

How to Install PyTorch on Ubuntu 22.04

Learn how to install PyTorch on Ubuntu 22.04 with step-by-step instructions for both CPU and GPU versions, including pip and conda installation methods.

Contents

01	Introduction	3
02	Test the PyTorch Installation	6
03	Conclusion	7

Introduction

PyTorch is a flexible and high-performance machine learning framework that includes a rich set of libraries for building and training deep learning models. It integrates smoothly with Python and supports fast, dynamic computation, making it a top choice for both research and production. Running PyTorch on Ubuntu 22.04, a stable and developer-friendly Linux distribution, offers an ideal environment for AI and machine learning workloads thanks to its strong support for modern software and hardware acceleration.

This guide walks you through the steps to install PyTorch on Ubuntu 22.04 Server, a stable and widely-used Linux distribution ideal for machine learning development and deployment. For containerized environments, you can also explore [how to build a PyTorch container image](#) to streamline deployment across platforms.

Prerequisites

- Deploy a fresh [Ubuntu Server on Vultr](#)
- Using [SSH](#), access the server as a non-root user with sudo privileges
- [Update the server](#)

Install PyTorch on Ubuntu 22.04 using GPU and CPU-Only Methods

To install PyTorch on a server, verify the system capabilities to correctly enable the framework. In addition, you can install PyTorch as a native system package or install it using an environment management tool such as Conda as described in the steps below.

Install PyTorch on a GPU Server without Conda

1. Verify that your Server has a supported GPU driver. For example, view the Vultr NVidia GPU usage

```
$ nvidia-smi
```

If the above command fails, you cannot the PyTorch GPU package on the server, When successful, install the PyTorch GPU package

2. Upgrade the Python Pip package manager

```
$ pip install --upgrade pip
```

3. Using Pip, install the latest PyTorch version on your server

```
$ pip3 install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu118
```

The above command installs the latest PyTorch version built on the CUDA version `11.8`. The additional packages, `torchvision` and `torchaudio` extend PyTorch support with image and audio processing capabilities.

Using Conda

To install PyTorch on a GPU server, either [install Anaconda](#) or [Miniconda](#) then follow the steps below.

1. Activate your target Conda environment. For example `env1`

```
$ conda activate env1
```

2. Install the latest PyTorch version from the `pytorch` and the `nvdiia` channels

```
$ conda install pytorch torchvision torchaudio pytorch-cuda=11.8 -c pytorch -c nvidia
```

The above command installs the latest PyTorch version with the CUDA version `11.8`. Verify the latest version and install it in your environment.

Install PyTorch on a CPU-only Server

To install PyTorch on a CPU-only server without any GPU attachment, install the latest version together with the `torch`, `torchvision`, and `torchaudio` processing packages as described below.

1. Upgrade the Python Pip package manager

```
$ pip install --upgrade pip
```

2. Using Pip, install the latest PyTorch version

```
$ pip3 install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cpu
```

Install with Conda

To install PyTorch using Conda on a CPU-only system, install also install the `cpuonly` package from the `pytorch` Conda channel. Because this is a CPU-only environment, do not use packages from the `nvidia` channel.

1. Activate your target Conda environment. For example `env1`

```
$ conda activate env1
```

2. Install the latest PyTorch version from the `pytorch` channel

```
$ conda install pytorch torchvision torchaudio cpuonly -c pytorch
```

The above command installs PyTorch with the `cpuonly`, `torchvision`, and `torchaudio` packages in your Conda environment.

Test the PyTorch Installation

To verify that PyTorch is available and correctly installed on your server, perform the following test operations.

1. Access the Python Shell

```
$ python3
```

2. Import the `torch` package

```
>>> import torch
```

3. Declare a random tensor

```
>>> x = torch.rand(1)
```

4. Print the tensor value

```
>>> print(x)
```

Output:

```
tensor([0.4169])
```

As displayed in the above output, PyTorch is actively running and performing computation tasks on your server

Test PyTorch GPU Access

1. Access the Python Shell

```
$ python3
```

2. Import the PyTorch `torch` package

```
>>> import torch
```

3. Verify that PyTorch has access to the server GPU

```
>>> torch.cuda.is_available()
```

Output:

```
True
```

When the above result is **True**, PyTorch is correctly running with GPU access, If **False**, PyTorch cannot run with GPU-acceleration.

Conclusion

You have installed PyTorch on a Ubuntu server using both GPU and CPU-Only methods. Using PyTorch, you can extensively use other computation packages on your server to run and develop applications. For more information on how to install PyTorch, visit the [official installation documentation](#).



VULTR

