

# How to install PyTorch v4.0+ on Raspberry Pi-3B+, Odroids, and other ARM-based devices



Amrit Das

Follow

Dec 12, 2018 · 3 min read

Installing Neural Network Frame Works on ‘ARM’ or ‘aarch’ based architectures are a real pain. Most of us reading here would have tried hell lot of commands and flags to pass through few errors. Neural Network has taken the over the world by amazement while aiding in various fields that were not even possible a couple years back. Now that we are comfortable with using these frameworks, we look forward to deploying these frameworks in mobile devices for various projects. By far Caffé is the only framework that easily accommodates itself with ARM devices. Tensorflow Lite is a work on progress.

I find working on PyTorch environment comfortable due to several reasons, like it’s completely python-based, the simplicity of usage and its dynamic approach to graph computation. Now installing PyTorch in a 64 bit PC is a piece of cake implementing the same on an arm-based/32-bit architecture is ‘Welcome To The Hell!’

## Step by Step Procedures on How to Install PyTorch from Source —

**Pre-Installation notes:** It’s a personal recommendation to use a 16 GB or 32 GB SD card. You can either go for the raspbian image or Ubuntu-mate (if at all GUI is necessary to you). Once you are done with Installation, update and upgrade from the terminal and install Vim ( or use nano/gedit or any other text editor)

```
sudo apt update && sudo apt upgrade
```

```
sudo apt install vim
```

**Add extra SWAP memory:** If you don't want to get 'Memory exhausted' or 'Cannot Allocate Memory', increase you swap memory.

```
sudo dd if=/dev/zero of=/swap1 bs=1M count=2048
sudo mkswap /swap1
sudo vim /etc/fstab
```

This will open up your fstab file. You may or may not have a swap file. If you see something like,

```
/swap0 swap swap
```

You already had a swap so replace that line with the following( if the line is absent add the following line to your fstab file)

```
/swap1 swap swap
```

Reboot your machine, to recognize the new swap space

**Install prerequisite packages:**

```
sudo apt install libopenblas-dev libblas-dev m4 cmake cython
python3-dev python3-yaml python3-setuptools
```

You can also try with python 2.7 but I am not sure there won't be any errors. The PyTorch developers call for python3.

**Get the PyTorch from their official Github Repo:**

```
mkdir pytorch_install && cd pytorch_install
git clone --recursive https://github.com/pytorch/pytorch
cd pytorch
```

## Setup Environment Variables:

```
export NO_CUDA=1
export NO_DISTRIBUTED=1
export NO_MKLDNN=1
export NO_NNPACK=1
export NO_QNNPACK=1
```

Most of the dev boards won't have an NVIDIA GPU and nor does the Pi. So the `no_cuda` flag will make sure that the compiler doesn't look for cuda files. Similarly, `mkldnn` is not available for 32-bit machines and it also shows error in 64-bit machines too. So it's better turned OFF. In any case, if you reboot your machine or change terminal, add the environment variables again.

**Build:** Here you start building the process and I would suggest having patience as this is going to cost you more than 2 hours in a Raspberry Pi 3B+. You might put this on and go for a shopping or probably watch some PyTorch Tutorials.

```
python3 setup.py build
```

You should not encounter error but if you encounter any error like below,

```
Failed to run 'bash tools/build_pytorch_libs.sh --use-cuda --use-
nnpack --use mkldnn --use qnnpack caffe2'
```

Probably you have not set the environment variables correctly so just 'make clean' follow the environment variable setup part again and build.

Once you have finished Building without any error,

```
sudo -E python3 setup.py install
```

You shouldn't skip the '-E' as this flags ensure to force use the environment variables.

If everything goes right, you should be able to use 'PyTorch' without any errors. Also, the PyTorch folder has its own torch folder inside the PyTorch folder so if you try to 'import torch' while inside the folder, you will encounter errors. So in order to test your installation,

```
cd
python3
>>> from __future__ import print_function
>>> import torch
>>> a = torch.rand(5,3)
>>> print (a)
```

If you get an output similar to

```
tensor([[0.3380, 0.3845, 0.3217],
        [0.8337, 0.9050, 0.2650],
        [0.2979, 0.7141, 0.9069],
        [0.1449, 0.1132, 0.1375],
        [0.4675, 0.3947, 0.1426]])
```

Yes you have successfully installed PyTorch in your dev-board!

I am currently a 3rd Year ECE student from SRM Institute of Science and Technology and a member of [SRM Team Humanoid](#) where I work as a Computer Vision Engineer. You can connect to me at my [page](#) or look of more what I do at my [Git Repo](#).

[Raspberry Pi](#) [Pytorch](#) [Odroid](#) [Deep Learning](#) [Installation](#)

[About](#) [Help](#) [Legal](#)

Get the Medium app

