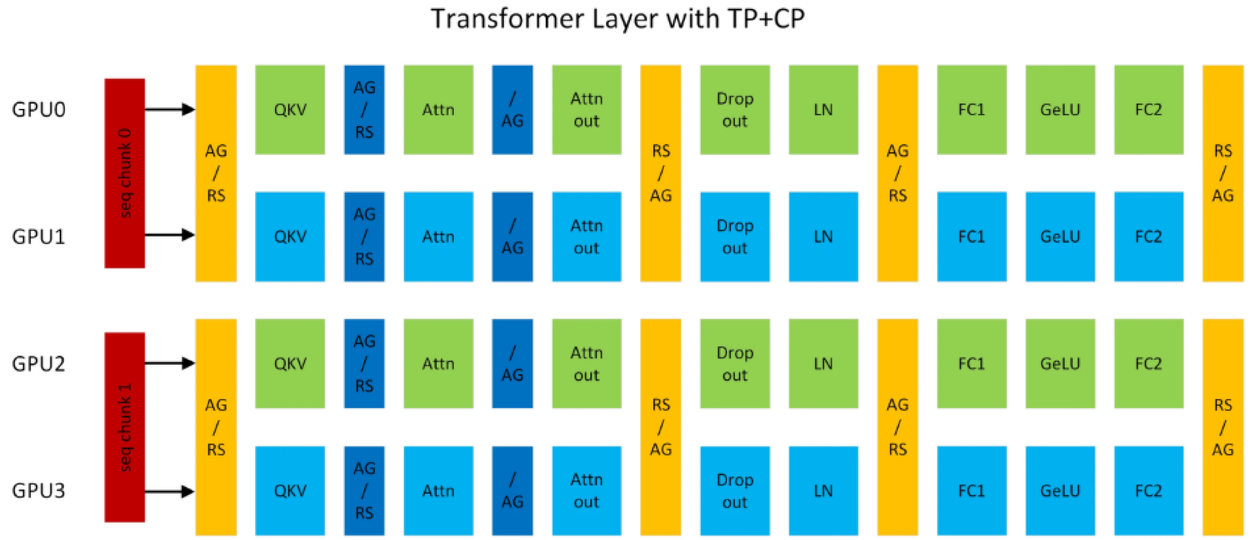
Megatron Core: Context Parallel Package

https://docs.nvidia.com/megatron-core/developer-guide/latest/api-guide/context_parallel.html

Megatron CoreのContext Parallelはシーケンス長の次元で並列化を行う手法

全結合層だけでなく、トークン間の演算が必要なAttention層の計算に対しても並列化

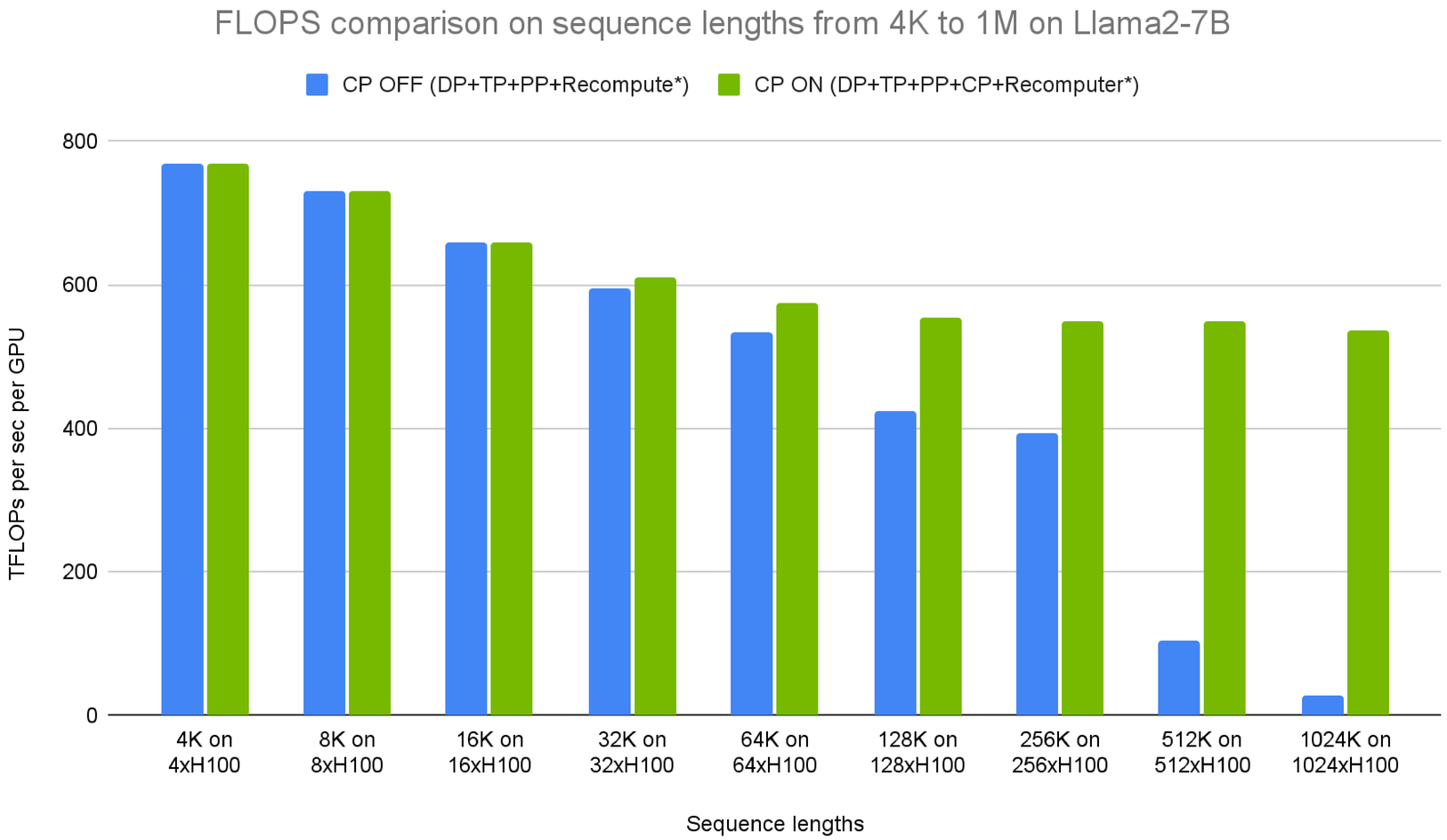
各GPUがシーケンスの一部だけ処理する為、各GPUが担当する計算量と通信量が削減される。ロングコンテキスト時に起こりやすいOOM問題も解消される



Column-split: QKV and FC1, Row-split: Attn-output and FC2, CP splits activations of whole transformer layer along seq dim
AG/RS: AG in fwd and RS in bwd, RS/AG: RS in fwd and AG in bwd, /AG: No-op in fwd and AG in bwd

Context Parallelism can provide up to 20x Speedup

Llama2-7B benchmark result



CP=1 for 4k to 16k, TP size is limited to 16. Assumed a fixed number of tokens per global batch (4M tokens / global batch).
For seq len cases (512K and 1024K), the maximum size of DP was very limited.
Results shown in FP8. Similar conclusions for BF16

New Feature: Sequence Packing Technique

NeMo Framework / NeMo-RL + Megatron Core (upcoming)

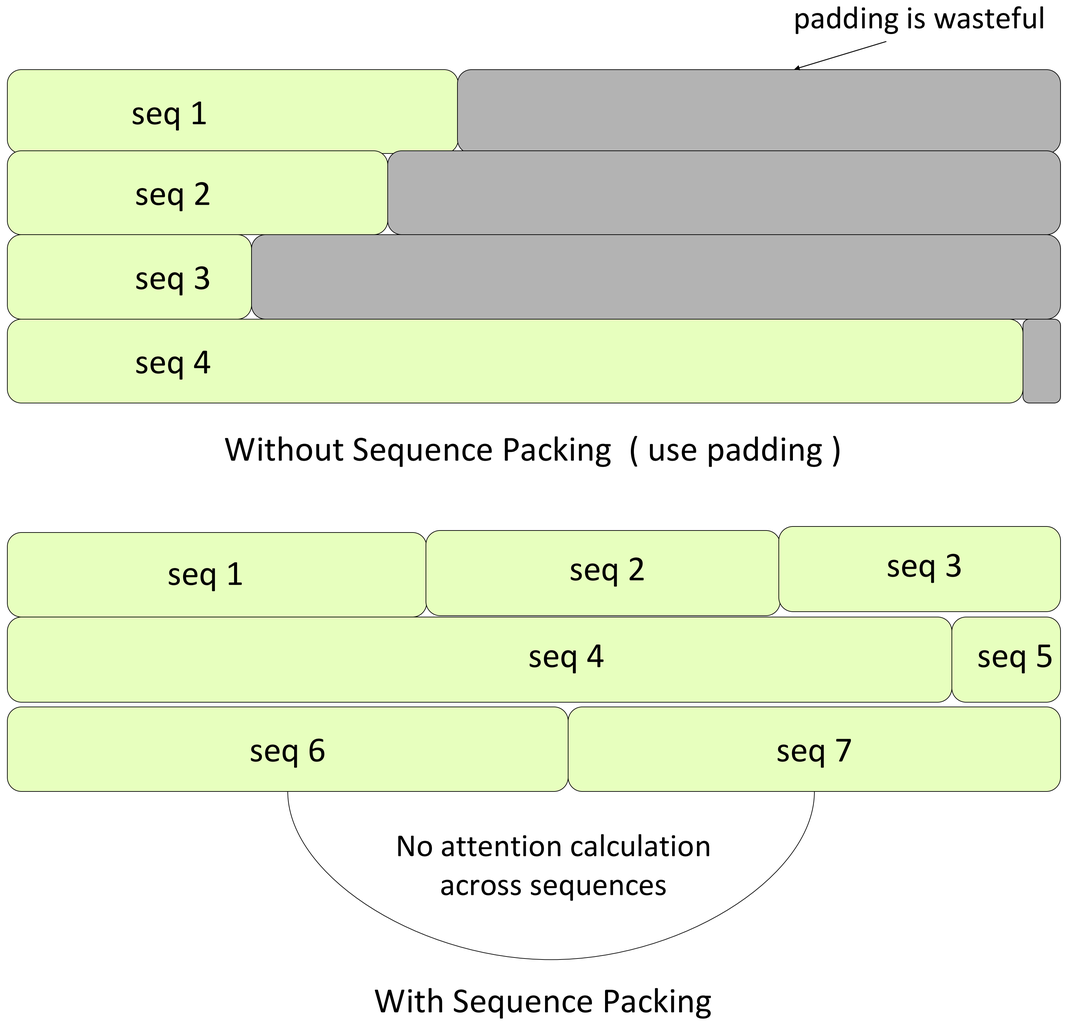
Sequence Packing の利点

- パディングが不要になる
- 各マイクロバッチでより多くのトークンが処理できる
  - シーケンス長の分布が偏っていて、短いシーケンスが多く、長いシーケンスが少数であるデータセットを用いたファインチューニングで特に効果がある

NeMo FrameworkではFlash AttentionとTransformer Engineの可変長アテンションを利用してSequence Packing機能を提供

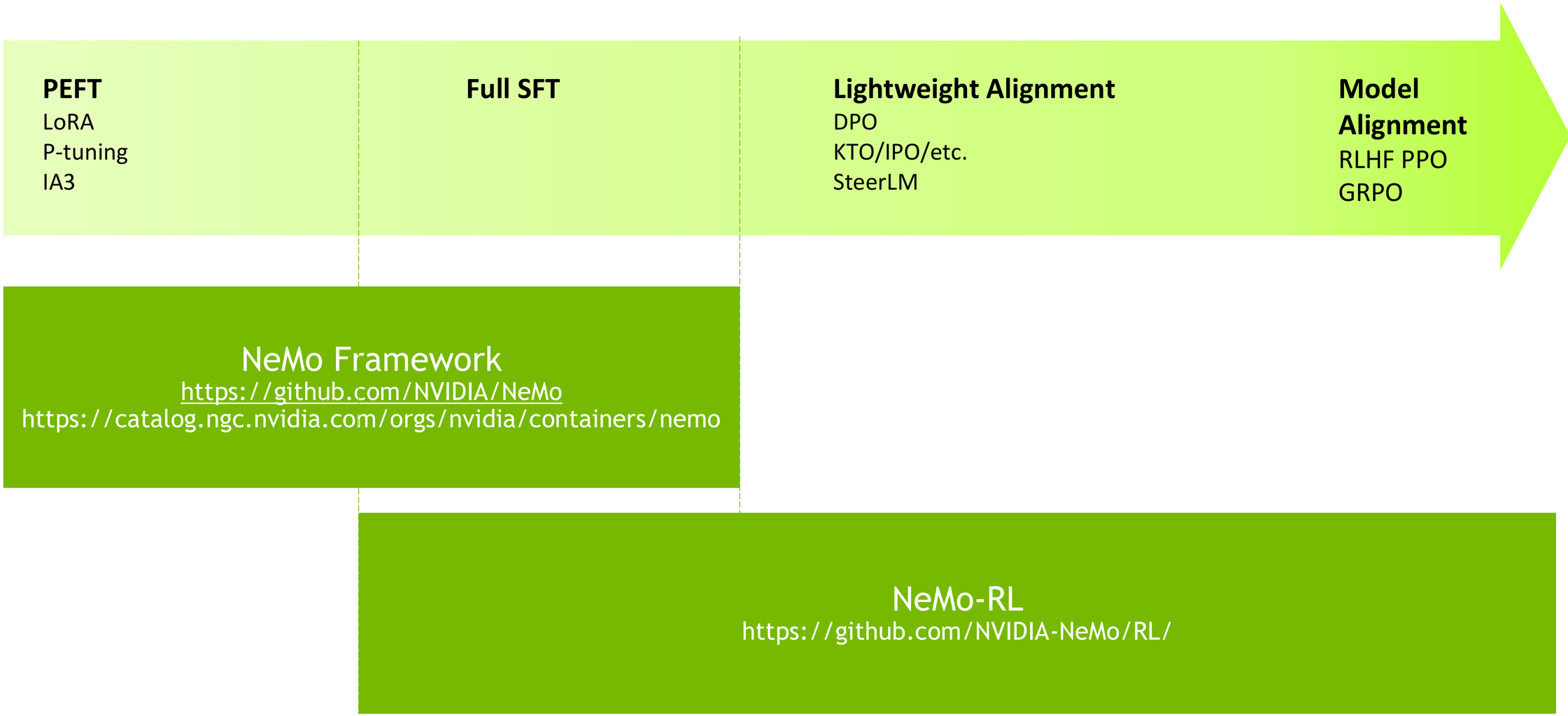
従来必要だったシーケンス間のアテンション計算を回避可能に

NeMo-RLのMegatron Coreバックエンドでも対応予定(NeMo-RL v0.3.0)



NVIDIA NeMoは様々なファインチューニング手法に対応

NeMo Framework/NeMo-RL

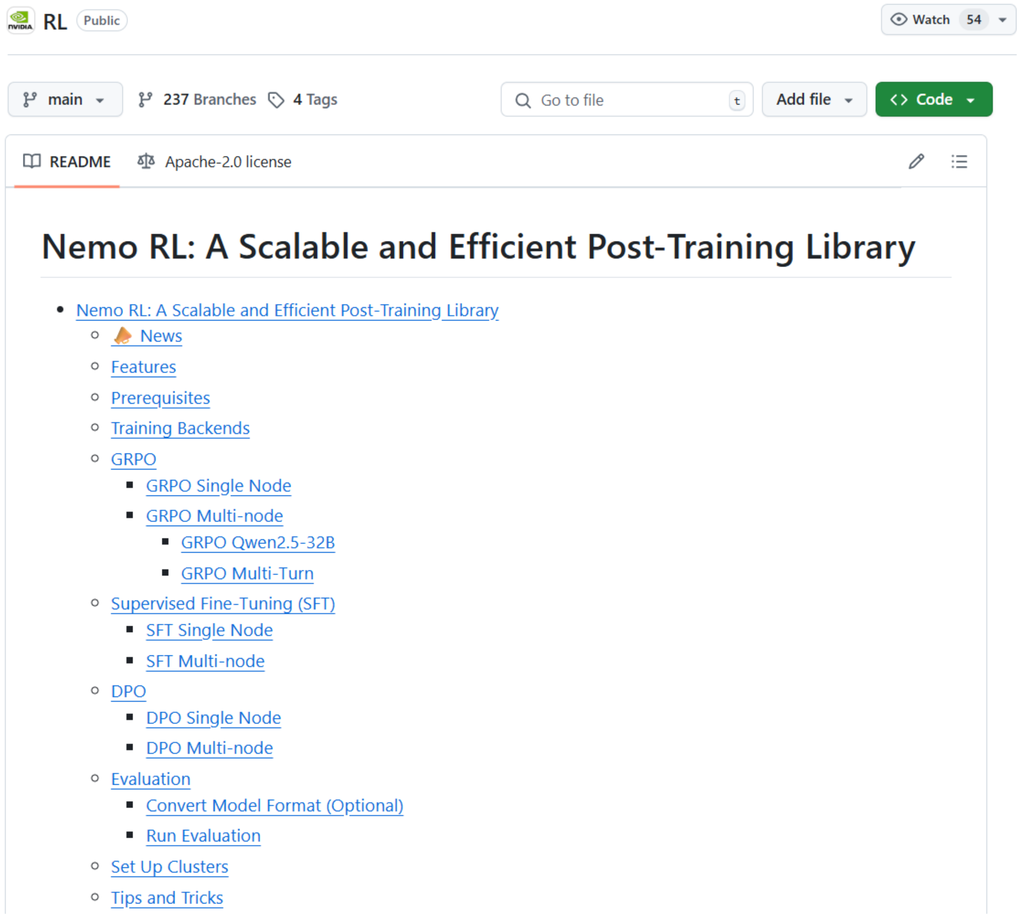


NeMo-RL

A Scalable and Efficient Post-Training Library

事後学習フレームワーク (NeMo-Aligner の後継)

- **簡単に使用可能**
  - HuggingFaceとのシームレスなインテグレーション
  - Rayによる効率的なリソースマネージメント
  - PyTorch FSDP2サポート
  - モデル(最大 32B)のネイティブ PyTorch サポート
- **SOTA ファインチューニング手法のサポート**：
  - SFT
  - DPO
  - GRPO
  - DAPO
  - PRIME (upcoming)
- **性能とスケーラビリティ**:
  - Megatron-Coreバックエンドのサポート (PR517)
    - 4D Parallelism
    - Context Parallel
    - Sequence Packing
  - TensorRT-LLMのサポート(upcoming)

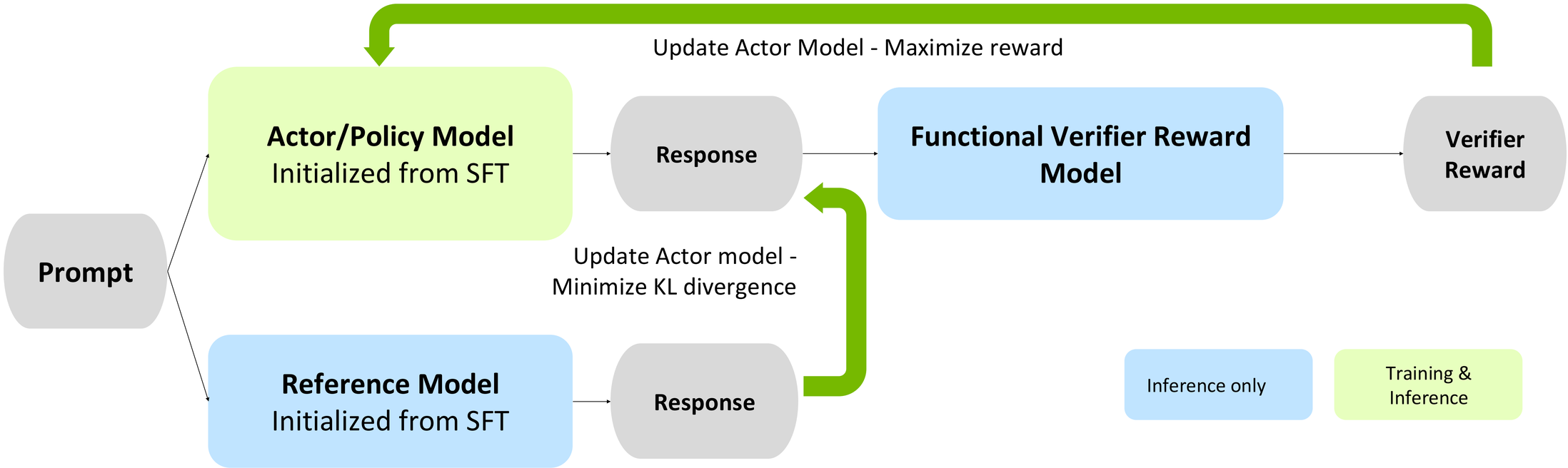


https://github.com/NVIDIA-NeMo/RL/

GRPOの概要

GRPO with Functional Verifier

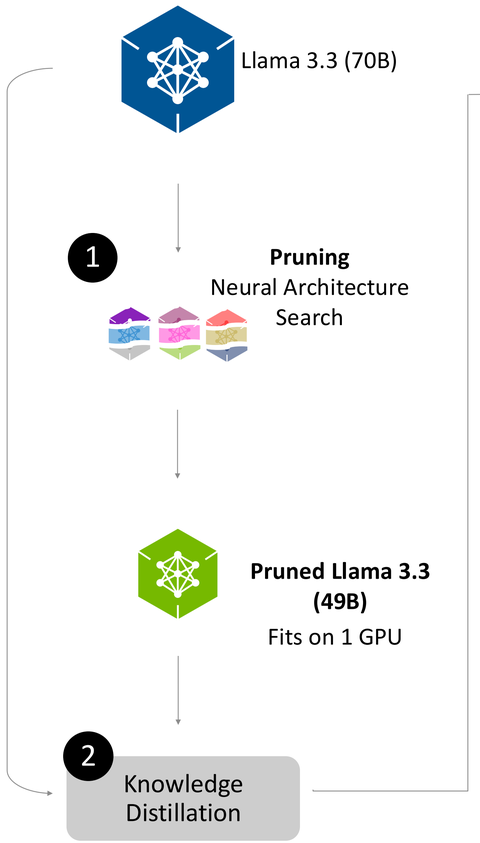
1. N個のプロンプトをサンプリングし、ポリシーネットワークからそれぞれ1つの応答を生成
2. 各プロンプトと応答のペアについて以下を計算
    1. 初期ポリシーにおける応答確率
    2. 応答に対する検証者の報酬
3. 初期ポリシーから大きく逸脱することなく応答確率が高くなるように検証者モデルをアップデート



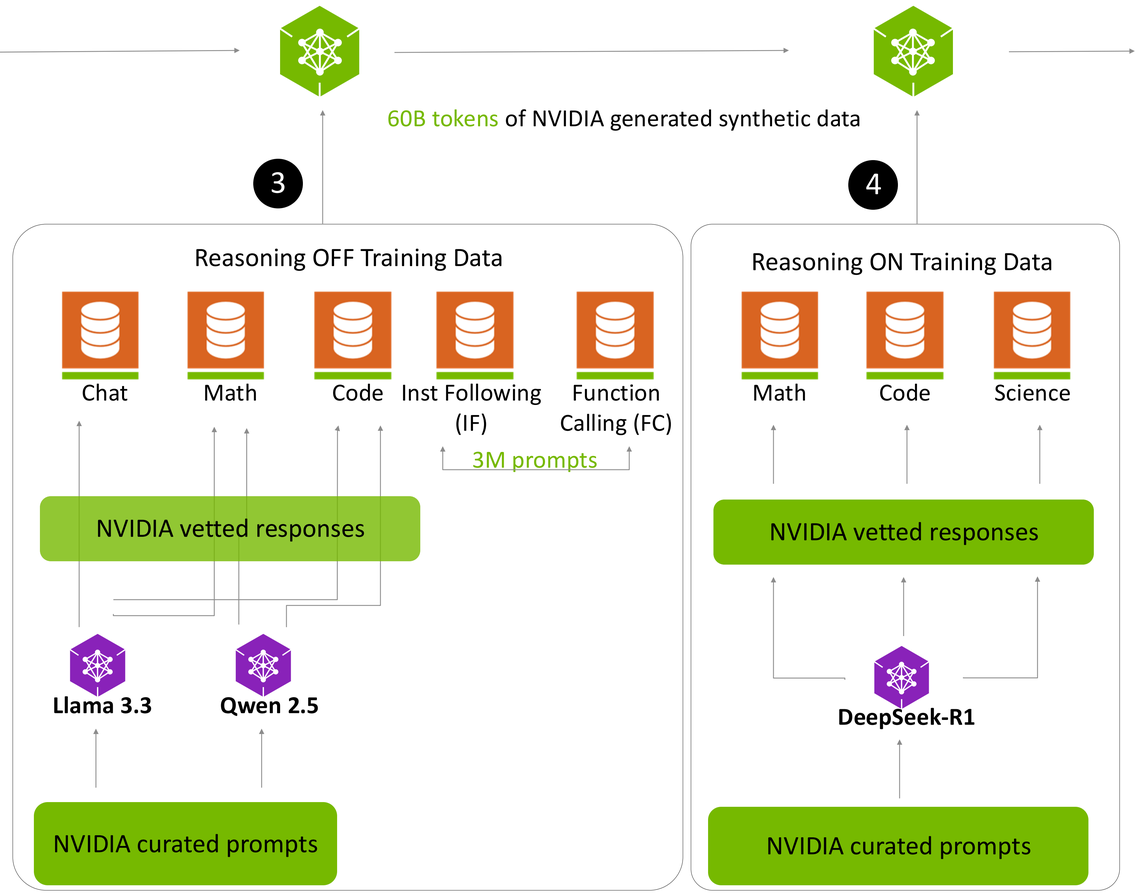
Example: Llama Nemotron with Reasoning

A Look Under The Hood

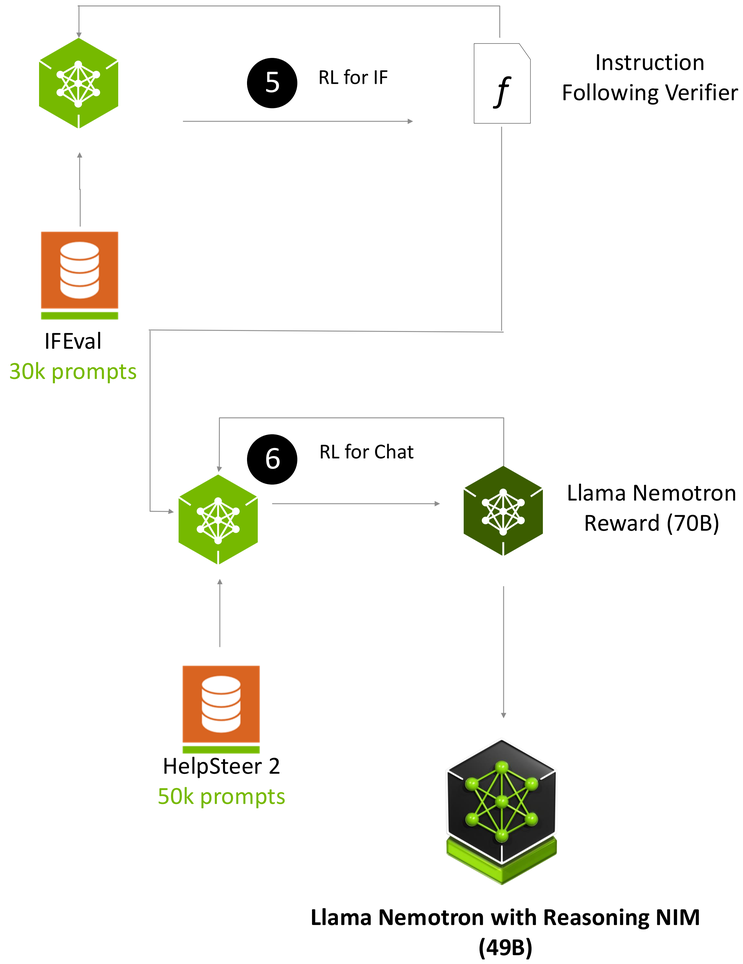
Distillation
Improve model efficiency



Supervised Fine-Tuning
Improving Agentic Skills with Reasoning



Reinforcement Learning (RL)
Aligning for Human Preferences



NVIDIA

まとめ

Key Take Aways

- 複数のツールを連携して高度なタスクを行うAIエージェントに注目が集まっている
- エージェンティックAIの為のモデルはモデル精度だけではなく、レイテンシの低さも極めて重要
- リーズニンモデルの登場によりTest-Time Scalingという概念が登場、モデルの推論の計算量が増加する傾向に。学習だけでなく推論の高速化がより重要に
- エヌビディアでは、エージェンティックAIの開発の為に必要な様々なツールをフルスタックで提供

Appendix.

- **Llama Nemotron**
  - [Technical Blog] NVIDIA Llama Nemotron Ultra Open Model Delivers Groundbreaking Reasoning Accuracy
  - https://arxiv.org/abs/2505.00949
  - Llama Nemotron Hugging Face リポジトリ
- **Nemotron-H**
  - https://research.nvidia.com/labs/adlr/nemotronh/
  - [Technical Blog] https://developer.nvidia.com/blog/nemotron-h-reasoning-enabling-throughput-gains-with-no-compromises
  - https://arxiv.org/abs/2504.03624
  - Nemotron-H HuggingFace リポジトリ
- **Other activities for nemotron**
  - https://research.nvidia.com/labs/adlr/cortexa/
  - https://huggingface.co/datasets/nvidia/Nemotron-Personas
  - https://huggingface.co/datasets/nvidia/Llama-Nemotron-Post-Training-Dataset
- **Transformer Engine**
  - https://github.com/NVIDIA/TransformerEngine
- **Megatron Core (+ Megatron-LM)**
  - https://github.com/NVIDIA/Megatron-LM
- **NVIDIA NeMo Framework**
  - https://catalog.ngc.nvidia.com/orgs/nvidia/containers/nemos
- **NeMo-RL**
  - https://github.com/NVIDIA-NeMo/RL
- **TensorRT-LLM**
  - https://github.com/NVIDIA/TensorRT-LLM