DALI

DU-09049-001 _v0.4.1 Beta Release | November 2018

Quick Start Guide

TABLE OF CONTENTS

Chapter 1. OVERVIEW

Today's deep learning applications include complex, multi-stage pre-processing data pipelines that include compute-intensive steps mainly carried out on the CPU. For instance, steps such as load data from disk, decode, crop, random resize, color and spatial augmentations and format conversions are carried out on the CPUs, limiting the performance and scalability of training and inference tasks. In addition, the deep learning frameworks today have multiple data pre-processing implementations, resulting in challenges such as portability of training and inference workflows and code maintainability.

NVIDIA® Data Loading Library™ (DALI) is a collection of highly optimized building blocks and an execution engine to accelerate input data pre-processing for deep learning applications. DALI provides both performance and flexibility of accelerating different data pipelines, as a single library, that can be easily integrated into different deep learning training and inference applications.

Key highlights of DALI include:

- Full data pipeline accelerated from reading disk to getting ready for training/ inference
- Flexibility through configurable graphs and custom operators
- Support for image classification and segmentation workloads
- Ease of integration through direct framework plugins and open source bindings
- Portable training workflows with multiple input formats - JPEG, PNG (fallback to CPU), raw formats, LMDB, RecordIO, TFRecord
- Extensible for user specific needs through open source license

Chapter 2. DALI AND NGC

DALI is pre-installed in the NVIDIA GPU Cloud TensorFlow, PyTorch, and MXNet containers in version 18.07 and later.

Chapter 3. INSTALLING DALI

DALI can be installed either directly using a pre-built binary or by compiling the sources from GitHub.

3.1. Installing Prebuilt DALI Packages

3.1.1. Prerequisites

Ensure you meet the following minimum requirements:

- Linux x64
- NVIDIA Driver (384.xx or later driver releases) supporting CUDA 9.0 or later
- One or more of the following deep learning frameworks:
  MXNet 1.3 or later
  Version 1.3 from the Python package with the following command:

  ```
  pip install mxnet-cu90==1.3.0
  ```

  PyTorch 0.4
  TensorFlow 1.7 or later

3.1.2. Binary Installation

Install DALI using **pip**.

```
pip install --extra-index-url
https://developer.download.nvidia.com/compute/redist nvidia-dali
```

3.2. Compiling DALI From Source

3.2.1. Prerequisites

Ensure you meet the following minimum requirements:

- Linux x64
- GCC 4.9.2 or later
- NVIDIA CUDA 9.0 (CUDA 8.0 compatibility is provided *unofficially*[1])
- nvJPEG library (This can be *unofficially* disabled[1])
- protobuf version 2 or later (version 3 or later is required for TensorFlow TFRecord file format support)
- CMake 3.5 or later
- libjpeg-turbo 1.5.x or later (This can be *unofficially* disabled[1])
- OpenCV 3 or later (OpenCV 2.x compatibility is provided *unofficially*[1])
- liblmdb 0.9.x or later
- One or more of the following deep learning frameworks:
  MXNet 1.3 or later
  Version 1.3 from the Python package with the following command:

```
pip install mxnet-cu90==1.3.0
```

  PyTorch 0.4
  TensorFlow 1.7 or later

TensorFlow installation is required to build the TensorFlow plugin for DALI.

3.2.2. GitHub Installation

1. Download the DALI source package from GitHub.

```
git clone --recursive https://github.com/NVIDIA/dali
cd dali
```

2. Create the build directory.

```
mkdir build
cd build
```

3. Compile DALI.
   a) To build DALI without LMDB support, issue the following command:

```
cmake ..
make -j"$(nproc)"
```

[1] Items marked *unofficial* are community contributions that are believed to work but not officially tested or maintained by NVIDIA.

b) To build DALI with LMDB support, issue the following command:

```
cmake -DBUILD_LMDB=ON ..
make -j"$(nproc)"
```

c) To build DALI using Clang, issue the following command:

Caution This build is experimental, meaning it is not maintained and tested like the default configuration, therefore, it's not guaranteed to work. We recommend using GCC for production builds.

```
cmake -DCMAKE_CXX_COMPILER=clang++ -DCMAKE_C_COMPILER=clang  ..
make -j"$(nproc)"
```

3.2.2.1. CMake Build Parameters

You can use the following optional CMake build parameters when configuring DALI:

BUILD_PYTHON
Use this parameter to build Python bindings. The default is **ON**.

BUILD_TEST
Use this parameter to include building the test suite. The default is **ON**.

BUILD_BENCHMARK
Use this parameter to include building benchmarks. The default is **ON**.

BUILD_LMDB
Use this parameter to build with support for LMDB. The default is **OFF**.

BUILD_NVTX
Use this parameter to build with NVTX profiling enabled. The default is **OFF**.

BUILD_TENSORFLOW
Use this parameter to build the TensorFlow plugin. The default is **OFF**.

BUILD_JPEG_TURBO*(unofficial)*
Use this parameter to build with libjpeg-turbo. The default is **ON**.[2]

BUILD_NVJPEG *(unofficial)*
Use this parameter to build with nvJPEG. The default is **ON**.[3]

3.2.3. Installing Python Bindings

Issue the **`pip install dali/python`** command to install Python bindings.

[2] Items marked *unofficial* are community contributions that are believed to work but not officially tested or maintained by NVIDIA.
[3] Items marked *unofficial* are community contributions that are believed to work but not officially tested or maintained by NVIDIA.

Chapter 4.
EXECUTING RESNET-50 INPUT PIPELINE

After you've installed DALI, you can run a pre-configured, ResNet-50 model accelerated by DALI, on MXNet, PyTorch, and TensorFlow frameworks for image classification training. Each of the following samples offload image loading and augmentation operations onto GPUs.

You can use Python toolchain from the command shell or Jupyter notebook to start the ResNet-50 training session.

The DALI integrated ResNet-50 Python samples are located:

- MXNet
- PyTorch
- TensorFlow

Chapter 5. UNINSTALLING DALI

Uninstall DALI.

```
pip uninstall -y nvidia-dali
```

Notice

Trademarks

Copyright