SGLang

Release Notes

RN-08516-001_v26.05 | June 2026

Table of Contents

Chapter 1. SGLang Overview

SGLang is a high-performance runtime engine and structured generation language for Large Language Models (LLMs). It offers a flexible Python interface and integrates smoothly with models from popular ecosystems like Hugging Face. At its core is RadixAttention, a novel GPU kernel that efficiently manages the KV Cache for complex generation programs involving loops and conditionals. This innovation enables massive parallelization of different program branches, eliminates redundant computation, and delivers state-of-the-art performance for structured prompting tasks.

SGLang also features a simple yet powerful language front-end, allowing developers to mix text generation, control flow, and external tools within a single program. These programs are aggressively optimized by the SGLang compiler and executed by a runtime featuring continuous batching and an efficient, fragmentation-free memory pool. This integrated approach simplifies the development of complex LLM applications and unlocks significant speedups, especially for tasks like multi-turn conversation, in-context learning, and agentic workflows.

For more information about SGLang, including documentation and examples, see:

- SGLang website

Chapter 2. Pulling A Container

About this task

Using the vLLM NGC Container requires the host system to have the following installed:

- Docker Engine
- NVIDIA GPU Drivers
- NVIDIA Container Toolkit

For supported versions, see the Framework Containers Support Matrix and the NVIDIA Container Toolkit Documentation.

No other installation, compilation, or dependency management is required. It is not necessary to install the NVIDIA CUDA Toolkit

Chapter 3. Running SGLang

Before you begin

Before you can run an NGC deep learning framework container, your Docker® environment must support NVIDIA GPUs. To run a container, issue the appropriate command as explained in Running A Container and specify the registry, repository, and tags.

About this task

To run a container, issue the appropriate command as explained in the Running A Container chapter in the NVIDIA Containers For Deep Learning Frameworks User's Guide and specify the registry, repository, and tags. For more information about using NGC, refer to the NGC Container User Guide.

If you have Docker 19.03 or later, a typical command to launch the container is:

```
docker run --gpus all -it --rm nvcr.io/nvidia/slgang:xx.yy-py3
```

If you have Docker 19.02 or later, a typical command to launch the container is:

```
nvidia-docker run -it --rm -v nvcr.io/nvidia/slang:xx.yy-py3
```

Where:

- *xx.yy* is the container version.

SGLang can be deployed in a client–server configuration. Start the HTTP inference server inside the container:

```
python3 -m sglang.launch_server --model-path nvidia/Llama-3.1-8B-Instruct-FP4 --host 0.0.0.0
 --port 30000 --trust-remote-code --tp 1 --quantization modelopt_fp4 &
```

From a client, issue a text-generation request by POST-ing to /generate with a JSON body containing the prompt and sampling parameters:

```
curl http://localhost:30000/v1/chat/completions \
      -H "Content-Type: application/json" \
      -d '{ "model": "nvidia/Llama-3.1-8B-Instruct-FP4", "messages": [{"role": "user",
 "content": "What is NVIDIA famous for?"}], "max_tokens": 1000 }'
```

See `/workspace/README.md` inside the container for information on getting started and customizing your SGang image.

You might want to pull in data and model descriptions from locations outside the container for use by SGLang. To accomplish this, the easiest method is to mount one or more host directories as Docker bind mounts. For example:

```
docker run --gpus all -it --rm -v local_dir:container_dir nvcr.io/nvidia/sglang:xx.xx-py3
```

Chapter 4. SGLang Release 26.04

This SGLang container release is intended for use on the NVIDIA® Hopper Architecture GPU, NVIDIA H100, the NVIDIA® Ampere Architecture GPU, NVIDIA A100, and the associated NVIDIA CUDA® 12 and NVIDIA cuDNN 9 libraries.

Driver Requirements

Release 26.04 is based on CUDA 13.2.1.009 which requires NVIDIA Driver release 570 or later. However, if you are running on a data center GPU (for example, B100, L40, or any other data center GPU), you can use NVIDIA driver release 470.57 (or later R470), 525.85 (or later R525), 535.86 (or later R535), or 550.54 (or later R550) in forward-compatibility mode.

The CUDA driver's compatibility package only supports particular drivers. Thus, users should upgrade from all R418, R440, R450, R460, R510, R520, R530, R545 and R555 and R560 drivers, which are not forward-compatible with CUDA 12.8. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

Contents of the SGLang container

This container image contains the complete source of the version of SGLang in `/opt/sglang`. It is pre-built and installed in the Python default environment `/usr/local/lib/python3.12/dist-packages/sglang/` in the container image. Visit SGLang Docs to learn more about SGLang.

The NVIDIA SGLang Container is optimized for use with NVIDIA GPUs, and contains the following software for GPU acceleration.

- Please see the CUDA section for the list of libraries inherited from CUDA container.
- NVIDIA CUDA 13.2.1.009
- SGLang 0.5.9
- flashinfer 0.6.7
- transformers 4.57.1
- flash-attention 2.7.4.post1
- xgrammar 0.1.32

- 2.12.0a0+0291f960b6

Driver Requirements

Release 26.04 is based on CUDA 13.2.1 For comprehensive and up-to-date driver compatibility information, please refer to the following documentation:

- NVIDIA CUDA Compatibility Guide - Compatibility information between CUDA versions and driver releases
- CUDA Toolkit Release Notes - Driver version requirements and compatibility matrices
- NVIDIA Drivers Download - Latest NVIDIA drivers

Key Features and Enhancements

This SGLang release includes the following key features and enhancements.

- Compatibility with CUDA 13.1
- Support for multi-node configurations.
- GB300/B300 support.
- RTX PRO™ 6000 Blackwell Server Edition support.
- DGX Spark support.
- Jetson Thor support.
- Support for 8-bit floating point (FP8) precision on Hopper GPUs and above.
- Support NVIDIA innovative 4-bit floating point NVFP4 format on Blackwell GPUs (including Jetson Thor and DGX Spark), which provides better training and inference performance with lower memory utilization.
- Supported for DeepSeek-R1, Llama-3.1-8B-Instruct.
- Support for openai/gpt-oss-20b and openai/gpt-oss-120b.
- Improved stability and performance for Nemotron Super NVFP4 on Spark

Announcements

- 25.10 is the first NVIDIA SGLang container release that brings optimizations for NVIDIA GPUs.
- 26.03.post1 released:
  - Improved stability for Nemotron Super V3 NVFP4 on Spark

Known Issues

- gpt-oss family models cannot run on DGX Spark and Jetson Thor due to a OpenAI Triton issue.
- FP8 models are failing on Thor.
- MTP is not supported for NVIDIA-Nemotron-3-Super models.

Chapter 5. SGLang Release 26.03

This SGLang container release is intended for use on the NVIDIA® Hopper Architecture GPU, NVIDIA H100, the NVIDIA® Ampere Architecture GPU, NVIDIA A100, and the associated NVIDIA CUDA® 12 and NVIDIA cuDNN 9 libraries.

Driver Requirements

Release 26.03 is based on CUDA 13.2.0.046 which requires NVIDIA Driver release 570 or later. However, if you are running on a data center GPU (for example, B100, L40, or any other data center GPU), you can use NVIDIA driver release 470.57 (or later R470), 525.85 (or later R525), 535.86 (or later R535), or 550.54 (or later R550) in forward-compatibility mode.

The CUDA driver's compatibility package only supports particular drivers. Thus, users should upgrade from all R418, R440, R450, R460, R510, R520, R530, R545 and R555 and R560 drivers, which are not forward-compatible with CUDA 12.8. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

Contents of the SGLang container

This container image contains the complete source of the version of SGLang in `/opt/sglang`. It is pre-built and installed in the Python default environment `/usr/local/lib/python3.12/dist-packages/sglang/` in the container image. Visit SGLang Docs to learn more about SGLang.

The NVIDIA SGLang Container is optimized for use with NVIDIA GPUs, and contains the following software for GPU acceleration.

- Please see the CUDA section for the list of libraries inherited from CUDA container.
- NVIDIA CUDA 13.2.0.046
- SGLang 0.5.9
- flashinfer 0.6.7
- transformers 4.57.1
- flash-attention 2.7.4.post1
- xgrammar 0.1.32

- 2.11.0a0+a6c236b9fd1

Driver Requirements

Release 26.03 is based on CUDA 13.2.0 For comprehensive and up-to-date driver compatibility information, please refer to the following documentation:

- NVIDIA CUDA Compatibility Guide - Compatibility information between CUDA versions and driver releases
- CUDA Toolkit Release Notes - Driver version requirements and compatibility matrices
- NVIDIA Drivers Download - Latest NVIDIA drivers

Key Features and Enhancements

This SGLang release includes the following key features and enhancements.

- Compatibility with CUDA 13.1
- Support for multi-node configurations.
- GB300/B300 support.
- RTX PRO™ 6000 Blackwell Server Edition support.
- DGX Spark support.
- Jetson Thor support.
- Support for 8-bit floating point (FP8) precision on Hopper GPUs and above.
- Support NVIDIA innovative 4-bit floating point NVFP4 format on Blackwell GPUs (including Jetson Thor and DGX Spark), which provides better training and inference performance with lower memory utilization.
- Supported for DeepSeek-R1, Llama-3.1-8B-Instruct.
- Support for openai/gpt-oss-20b and openai/gpt-oss-120b.
- Improved stability and performance for Nemotron Super NVFP4 on Spark

Announcements

- 25.10 is the first NVIDIA SGLang container release that brings optimizations for NVIDIA GPUs.
- 26.03.post1 released:
  - Improved stability for Nemotron Super V3 NVFP4 on Spark

Known Issues

- gpt-oss family models cannot run on DGX Spark and Jetson Thor due to a OpenAI Triton issue.
- FP8 models are failing on Thor.
- MTP is not supported for NVIDIA-Nemotron-3-Super models.

Chapter 6. SGLang Release 26.02

This SGLang container release is intended for use on the NVIDIA® Hopper Architecture GPU, NVIDIA H100, the NVIDIA® Ampere Architecture GPU, NVIDIA A100, and the associated NVIDIA CUDA® 12 and NVIDIA cuDNN 9 libraries.

Driver Requirements

Release 26.02 is based on CUDA 13.1.1.006 which requires NVIDIA Driver release 570 or later. However, if you are running on a data center GPU (for example, B100, L40, or any other data center GPU), you can use NVIDIA driver release 470.57 (or later R470), 525.85 (or later R525), 535.86 (or later R535), or 550.54 (or later R550) in forward-compatibility mode.

The CUDA driver's compatibility package only supports particular drivers. Thus, users should upgrade from all R418, R440, R450, R460, R510, R520, R530, R545 and R555 and R560 drivers, which are not forward-compatible with CUDA 12.8. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

Contents of the SGLang container

This container image contains the complete source of the version of SGLang in `/opt/sglang`. It is pre-built and installed in the Python default environment `/usr/local/lib/python3.12/dist-packages/sglang/` in the container image. Visit SGLang Docs to learn more about SGLang.

The NVIDIA SGLang Container is optimized for use with NVIDIA GPUs, and contains the following software for GPU acceleration.

- Please see the CUDA section for the list of libraries inherited from CUDA container.
- NVIDIA CUDA 13.1.1.006
- SGLang 0.5.8
- flashinfer 0.6.1
- transformers 4.57.1
- flash-attention 2.7.4.post1
- xgrammar 0.1.27

- 2.11.0a0+eb65b36914

Driver Requirements

Release 26.02 is based on CUDA 13.1.1 For comprehensive and up-to-date driver compatibility information, please refer to the following documentation:

- NVIDIA CUDA Compatibility Guide - Compatibility information between CUDA versions and driver releases
- CUDA Toolkit Release Notes - Driver version requirements and compatibility matrices
- NVIDIA Drivers Download - Latest NVIDIA drivers

Key Features and Enhancements

This SGLang release includes the following key features and enhancements.

- Compatibility with CUDA 13.1
- Support for multi-node configurations.
- GB300/B300 support.
- RTX PRO™ 6000 Blackwell Server Edition support.
- DGX Spark support.
- Jetson Thor support.
- Support for 8-bit floating point (FP8) precision on Hopper GPUs and above.
- Support NVIDIA innovative 4-bit floating point NVFP4 format on Blackwell GPUs (including Jetson Thor and DGX Spark), which provides better training and inference performance with lower memory utilization.
- Supported for DeepSeek-R1, Llama-3.1-8B-Instruct.
- Support for openai/gpt-oss-20b and openai/gpt-oss-120b.

Announcements

- 25.10 is the first NVIDIA SGLang container release that brings optimizations for NVIDIA GPUs.

Known Issues

- gpt-oss family models cannot run on DGX Spark and Jetson Thor due to a OpenAI Triton issue.
- FP8 models are failing on Thor.
- When using Nemotron Nano-V2-9B or Nemotron Nano-V3-30B models on Spark/Thor with SGLang the user will need to install the flashinfer-jit-cache before running.

Chapter 7. SGLang Release 26.01

This SGLang container release is intended for use on the NVIDIA® Hopper Architecture GPU, NVIDIA H100, the NVIDIA® Ampere Architecture GPU, NVIDIA A100, and the associated NVIDIA CUDA® 12 and NVIDIA cuDNN 9 libraries.

Driver Requirements

Release 26.01 is based on CUDA 13.1.0.36 which requires NVIDIA Driver release 570 or later. However, if you are running on a data center GPU (for example, B100, L40, or any other data center GPU), you can use NVIDIA driver release 470.57 (or later R470), 525.85 (or later R525), 535.86 (or later R535), or 550.54 (or later R550) in forward-compatibility mode.

The CUDA driver's compatibility package only supports particular drivers. Thus, users should upgrade from all R418, R440, R450, R460, R510, R520, R530, R545 and R555 and R560 drivers, which are not forward-compatible with CUDA 12.8. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

Contents of the SGLang container

This container image contains the complete source of the version of SGLang in `/opt/sglang`. It is pre-built and installed in the Python default environment `/usr/local/lib/python3.12/dist-packages/sglang/` in the container image. Visit SGLang Docs to learn more about SGLang.

The NVIDIA SGLang Container is optimized for use with NVIDIA GPUs, and contains the following software for GPU acceleration.

- Please see the CUDA section for the list of libraries inherited from CUDA container.
- NVIDIA CUDA 13.1.0.36
- SGLang 0.5.5.post2
- flashinfer 0.5.2
- transformers 4.57.1
- flash-attention 2.7.4.post1
- xgrammar 0.1.25

- PyTorch2.10.0a0+a36e1d39eb

Driver Requirements

Release 26.01 is based on CUDA 13.1.1 For comprehensive and up-to-date driver compatibility information, please refer to the following documentation:

- NVIDIA CUDA Compatibility Guide - Compatibility information between CUDA versions and driver releases
- CUDA Toolkit Release Notes - Driver version requirements and compatibility matrices
- NVIDIA Drivers Download - Latest NVIDIA drivers

Key Features and Enhancements

This SGLang release includes the following key features and enhancements.

- Compatibility with CUDA 13.0
- Support for multi-node configurations.
- GB300/B300 support.
- RTX PRO™ 6000 Blackwell Server Edition support.
- DGX Spark support.
- Jetson Thor support.
- Support for 8-bit floating point (FP8) precision on Hopper GPUs and above.
- Support NVIDIA innovative 4-bit floating point NVFP4 format on Blackwell GPUs (including Jetson Thor and DGX Spark), which provides better training and inference performance with lower memory utilization.
- Supported for DeepSeek-R1, Llama-3.1-8B-Instruct.
- Support for openai/gpt-oss-20b and openai/gpt-oss-120b.

Announcements

- 25.10 is the first NVIDIA SGLang container release that brings optimizations for NVIDIA GPUs.

Known Issues

- gpt-oss family models cannot run on DGX Spark and Jetson Thor due to a OpenAI Triton issue.
- FP8 models are failing on Thor.

Chapter 8. SGLang Release 25.12

This SGLang container release is intended for use on the NVIDIA® Hopper Architecture GPU, NVIDIA H100, the NVIDIA® Ampere Architecture GPU, NVIDIA A100, and the associated NVIDIA CUDA® 12 and NVIDIA cuDNN 9 libraries.

Driver Requirements

Release 25.12 is based on CUDA 13.1.0.36 which requires NVIDIA Driver release 570 or later. However, if you are running on a data center GPU (for example, B100, L40, or any other data center GPU), you can use NVIDIA driver release 470.57 (or later R470), 525.85 (or later R525), 535.86 (or later R535), or 550.54 (or later R550) in forward-compatibility mode.

The CUDA driver's compatibility package only supports particular drivers. Thus, users should upgrade from all R418, R440, R450, R460, R510, R520, R530, R545 and R555 and R560 drivers, which are not forward-compatible with CUDA 12.8. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

Contents of the SGLang container

This container image contains the complete source of the version of SGLang in `/opt/sglang`. It is pre-built and installed in the Python default environment `/usr/local/lib/python3.12/dist-packages/sglang/` in the container image. Visit SGLang Docs to learn more about SGLang.

The NVIDIA SGLang Container is optimized for use with NVIDIA GPUs, and contains the following software for GPU acceleration.

- Please see the CUDA section for the list of libraries inherited from CUDA container.
- NVIDIA CUDA 13.1.0.36
- SGLang 0.5.5.post2
- flashinfer 0.5.2
- transformers 4.57.1
- flash-attention 2.7.4.post1
- xgrammar 0.1.25

- torch-2.10.0a0+b4e4ee81d3

Driver Requirements

Release 25.12 is based on CUDA 13.1.0 For comprehensive and up-to-date driver compatibility information, please refer to the following documentation:

- NVIDIA CUDA Compatibility Guide - Compatibility information between CUDA versions and driver releases
- CUDA Toolkit Release Notes - Driver version requirements and compatibility matrices
- NVIDIA Drivers Download - Latest NVIDIA drivers

Key Features and Enhancements

This SGLang release includes the following key features and enhancements.

- Compatibility with CUDA 13.0
- Support for multi-node configurations.
- GB300/B300 support.
- RTX PRO™ 6000 Blackwell Server Edition support.
- DGX Spark support.
- Jetson Thor support.
- Support for 8-bit floating point (FP8) precision on Hopper GPUs and above.
- Support NVIDIA innovative 4-bit floating point NVFP4 format on Blackwell GPUs (including Jetson Thor and DGX Spark), which provides better training and inference performance with lower memory utilization.
- Supported for DeepSeek-R1, Llama-3.1-8B-Instruct.
- Support for openai/gpt-oss-20b and openai/gpt-oss-120b.

Announcements

- 25.10 is the first NVIDIA SGLang container release that brings optimizations for NVIDIA GPUs.

Known Issues

- gpt-oss family models cannot run on DGX Spark and Jetson Thor due to a OpenAI Triton issue.
- FP8 models are failing on Thor.

Chapter 9. SGLang Release 25.11

This SGLang container release is intended for use on the NVIDIA® Hopper Architecture GPU, NVIDIA H100, the NVIDIA® Ampere Architecture GPU, NVIDIA A100, and the associated NVIDIA CUDA® 12 and NVIDIA cuDNN 9 libraries.

Driver Requirements

Release 25.11 is based on CUDA 13.0.2.006 which requires NVIDIA Driver release 570 or later. However, if you are running on a data center GPU (for example, B100, L40, or any other data center GPU), you can use NVIDIA driver release 470.57 (or later R470), 525.85 (or later R525), 535.86 (or later R535), or 550.54 (or later R550) in forward-compatibility mode.

The CUDA driver's compatibility package only supports particular drivers. Thus, users should upgrade from all R418, R440, R450, R460, R510, R520, R530, R545 and R555 and R560 drivers, which are not forward-compatible with CUDA 12.8. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

Contents of the SGLang container

This container image contains the complete source of the version of SGLang in `/opt/sglang`. It is pre-built and installed in the Python default environment `/usr/local/lib/python3.12/dist-packages/sglang/` in the container image. Visit SGLang Docs to learn more about SGLang.

The NVIDIA SGLang Container is optimized for use with NVIDIA GPUs, and contains the following software for GPU acceleration.

- Please see the CUDA section for the list of libraries inherited from CUDA container.
- NVIDIA CUDA 13.0.2.006
- SGLang 0.5.3.post3
- flashinfer 0.5.0
- transformers 4.57.1
- flash-attention 2.7.4.post1
- xgrammar 0.1.25

- torch2.10.0a0+b558c986e8

Driver Requirements

Release 25.11 is based on CUDA 13.0.2 For comprehensive and up-to-date driver compatibility information, please refer to the following documentation:

- NVIDIA CUDA Compatibility Guide - Compatibility information between CUDA versions and driver releases
- CUDA Toolkit Release Notes - Driver version requirements and compatibility matrices
- NVIDIA Drivers Download - Latest NVIDIA drivers

Key Features and Enhancements

This SGLang release includes the following key features and enhancements.

- Compatibility with CUDA 13.0
- Support for multi-node configurations.
- GB300/B300 support.
- RTX PRO™ 6000 Blackwell Server Edition support.
- DGX Spark support.
- Jetson Thor support.
- Support for 8-bit floating point (FP8) precision on Hopper GPUs and above.
- Support NVIDIA innovative 4-bit floating point NVFP4 format on Blackwell GPUs (including Jetson Thor and DGX Spark), which provides better training and inference performance with lower memory utilization.
- Supported for DeepSeek-R1, Llama-3.1-8B-Instruct.
- Support for openai/gpt-oss-20b and openai/gpt-oss-120b.

Announcements

- 25.10 is the first NVIDIA SGLang container release that brings optimizations for NVIDIA GPUs.

Known Issues

- gpt-oss family models cannot run on DGX Spark and Jetson Thor due to a OpenAI Triton issue.
- FP8 models are failing on Thor.

Chapter 10. SGLang Release 25.10

This SGLang container release is intended for use on the NVIDIA® Hopper Architecture GPU, NVIDIA H100, the NVIDIA® Ampere Architecture GPU, NVIDIA A100, and the associated NVIDIA CUDA® 12 and NVIDIA cuDNN 9 libraries.

Driver Requirements

Release 25.10 is based on CUDA 13.0.2.006 which requires NVIDIA Driver release 570 or later. However, if you are running on a data center GPU (for example, B100, L40, or any other data center GPU), you can use NVIDIA driver release 470.57 (or later R470), 525.85 (or later R525), 535.86 (or later R535), or 550.54 (or later R550) in forward-compatibility mode.

The CUDA driver's compatibility package only supports particular drivers. Thus, users should upgrade from all R418, R440, R450, R460, R510, R520, R530, R545 and R555 and R560 drivers, which are not forward-compatible with CUDA 12.8. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

Contents of the SGLang container

This container image contains the complete source of the version of SGLang in `/opt/sglang`. It is pre-built and installed in the Python default environment `/usr/local/lib/python3.12/dist-packages/sglang/` in the container image. Visit SGLang Docs to learn more about SGLang.

The NVIDIA SGLang Container is optimized for use with NVIDIA GPUs, and contains the following software for GPU acceleration.

- Please refer to CUDA section for the list of libraries inherited from CUDA container.
- SGLang 0.5.3rc1
- flashinfer 0.4.0
- transformers 4.56.1
- flash-attention 2.7.4
- xgrammar 0.1.24
- NVIDIA PyTorch 25.10

Driver Requirements

Release 25.10 is based on CUDA 13.0. For comprehensive and up-to-date driver compatibility information, please refer to the following documentation:

- NVIDIA CUDA Compatibility Guide - Compatibility information between CUDA versions and driver releases
- CUDA Toolkit Release Notes - Driver version requirements and compatibility matrices
- NVIDIA Drivers Download - Latest NVIDIA drivers

Key Features and Enhancements

This SGLang release includes the following key features and enhancements.

- Compatibility with CUDA 13.0
- Support for multi-node configurations.
- GB300/B300 support.
- RTX PRO™ 6000 Blackwell Server Edition support.
- DGX Spark support.
- Jetson Thor support.
- Support for 8-bit floating point (FP8) precision on Hopper GPUs and above.
- Support NVIDIA innovative 4-bit floating point NVFP4 format on Blackwell GPUs (including Jetson Thor and DGX Spark), which provides better training and inference performance with lower memory utilization.
- Supported for DeepSeek-R1, Llama-3.1-8B-Instruct.
- Support for openai/gpt-oss-20b and openai/gpt-oss-120b.

Announcements

- 25.10 is the first NVIDIA SGLang container release that brings optimizations for NVIDIA GPUs.

Known Issues

- gpt-oss family models cannot run on DGX Spark and Jetson Thor due to a OpenAI Triton issue.
- FP8 models are failing on Thor.

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

HDMI

HDMI, the HDMI logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC.

OpenCL

OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
https://www.nvidia.com

Trademarks

NVIDIA, the NVIDIA logo, and cuBLAS, CUDA, DALI, DGX, DGX-1, DGX-2, DGX Station, DLProf, Jetson, Kepler, Maxwell, NCCL, Nsight Compute, Nsight Systems, NvCaffe, PerfWorks, Pascal, SDK Manager, Tegra, TensorRT, Triton Inference Server, Tesla, TF-TRT, and Volta are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
https://www.nvidia.com