Installing TensorFlow For Jetson Platform

Installation Guide

SWE-SWDOCTFX-001-INST _v001 | October 2020

Table of Contents

Chapter 1. Overview....1
1.1. Benefits of TensorFlow on Jetson Platform....1
Chapter 2. Prerequisites and Dependencies....3
Chapter 3. Installing TensorFlow....4
3.1. Installing Multiple TensorFlow Versions....5
3.2. Upgrading TensorFlow....6
Chapter 4. Verifying The Installation....7
Chapter 5. Best Practices....8
Chapter 6. Uninstalling....9
Chapter 7. Troubleshooting....10
Chapter 8. Support....11

Chapter 1. Overview

TensorFlow on Jetson Platform

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.

1.1. Benefits of TensorFlow on Jetson Platform

Installing TensorFlow for Jetson Platform provides you with the access to the latest version of the framework on a lightweight, mobile platform without being restricted to TensorFlow Lite.

Chapter 2. Prerequisites and Dependencies

Before you install TensorFlow for Jetson, ensure you:

1. Install JetPack on your Jetson device.
2. Install system packages required by TensorFlow:

```
$ sudo apt-get update
$ sudo apt-get install libhdf5-serial-dev hdf5-tools libhdf5-dev zlib1g-dev zip libjpeg8-dev liblapack-dev libblas-dev gfortran
```

3. Install and upgrade `pip3`.

```
$ sudo apt-get install python3-pip
$ sudo pip3 install -U pip testresources setuptools=49.6.0
```

4. Install the Python package dependencies.

```
$ sudo pip3 install -U numpy==1.16.1 future==0.18.2 mock==3.0.5 h5py==2.10.0
 keras_preprocessing==1.1.1 keras_applications==1.0.8 gast==0.2.2 futures protobuf
 pybind11
```

Refer to the TensorFlow For Jetson Platform Release Notes for information about the Python package versions used for the most recent release.

Chapter 3. Installing TensorFlow

Note: As of the 20.02 TensorFlow release, the package name has changed from `tensorflow-gpu` to `tensorflow`. See the section on Upgrading TensorFlow for more information.

Install TensorFlow using the `pip3` command. This command will install the latest version of TensorFlow compatible with JetPack 4.4.

```
$ sudo pip3 install --pre --extra-index-url https://developer.download.nvidia.com/compute/redist/jp/v44 tensorflow
```

Note: TensorFlow version 2 was recently released and is not fully backward compatible with TensorFlow 1.x. If you would prefer to use a TensorFlow 1.x package, it can be installed by specifying the TensorFlow version to be less than 2, as in the following command:

```
$ sudo pip3 install --pre --extra-index-url https://developer.download.nvidia.com/compute/redist/jp/v44 'tensorflow<2'
```

If you want to install the latest version of TensorFlow supported by a particular version of JetPack, issue the following command:

```
$ sudo pip3 install --extra-index-url https://developer.download.nvidia.com/compute/redist/jp/v$JP_VERSION tensorflow
```

Where:

JP_VERSION
The major and minor version of JetPack you are using, such as `42` for JetPack 4.2.2 or `33` for JetPack 3.3.1.

If you want to install a specific version of TensorFlow, issue the following command:

```
$ sudo pip3 install --extra-index-url https://developer.download.nvidia.com/compute/redist/jp/v$JP_VERSION tensorflow==$TF_VERSION+nv$NV_VERSION
```

Where:

JP_VERSION
The major and minor version of JetPack you are using, such as `42` for JetPack 4.2.2 or `33` for JetPack 3.3.1.

TF_VERSION
The released version of TensorFlow, for example, `1.13.1`.

NV_VERSION
The monthly NVIDIA container version of TensorFlow, for example, `19.01`.

Note: The version of TensorFlow you are trying to install must be supported by the version of JetPack you are using. Also, the package name may be different for older releases. See the TensorFlow For Jetson Platform Release Notes for a list of some recent TensorFlow

releases with their corresponding package names, as well as NVIDIA container and JetPack compatibility.

For example, to install TensorFlow 1.13.1 as of the 19.03 release, the command would look similar to the following:

```
$ sudo pip3 install --extra-index-url https://developer.download.nvidia.com/compute/redist/jp/v42 tensorflow-gpu==1.13.1+nv19.3
```

3.1. Installing Multiple TensorFlow Versions

If you want to have multiple versions of TensorFlow available at the same time, this can be accomplished using virtual environments. See below.

Set up the Virtual Environment

First, install the `virtualenv` package and create a new Python 3 virtual environment:

```
$ sudo apt-get install virtualenv
$ python3 -m virtualenv -p python3 <chosen_venv_name>
```

Activate the Virtual Environment

Next, activate the virtual environment:

```
$ source <chosen_venv_name>/bin/activate
```

Install the desired version of TensorFlow and its dependencies:

```
$ pip3 install -U numpy grpcio absl-py py-cpuinfo psutil portpicker six mock requests gast
 h5py astor termcolor protobuf keras-applications keras-preprocessing wrapt google-pasta
 setuptools testresources
$ pip3 install --extra-index-url https://developer.download.nvidia.com/compute/redist/jp/v44
 tensorflow==$TF_VERSION+nv$NV_VERSION
```

Deactivate the Virtual Environment

Finally, deactivate the virtual environment:

```
$ deactivate
```

Run a Specific Version of TensorFlow

After the virtual environment has been set up, simply activate it to have access to the specific version of TensorFlow. Make sure to deactivate the environment after use:

```
$ source <chosen_venv_name>/bin/activate
$ <Run the desired TensorFlow scripts>
$ deactivate
```

3.2. Upgrading TensorFlow

Note: Due to the recent package renaming, if the TensorFlow version you currently have installed is older than the 20.02 release, you must uninstall it before upgrading to avoid conflicts. See the section on Uninstalling TensorFlow for more information.

To upgrade to a more recent release of TensorFlow, if one is available, run the install command with the 'upgrade' flag:

```
$ sudo pip3 install --upgrade --extra-index-url https://
developer.download.nvidia.com/compute/redist/jp/v$44 tensorflow
```

Chapter 4. Verifying The Installation

About this task

To verify that TensorFlow has been successfully installed on Jetson AGX Xavier, you'll need to launch a Python prompt and import TensorFlow.

Procedure

1. From the terminal, run:

   ```
   $ python3
   ```

2. Import TensorFlow:

   ```
   >>> import tensorflow
   ```

   If TensorFlow was installed correctly, this command should execute without error.

Chapter 5. Best Practices

Performance model

It is recommended to choose the right performance mode to get the best possible performance given energy usage limitations. There is a command line tool (`nvpmodel`) that can be used to change the performance mode. In order to check the current performance mode, issue:

```
$ sudo nvpmodel -q --verbose
```

To change the mode to MAX-N, issue:

```
$ sudo nvpmodel -m 0
```

For more information, see:

- How do you switch between max-q and max-p?
- Jetson/Performance
- Two cores disabled

Swap space on Jetson Xavier

On Jetson Xavier, certain applications could run out of memory (16GB shared between CPU and GPU). This problem can be resolved by creating a swap partition on the external memory. Typically 4GB of swap space is enough.

Chapter 6. Uninstalling

TensorFlow can easily be uninstalled using the `pip3 uninstall` command, as below:

```
$ sudo pip3 uninstall -y tensorflow
```

Note: If you are If you are using a version of TensorFlow older than the 20.02 release, the package name is `tensorflow-gpu`, and you will need to run the following command to uninstall TensorFlow instead. See the TensorFlow For Jetson Platform Release Notes for more information.

```
$ sudo pip3 uninstall -y tensorflow-gpu
```

Chapter 7. Troubleshooting

Join the NVIDIA Jetson and Embedded Systems community to discuss Jetson Platform-specific issues.

Chapter 8. Support

TensorFlow

For more information about TensorFlow, see:

- TensorFlow tutorials
- TensorFlow API
- Install TensorFlow on Ubuntu
- NVIDIA TensorFlow documentation

Jetson Platform

For more information about Jetson Platform, see:

- NVIDIA Jetson AGX Xavier Developer Kit
- NVIDIA Jetson Nano Developer Kit
- Jetson software documentation

NVIDIA SDK Manager

- See NVIDIA SDK Manager for more information.

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 Association 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.

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 cuBLAS, CUDA, cuDNN, DALI, DIGITS, DGX, DGX-1, DGX-2, DGX Station, DLProf, Jetson, Kepler, Maxwell, NCCL, Nsight Compute, Nsight Systems, NvCaffe, NVIDIA Ampere GPU Architecture, 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

© 2018-2020 NVIDIA Corporation. All rights reserved.

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