**Building a custom (keras installation) NVidia container using singularity**

**Author:** Asa Thibodeau

**Objectives:**

* Learn how to set up a singularity “sandbox” from an existing image file such as docker.
* Learn how to install python packages (i.e., keras) within a sandbox
* Learn how to convert a sandbox to a “Singularity Image Format” (.sif) file
* Run example code from keras

**1) Building a singularity sandbox from an existing image**

To begin, we will build a sandbox from an existing container from NVidia. Rather than pulling the docker image into a “Singularity Image Format” (.sif) file, we want to build sandbox that we can manipulate without having root access.

First, log into an interactive session and load the singularity module:

|  |
| --- |
| srun --mem=4G --gres gpu:1 --pty bash  module load singularity |

Now build the sandbox directory from an NVidia docker image () using the following:

|  |
| --- |
| singularity build --sandbox ~/keras-tensorflow/ docker://nvcr.io/nvidia/tensorflow:19.01-py3 |

A new directory representing the docker image will be created (keras-tensorflow) in your home directory:

|  |
| --- |
| [user@winter014]$ ls  keras-tensorflow |

**2) Using the sandbox shell and installing additional python packages**

Now that we have built a sandbox directory containing all of the files required to run the image, we can now use shell to go into this directory (the image container) with the following command:

|  |
| --- |
| singularity shell --writable keras-tensorflow/ |

Building the image using a sandbox directory allows us to customize it without having root access. We can now begin installing additional python packages using pip, which will install these libraries for the python installation within the sandbox directory. In this example we will install keras:

|  |
| --- |
| pip install keras |

With keras installed, the last step that we need to get our customized image running is to convert the sandbox directory to “Singularity Image Format” (.sif).

**3) Converting a sandbox directory to a SIF file**

The sandbox should now be ready to run any python code that depends on keras. If you haven’t done so already, exit the sandbox shell:

|  |
| --- |
| exit |

Now, we will build a .sif from the sandbox:

|  |
| --- |
| singularity build keras-tensorflow.sif keras-tensorflow/ |

You should now see “.sif” within your current directory:

|  |
| --- |
| [user@winter014]$ ls  keras-tensorflow keras-tensorflow.sif |

**3) Running example code from keras**

To test that our customized image can run keras, we will download the keras source code and run an example script. In this case, it will be the MNIST convolutional neural network example.

Download the keras source code by cloning the git repository:

|  |
| --- |
| git clone https://github.com/keras-team/keras.git |

Now run python on the mnist\_cnn.py using our custom image:

|  |
| --- |
| singularity run --nv keras-tensorflow.sif python keras/examples/mnist\_cnn.py |

You should now see the example script running. The model shown in the example is training over 12 epochs which should complete within ~3-4 seconds for each round:

