TensorFlow For Jetson Platform

Release Notes

SWE-SWDOCTFX-001-RELN _v001 | July 2023

Table of Contents

Chapter 1. Overview

TensorFlow

TensorFlow™ is an open-source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) that flow between them. This flexible architecture lets you deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device without rewriting code.

Jetson AGX Xavier

The NVIDIA Jetson AGX Xavier developer kit for Jetson platform is the world's first AI computer for autonomous machines. The Jetson AGX Xavier delivers the performance of a GPU workstation in an embedded module under 30W.

Jetson Nano

NVIDIA Jetson Nano is a small, powerful computer for embedded AI systems and IoT that delivers the power of modern AI in a low-power platform. The Jetson Nano is targeted to get started fast with the NVIDIA Jetpack SDK and a full desktop Linux environment, and start exploring a new world of embedded products.

Jetson TX2

The Jetson TX2 Developer Kit enables a fast and easy way to develop hardware and software for the Jetson TX2 AI supercomputer on a module. It exposes the hardware capabilities and interfaces of the developer board, comes with design guides and other documentation, and is pre-flashed with a Linux development environment. The Jetson TX2 also supports NVIDIA Jetpack—a complete SDK that includes the BSP, libraries for deep learning, computer vision, GPU computing, multimedia processing, and much more.

Chapter 2. TensorFlow For Jetson Platform

This document describes the key features, software enhancements and improvements, and known issues regarding TensorFlow 2.12.0 on the Jetson platform.

Key Features and Enhancements

This release includes the following key features and enhancements.

- As of the 20.02 release, the TensorFlow package has been renamed from `tensorflow-gpu` to `tensorflow`. This change will only affect the package's installation and its name within the python package manager, not how TensorFlow is imported or accessed once installed on the system. In order to properly upgrade your version of TensorFlow, you may need to uninstall the `tensorflow-gpu` package before installing `tensorflow`, as having both installed simultaneously will lead to undefined behavior.
- TensorFlow 2.0 is now available for installation. See the TensorFlow's Effective TensorFlow 2 guide for details about the update. Note that TensorFlow 2.x is not fully compatible with TensorFlow 1.x releases, therefore, code written for the older framework may not work with the newer package.
- Bug fixes and improvements for TF-TRT. For more information, see the TensorFlow-TensorRT (TF-TRT) User Guide and the TensorFlow Container Release Notes.

Compatibility

Table 1. TensorFlow compatibility with NVIDIA containers and Jetpack

TensorFlow Version	NVIDIA TensorFlow Container	JetPack Version
2.12.0	23.06, 23.05, 23.04	5.1.x
2.11.0	23.03, 23.02, 23.01	
2.10.1	22.12	5.0.2

TensorFlow Version	NVIDIA TensorFlow Container	JetPack Version
2.10.0	22.11, 22.10	
2.9.1	22.09, 22.07	
	22.06	5.0.1
2.8.0	22.05, 22.04, 22.03	5.0
2.7.0	22.01	4.6.1
2.6.2	21.12	4.6
2.6.0	21.11, 21.09	
2.5.0	21.08, 21.07	
	21.06	4.5
2.4.0	21.05, 21.04, 21.03, 21.02	
2.3.1	20.12	
	20.12, 20.11, 20.10	4.4.x
2.3.0	20.09	
2.2.0	20.08, 20.07, 20.06	
2.1.0	20.04	
	20.03, 20.02	4.3
1.15.5	23.03, 23.02, 23.01	5.1.x
	22.12, 22.11, 22.10, 22.09, 22.07	5.0.2
	22.06	5.0.1
	22.05, 22.04, 22.03	5.0
	22.01	4.6.1
	21.12, 21.11, 21.09, 21.08, 21.07	4.6
	21.06, 21.05, 21.04, 21.03, 21.02	4.5
1.15.4	20.12	
	20.12, 20.11, 20.10	4.4.x
1.15.3	20.09, 20.08, 20.07	
1.15.2	20.06, 20.04	
	20.03, 20.02	4.3

TensorFlow Version	NVIDIA TensorFlow Container	JetPack Version
Older packages below are installed as `tensorflow-gpu`; more recent releases above as `tensorflow`.		
2.0.0	20.01, 19.12	4.3
	19.11	4.2.x
1.15.0	20.01, 19.12	4.3
	19.11	4.2.x
1.14.0	19.10, 19.09, 19.07	
1.14.0	19.09	3.3.1
1.13.1	19.05, 19.04, 19.03	4.2.x

- If you are using TensorRT with TensorFlow, ensure you are familiar with the TensorRT Container Release Notes for any known issues.

Using TensorFlow With The Jetson Platform

Memory

If you observe any out-of-memory problems in TensorFlow, you can use a custom configuration to limit the amount of memory TensorFlow tries to allocate. This can be accomplished by allowing the GPU memory allocation to grow, or setting a hard limit on the amount of memory the allocator will attempt to use. Depending on which version of the framework you are using, please see either the TensorFlow 2 guide or the archived TensorFlow 1.x documentation for details.

Storage

If you need more storage, we recommend connecting an external SSD via SATA on AGX Orin or Xavier devices, or USB on the NX series.

Operators

In TensorFlow 1.x, if you want to see which operators of your graph are placed on your module, use `tf.ConfigProto(log_device_placement=True)` to see all the device placements.

Known Issues

- Due to an issue with glibc on ARM, certain combinations of libraries may error out with a `failure to allocate memory in static TLS block`. If this occurs when trying to load TensorFlow on a Jetson device, a workaround is to set the `LD_PRELOAD` environment variable to ensure the problematic libraries are loaded with priority. For instance:

```
export LD_PRELOAD=/usr/lib/aarch64-linux-gnu/libGLdispatch.so
```

- A previous version of the TensorFlow 2.1.0 package corresponding to the 20.04 release was not compiled for the correct version of TensorFlow, and has been replaced by a fixed version. The problematic package can by identified by running `tensorflow.__version__` after importing TensorFlow. If no error is thrown, the version installed is correct; otherwise, it is recommended to upgrade to the fixed version of a newer TensorFlow release altogether.
- In TensorFlow 1.14, when using NumPy 1.17 there is a known issue where importing TensorFlow will print out a number of `FutureWarning` messages. This does not affect performance and can be avoided by installing NumPy version 1.16 instead of 1.17.
- Certain tasks with large memory requirements may fail due to the memory limitations of embedded systems. Creating a swap partition will effectively enable your device to make use of more memory at the cost of speed and storage space, and may solve the problem.
- When accelerating the inference in TensorFlow with TensorRT (TF-TRT), you may experience problems with `tf.estimator` and standard allocator (BFC allocator). It may cause crashes and nondeterministic accuracy. If you run into issues, use `cuda_malloc` as an allocator (type `exportTF_GPU_ALLOCATOR="cuda_malloc"`).
- Convolutions with a large kernel size using native TensorFlow are slow. It's because the `SwapDimension0And2InTensor3Simple` kernel has no implementation using tiling. This issue will be resolved in a future release. Use TF-TRT to speed up the process.
- The TF-TRT conversion sometimes crashes without any error message. It usually happens when the OS runs out of swap memory. In this case, it can kill processes to reclaim space. If you encounter this issue, add more swap memory.
- Some NVIDIA tools, for example, `nvidia-smi`, are not supported on the Jetson platform. For more information, see the JetPack documentation.
- Jetson AGX Xavier has 16GB of memory which is shared between the CPU and GPU. As a result, some applications that work on a GPU with 16GB of memory may not work on Jetson AGX Xavier because not all of the 16GB is available to the GPU.
- The following TensorFlow unit tests will fail on Jetson AGX Xavier. Some of the TensorFlow failures are due to bugs in the test scripts; which affect only the AArch64 platform.
  - `//tensorflow/core/kernels:sparse_matmul_op_test_gpu`
  - `//tensorflow/core/kernels:requantize_op_test`
  - `//tensorflow/core/kernels:quantized_bias_add_op_test`
  - `//tensorflow/core:util_tensor_slice_set_test`
  - `//tensorflow/core:platform_stacktrace_handler_test`
  - `//tensorflow/core:platform_profile_utils_cpu_utils_test`
  - `//tensorflow/core:common_runtime_gpu_gpu_device_test`
  - `//tensorflow/core:gpu_device_unified_memory_test_gpu`
  - `//tensorflow/core:common_runtime_direct_session_with_debug_test`

- `//tensorflow/core:common_runtime_direct_session_with_debug_test`
- `//tensorflow/core:device_tracer_test`
- `//tensorflow/core:framework_unique_tensor_references_test`
- `//tensorflow/python:basic_session_run_hooks_test`

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