NVIDIA TensorRT

Operator's Reference

DA-11116-001 _v8.5.2 | December 2022

Table of Contents

Chapter 1. Layers and Features

The section lists the supported TensorRT layers and each of the features.

Note:

- **Supports broadcast** indicates support for broadcast in this layer. This layer allows its two input tensors to be of dimensions [1, 5, 4, 3] and [1, 5, 1, 1], and its output is [1, 5, 4, 3]. The second input tensor has been broadcast in the innermost two dimensions.
- **Supports broadcast across batch** indicates support for broadcast across the batch dimension. "NA" in this column means it is not allowed in networks with an implicit batch dimension.

Table 1. List of Supported Features per TensorRT Layer

Layer	Dimensions of input tensor	Dimensions of output tensor	Does the operation apply to only the innermost 3 dimensions?	Supports broadcast	Supports broadcast across batch
IActivationLayer	0-7 dimensions	0-7 dimensions	No	No	No
IAssertionLayer	0-1 dimensions	No output	No	No	No
IConcatenationLayer	1-7 dimensions	1-7 dimensions	No	No	No
IConstantLayer	Has no inputs	0-7 dimensions	No	No	Always
IConvolutionLayer > 2D Convolution	Three or more dimensions	Three or more dimensions	Yes	No	No
IConvolutionLayer > 3D Convolution	Four or more dimensions	Four or more dimensions	No	No	No

Layer	Dimensions of input tensor	Dimensions of output tensor	Does the operation apply to only the innermost 3 dimensions?	Supports broadcast	Supports broadcast across batch
IDeconvolutionLayer > 2D Deconvolution	Three or more dimensions	Three or more dimensions	Yes	No	No
IDeconvolutionLayer > 3D Deconvolution	Four or more dimensions	Four or more dimensions	No	No	No
IDequantizeLayer	Two or more dimensions	Two or more dimensions	Yes	No	No
IEinsumLayer	0-7 dimensions	0-7 dimensions	No	No	Yes
IElementWiseLayer	0-7 dimensions	0-7 dimensions	No	Yes	Yes
IFillLayer	One dimension	0-7 dimensions	No	Not Applicable	Not Applicable
IFullyConnectedLayer	Three or more dimensions	Three or more dimensions	Yes	No	No
IGatherLayer	▶ Input1: 1-7 dimensions ▶ Input2: 0-7 dimensions	0-7 dimensions	No	No	Yes
IIdentityLayer	0-7 dimensions	0-7 dimensions	No	No	No
ILRNLayer	Three or more dimensions	Three or more dimensions	Yes	No	No
IMatrixMultiplyLayer	Two or more dimensions	Two or more dimensions	No	Yes	Yes
IPaddingLayer	Three or more dimensions	Three or more dimensions	Yes	No	No
IParametricReluLayer	1-7 dimensions	1-7 dimensions	No	No	No
IPluginV2Layer	User defined	User defined	User defined	User defined	User defined
IPoolingLayer > 2D Pooling	Three or more dimensions	Three or more dimensions	Yes	Yes	Yes
IPoolingLayer > 3D Pooling	Four or more dimensions	Four or more dimensions	No	Yes	Yes

Layer	Dimensions of input tensor	Dimensions of output tensor	Does the operation apply to only the innermost 3 dimensions?	Supports broadcast	Supports broadcast across batch
IQuantizeLayer	Two or more dimensions	Two or more dimensions	Yes	No	No
IRaggedSoftMaxLayer	▶ Input: Two dimensions ▶ Bounds: Two dimensions	Two or more dimensions	No	No	Yes
IReduceLayer	1-7 dimensions	0-7 dimensions	No	No	No
IResizeLayer	1-7 dimensions	1-7 dimensions	No	No	No
IRNNLayer	▶ Data/ Hidden/ Cell: Two or more dimensions ▶ Seqlen: Zero or more dimensions	Data/Hidden/ Cell: Two or more dimensions	No	No	No
IScaleLayer	Three or more dimensions	Three or more dimensions	Yes	No	No
IScatterLayer	0-7 dimensions	0-7 dimensions	No	No	No
ISelectLayer	0-7 dimensions	0-7 dimensions	No	Yes	Not Applicable
IShapeLayer	One or more dimensions	One dimension	No	No	Not Applicable
IShuffleLayer	0-7 dimensions	0-7 dimensions	No	No	No
ISliceLayer	1-7 dimensions	1-7 dimensions	No	No	Yes
ISoftMaxLayer	1-7 dimensions	1-7 dimensions	No	No	Yes
ITopKLayer	1-7 dimensions	▶ Output1: 1-7 dimensions	Yes	No	Yes

Layer	Dimensions of input tensor	Dimensions of output tensor	Does the operation apply to only the innermost 3 dimensions?	Supports broadcast	Supports broadcast across batch
		▶ Output2: 1-7 dimensions			
IUnaryLayer	1-7 dimensions	1-7 dimensions	No	No	No

Chapter 2. Layers and Precision

The section lists the TensorRT layers and the precision modes that each layer supports. It also lists the ability of the layer to run on Deep Learning Accelerator (DLA).

For more information about additional constraints, see DLA Supported Layers.

Table 2. List of Supported Precision Modes per TensorRT Layer

Layer	FP32	FP16	INT8	INT32	Bool	DLA FP16	DLA INT8
IActivationLayer	Yes	Yes	Yes	No	No	Yes[1]	Yes[2]
IAssertionLayer	No	No	No	No	Yes	No	No
IConcatenationLayer	Yes	Yes	Yes	Yes	Yes	Yes[3]	Yes[5]
IConstantLayer	Yes	Yes	Yes	Yes	Yes	No	No
IConvolutionLayer > 2D Convolution	Yes	Yes	Yes	No	No	Yes	Yes
IConvolutionLayer > 3D Convolution	Yes	Yes	Yes	No	No	No	No
IDeconvolutionLayer > 2D Deconvolution	Yes	Yes	Yes	No	No	Yes	Yes[4]
IDeconvolutionLayer > 3D Deconvolution	Yes	Yes	No	No	No	No	No
IDequantizeLayer	No	No	Yes	No	No	No	No
IEinsumLayer	Yes	Yes	No	No	No	No	No

[1] Partial support. *Yes* for `ReLU`, `Clipped ReLU`, `Leaky ReLU`, `Sigmoid`, and `TanH` activation types only.
[2] Partial support. *Yes* for `ReLU`, `Clipped ReLU`, `Leaky ReLU`, `Sigmoid`, and `TanH` activation types only.
[3] Partial support. *Yes* for concatenation across `C` dimension only.
[4] Partial support. *Yes* for ungrouped deconvolutions and *No* for grouped.

Layer	FP32	FP16	INT8	INT32	Bool	DLA FP16	DLA INT8
IElementWiseLayer	Yes	Yes	Yes	Yes	Yes	Yes[5]	Yes[6]
IFillLayer	Yes	No	No	Yes	No	No	No
IFullyConnectedLayer	Yes	Yes	Yes	No	No	Yes	Yes
IGatherLayer	Yes	Yes	No	Yes	Yes	No	No
IIdentityLayer	Yes	Yes	Yes	Yes	No	No	No
ILRNLayer	Yes	Yes	Yes	No	No	Yes	No
IMatrixMultiplyLayer	Yes	Yes	Yes[7]	No	No	No	No
IPaddingLayer	Yes	Yes	Yes	No	No	No	No
IParametricReLULayer	Yes	Yes	Yes	No	No	Yes	Yes
IPluginV2Layer	Yes	Yes	Yes	No	No	No	No
IPoolingLayer > 2D Pooling	Yes	Yes	Yes	No	No	Yes[8]	Yes[9]
IPoolingLayer > 3D Pooling	Yes	Yes	No	No	No	No	No
IQuantizeLayer	Yes	No	No	No	No	No	No
IRaggedSoftMaxLayer	Yes	No	No	No	No	No	No
IReduceLayer	Yes	Yes	Yes	Yes	No	No	No
IResizeLayer	Yes	Yes	Yes	No	No	Yes	Yes
IRNNLayer	Yes	Yes	No	No	No	No	No
IScaleLayer	Yes	Yes	Yes	No	No	Yes[9]	Yes[10]
IScatterLayer	Yes	Yes	Yes	Yes	Yes	No	No
ISelectLayer	Yes	Yes	No	Yes	Yes	No	No
IShapeLayer[10]	Yes	Yes	Yes	Yes	Yes	No	No
IShuffleLayer	Yes	Yes	Yes	Yes	Yes	Yes[11]	Yes[12]
ISliceLayer	Yes	Yes	No[13]	Yes	Yes	Yes	No

[5] Partial support. *Yes* for `sum`, `sub`, `prod`, `min` , and `max` elementwise operations only.
[6] Partial support. *Yes* for `sum`, `sub`, `prod`, `min` , and `max` elementwise operations only.
[7] Partial support. *Yes* for the case the second input is build-time constant and the first input is not transposed - either produced by a Shuffle layer or `opA == kTRANSPOSE`.
[8] Partial support. *Yes* for `max` and average padding inclusive pooling type only.
[9] Partial support. DLA does not support power on the scale layer.
[10] Output is always INT32.
[11] Partial support in TensorRT 8.4.12 only.
[12] Partial support in TensorRT 8.4.12 only.
[13] Partial support. *Yes* for unstrided Slice and *No* for strided.

Layer	FP32	FP16	INT8	INT32	Bool	DLA FP16	DLA INT8
ISoftMaxLayer	Yes	Yes	No	No	No	Yes	No
ITopKLayer	Yes	Yes	No	No	No	No	No
IUnaryLayer[14]	Yes	Yes	Yes	Yes	Yes	No	No

Note: DLA with FP16/INT8 precision with some restrictions on layer parameters.

[14] Datatype support is limited to the type of unary operation used.

Chapter 3. Layers for Flow-Control Constructs

The following table lists the TensorRT layers that can be used as interior layers in TensorRT flow-control constructs.

Currently, TensorRT supports loop constructs (using `ILoopLayer`) and ternary conditional constructs (using `IIfConditionalLayer`). Interior layers are layers that include the body of a loop or one of the two branches of an if-conditional.

An `ILoopLayer` interior layer may contain other loops and if-conditionals. An `IIfConditionalLayer` branch may contain other if-conditionals and loops.

Flow-control constructs do not support INT8 calibration and interior-layers cannot employ implicit-quantization (INT8 is supported only in explicit-quantization mode).

Table 3. List of TensorRT Layers that are Supported as Interior Layers of Flow-control Constructs

Layer	Supported
IActivationLayer	Yes, when the operation is one of: `kRELU`, `kSIGMOID`, `kTANH`, `kELU`
IAssertionLayer	Yes
IConcatenationLayer	Yes
IConstantLayer	Yes
IConvolutionLayer > 2D Convolution	singleton channel and spatial dims, that are, the dimensions must be static or have a single value in each optimization profile
IConvolutionLayer > 3D Convolution	singleton channel and spatial dims
IDeconvolutionLayer > 2D Deconvolution	No
IDeconvolutionLayer > 3D Deconvolution	No
IDequantizeLayer	No
IEinsumLayer	Yes

Layer	Supported
IElementWiseLayer	Yes
IFillLayer	`kRANDOM_UNIFORM` only
IFullyConnectedLayer	Yes
IGatherLayer	Yes
IIdentityLayer	Yes
ILRNLayer	No
IMatrixMultiplyLayer	Yes
IPaddingLayer	No
IParametricReluLayer	No
IPluginV2Layer	Yes
IPoolingLayer > 2D Pooling	No
IPoolingLayer > 3D Pooling	No
IQuantizeLayer	No
IRaggedSoftMaxLayer	No
IReduceLayer	Yes
IResizeLayer	No
IRNNLayer	No
IScaleLayer	Yes
IScatterLayer	Yes
ISelectLayer	Yes
IShapeLayer	Yes
IShuffleLayer	Yes
ISliceLayer	Yes
ISoftMaxLayer	Yes
ITopKLayer	No
IUnaryLayer	Yes, when the operation is one of: `kABS`, `kCEIL`, `kERF`, `kEXP`, `kFLOOR`, `kLOG`, `kNEG`, `kNOT`, `kRECIP`, `kROUND`, `kSIGN`, `kSQRT`, `kSIN`, `kCOS`, `kATAN`

Chapter 4. Operators

To view the operators, refer to the TensorRT Operators.

TensorRT can optimize performance by fusing layers. For information about how to enable layer fusion optimizations, refer to Types of Fusions. For information about optimizing individual layer performance, refer to Optimizing Layer Performance.

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.

VESA DisplayPort

DisplayPort and DisplayPort Compliance Logo, DisplayPort Compliance Logo for Dual-mode Sources and DisplayPort Compliance Logo for Active Cables are trademarks owned by the Video Electronics Standards Associations in the United States and other countries.

HDMI

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

Arm

Arm, AMBA and Arm Powered are registered trademarks of Arm Limited. Cortex, MPCore and Mali are trademarks of Arm Limited. "Arm" is used to represent Arm Holdings plc; its operating company Arm Limited; and the regional subsidiaries Arm Inc.; Arm KK; Arm Korea Limited.; Arm Taiwan Limited; Arm France SAS; Arm Consulting (Shanghai) Co. Ltd.; Arm Germany GmbH; Arm Embedded Technologies Pvt. Ltd.; Arm Norway, AS and Arm Sweden AB.

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
http://www.nvidia.com

Trademarks

NVIDIA, the NVIDIA logo, and BlueField, CUDA, DALI, DRIVE, Hopper, JetPack, Jetson AGX Xavier, Jetson Nano, Kepler, Maxwell, NGC, Nsight, Orin, Pascal, Quadro, Tegra, TensorRT, Triton, Turing and Volta are trademarks and/or registered trademarks of NVIDIA Corporation in the United States 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
http://www.nvidia.com