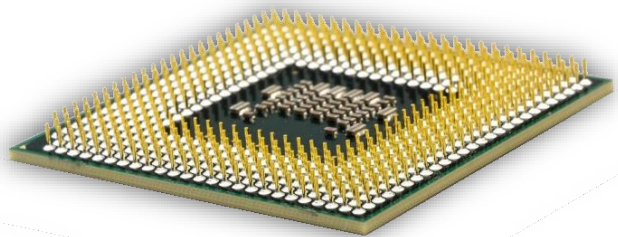


# Dispositivi hardware

# Dispositivi hardware

## CPU, GPU e TPU

In generale, le reti neurali sono implementate utilizzando software che viene poi eseguito su di un dispositivo hardware dedicato. Fra le diverse opzioni hardware, troviamo: CPU, GPU e TPU.



*Central processing unit*



*Graphics processing unit*



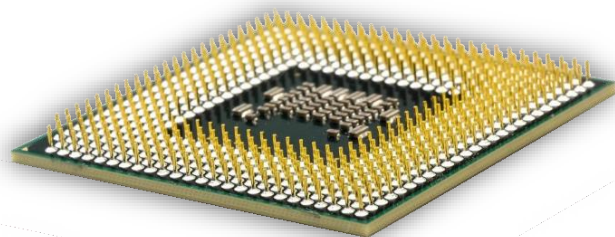
*Tensor processing unit*

Ognuno di questi dispositivi offre vantaggi unici per il training delle reti neurali.

# Dispositivi hardware

## CPU (Central Processing Unit)

- ▶ Fra i componenti fondamentali di ogni computer.
- ▶ Responsabile dell'esecuzione di operazioni, istruzioni logiche ed aritmetiche.
- ▶ Contribuisce all'esecuzione del sistema operativo e delle applicazioni.
- ▶ È l'hardware più semplice sul quale fare addestramento ed inferenza.
- ▶ Velocità di addestramento e inferenza di reti neurali sono limitate.



( Rif: [Microprocessori](#) )

# Dispositivi hardware

## GPU (Graphic Processing Unit)

- ▶ È un hardware specializzato.
- ▶ Progettato per velocizzare il processo di rendering grafico.
- ▶ Può eseguire molte operazioni in parallelo.
- ▶ Le stesse operazioni, se eseguite su CPU, sono spesso ordini di grandezza più lente.
- ▶ Fra i principali distributori (e competitor) per il ramo ML-DL: **Nvidia** e **AMD**.



Riferimenti:

- ▶ [NVIDIA](#)
- ▶ [Lista GPU](#)



Riferimenti:

- ▶ [AMD](#)
- ▶ [Lista GPU](#)

# Dispositivi hardware

## Tipi di GPU: integrata

Le GPU possono essere distinte per tipologia. Questo, sulla base principalmente di come vanno ad integrarsi al sistema con il quale dovranno interagire:

### *Integrata*

- ▶ La GPU è integrata nella CPU.
- ▶ Costruita sul processore.
- ▶ Richiede meno potenza.
- ▶ Produce meno calore.
- ▶ Per lo più presenti nei laptop.



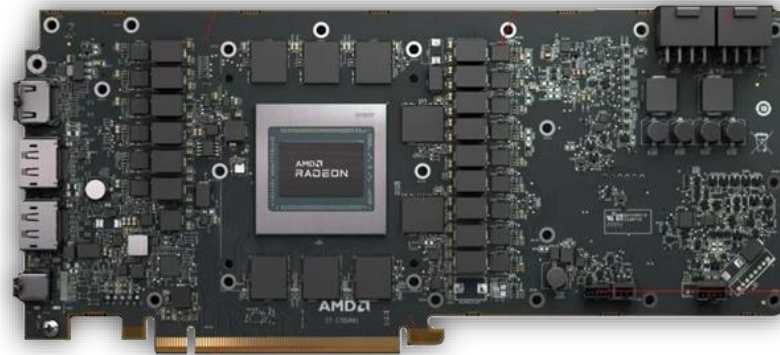
# Dispositivi hardware

## Tipi di GPU: discrete

Le GPU possono essere distinte per tipologia. Questo, sulla base principalmente di come vanno ad integrarsi al sistema con il quale dovranno interagire:

### *Embedded (discreta)*

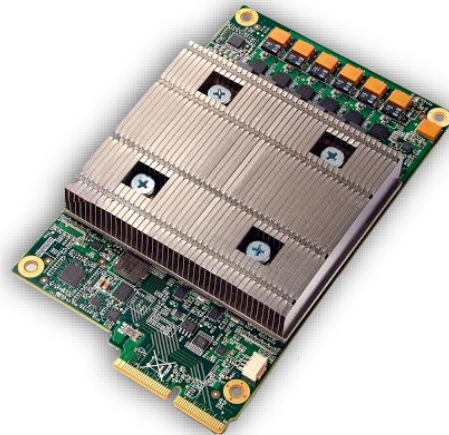
- ▶ La GPU è separata dal processore (CPU)
- ▶ Ha una sua memoria grafica dedicata.
- ▶ Richiede maggiore potenza.
- ▶ Produce più calore.
- ▶ Offre prestazioni superiori alle integrate.
- ▶ Per lo più presente nei PC Desktop.



# Dispositivi hardware

## TPU (Tensor Processing Unit)

- ▶ Sviluppato da Google appositamente per il machine learning.
- ▶ Progettato per accelerare il training delle reti neurali a basso costo energetico.
- ▶ A parità di potenza, gestiscono un numero di operazioni I/O superiore a quello di una GPU.
- ▶ Offre prestazioni potenzialmente superiori a quelle delle GPU.







# Dispositivi hardware

## PyTorch per CPU

Dal sito ufficiale di PyTorch è possibile selezionare diverse combinazioni di dispositivi e versioni per installare tutto il pacchetto PyTorch correttamente.

The screenshot shows the PyTorch 'GET STARTED' page. On the left, a dark purple box contains the text 'GET STARTED'. To the right, a configuration grid allows selecting various options. The 'Stable (2.0.0)' version is selected over 'Preview (Nightly)'. The 'Windows' operating system is selected over 'Linux' and 'Mac'. 'Conda' is selected as the package manager over 'Pip', 'LibTorch', and 'Source'. 'Python' is selected as the language over 'C++ / Java'. 'CPU' is selected as the compute platform over 'CUDA 11.7', 'CUDA 11.8', and 'ROCm 5.4.2'. At the bottom, a red-bordered box contains the command: `conda install pytorch torchvision torchaudio cpuonly -c pytorch`.

Stable (2.0.0)		Preview (Nightly)	
Linux	Mac	Windows	
Conda	Pip	LibTorch	Source
Python		C++ / Java	
CUDA 11.7	CUDA 11.8	ROCm 5.4.2	CPU

Run this Command:

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

In un environment conda (attivo) digitare il comando suggerito.

( Rif: [PyTorch](#) )

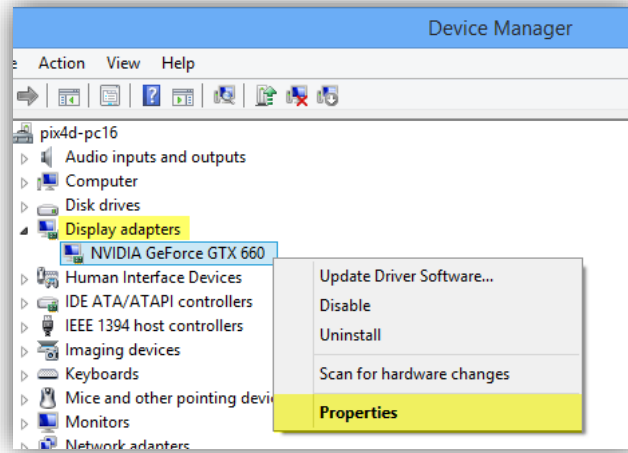




# Dispositivi hardware

## PyTorch per GPU (NVIDIA): ricerca della GPU

Per prima cosa, verificare di possedere una GPU nel proprio dispositivo.



Cercando in *gestione dispositivi*

Aprendo un *prompt dei comandi* e lanciando in esso il comando:

***nvidia-smi***

Se il comando eseguirà correttamente, saranno presentate le GPU presenti nel sistema e le loro caratteristiche.

La GPU rilevata deve essere fra quelle abilitate all'utilizzo del framework **CUDA**: un framework dedicato allo sviluppo e all'esecuzione di codice su GPU.

Rif: [CUDA GPUs - Compute Capability | NVIDIA Developer](#)



# Dispositivi hardware

## PyTorch per GPU (NVIDIA): aggiornamento dei driver

Verificato il tipo di scheda grafica NVIDIA presente, si procede quindi ad aggiornare i driver. Ci sono almeno due modi per farlo:

- ▶ Tentare l'aggiornamento dei driver da **gestione dispositivi** (Windows)
- ▶ Scaricarli ed installarli a partire dal seguente link: [DriverNVIDIA](#)  
Per farlo, indicare tipo e modello della scheda video e sistema operativo in cui si andrà a lavorare.

### Download di driver NVIDIA

Seleziona dal menu a discesa sotto per trovare il driver appropriato per il tuo prodotto NVIDIA. [Suggerimenti della Guida](#)

Tipo di prodotto:

GeForce

▼

Serie del prodotto:

GeForce RTX 40 Series (Notebooks)

▼

Famiglia di prodotti:

GeForce RTX 4090 Laptop GPU

▼

Sistema operativo:

Windows 10 64-bit

▼

Tipo di download:

Driver Game-Ready (GRD)

▼ ?

Lingua:

Italiano

▼

Ricerca



# Dispositivi hardware

## PyTorch per GPU (NVIDIA): CUDA Toolkit.

Aggiornato il driver, tenere presente che:

- ▶ Per utilizzare la tecnologia **CUDA** è necessario installare il **CUDA Toolkit**.
- ▶ Il Cuda Toolkit installabile dipende dalla versione del driver.

Rif: [CUDA Compatibility :: NVIDIA Data Center GPU Driver Documentation](#)

Table 1. Example CUDA Toolkit 11.x Minimum Required Driver Versions (Refer to CUDA Release Notes)

CUDA Toolkit	Linux x86_64 Minimum Required Driver Version	Windows Minimum Required Driver Version
CUDA 12.x	>=525.60.13	>=527.41
CUDA 11.x	>= 450.80.02*	>=452.39*

Table 2. CUDA Toolkit 10.x Minimum Required Driver Versions

CUDA Toolkit	Linux x86_64 Minimum Required Driver Version	Windows Minimum Required Driver Version
CUDA 10.2	>= 440.33	>=441.22
CUDA 10.1	>= 418.39	>=418.96
CUDA 10.0	>= 410.48	>=411.31



# Dispositivi hardware

## PyTorch per GPU (NVIDIA): CUDA Toolkit.

Scegliere quindi il CUDA Toolkit adeguato verificando sul sito di NVIDIA quale versione è compatibile in base alla versione del driver precedentemente aggiornato.

CUDA Toolkit	Minimum Required Driver Version for CUDA Minor Version Compatibility*	
	Linux x86_64 Driver Version	Windows x86_64 Driver Version
CUDA 12.1.x	>=525.60.13	>=527.41
CUDA 12.0.x	>=525.60.13	>=527.41
CUDA 11.8.x	>=450.80.02	>=452.39
CUDA 11.7.x	>=450.80.02	>=452.39
CUDA 11.6.x	>=450.80.02	>=452.39
CUDA 11.5.x	>=450.80.02	>=452.39
CUDA 11.4.x	>=450.80.02	>=452.39
CUDA 11.3.x	>=450.80.02	>=452.39
CUDA 11.2.x	>=450.80.02	>=452.39
CUDA 11.1 (11.1.0)	>=450.80.02	>=452.39
CUDA 11.0 (11.0.3)	>=450.36.06**	>=451.22**

( Rif: [DriverNVIDIA](#), [CUDA-ReleaseNotes](#) )



# Dispositivi hardware

## PyTorch per GPU (NVIDIA): CUDA Toolkit

Scaricare ed installare quindi il CUDA Toolkit: [CUDA Toolkit Archive](#)

Operating System

LinuxWindows

Architecture

x86\_64

Version

1011Server 2019Server 2022

Installer Type

exe (local)exe (network)

### Download Installer for Windows 10 x86\_64

The base installer is available for download below.

>Base Installer

Download (29.1 MB)

Installation Instructions:

1. Double click cuda\_12.1.0\_windows\_network.exe
2. Follow on-screen prompts

The checksums for the installer and patches can be found in [Installer Checksums](#).

For further information, see the [Installation Guide for Microsoft Windows](#) and the [CUDA Quick Start Guide](#).



# Dispositivi hardware

## PyTorch per GPU (NVIDIA): cuDnn

Installato il CUDA Toolkit, si procede con **cuDNN**, una libreria utilizzabile con CUDA e che raccoglie funzioni ottimizzate per il deep learning su GPU. Per scaricare ed installare le cuDNN, in base alla versione scelta del CUDA Toolkit, basta seguire il link, [cuDNN](#), e loggarsi.

### NVIDIA cuDNN

The NVIDIA CUDA® Deep Neural Network library (cuDNN) provides highly tuned implementations for normalization, and activation layers.

Deep learning researchers and framework developers worldwide rely on cuDNN to accelerate their software applications rather than spending time on low-level GPU performance optimization. cuDNN is used by **MxNet**, **PaddlePaddle**, **PyTorch**, and **TensorFlow**. For access to NVIDIA open source software, visit [NVIDIA Open Source](#) to learn more and get started.

[Download cuDNN >](#)

### cuDNN Archive

**NVIDIA cuDNN** is a GPU-accelerated library of primitives for deep neural networks.

[Download cuDNN v8.8.1 \(March 8th, 2023\), for CUDA 12.x](#)

[Download cuDNN v8.8.1 \(March 8th, 2023\), for CUDA 11.x](#)

[Download cuDNN v8.8.0 \(February 7th, 2023\), for CUDA 12.0](#)

[Download cuDNN v8.8.0 \(February 7th, 2023\), for CUDA 11.x](#)

[Download cuDNN v8.7.0 \(November 28th, 2022\), for CUDA 11.x](#)



# Dispositivi hardware

## PyTorch per GPU (NVIDIA): verifica installazione

Per verificare l'installazione di CUDA:

- ▶ Aprire un *prompt dei comandi*.
- ▶ Lanciare il comando ***nvcc --version***.

Se è stato installato correttamente, si leggerà:

- ▶ La versione installata, con corrispondente release.
- ▶ La data di installazione.

```

C:\> Prompt dei comandi
Microsoft Windows [Versione 10.0.17763.4131]
(c) 2018 Microsoft Corporation. Tutti i diritti sono riservati.

C:\Users\18P029-Dell>nvcc --version
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2021 NVIDIA Corporation
Built on Wed_Jun__2_19:25:35_Pacific_Daylight_Time_2021
Cuda compilation tools, release 11.4, V11.4.48
Build cuda_11.4.r11.4/compiler.30033411_0

```





# Dispositivi hardware

## PyTorch per GPU (NVIDIA): PyTorch

Dal sito ufficiale di PyTorch è possibile selezionare diverse combinazioni di dispositivi e versioni per installare tutto il pacchetto PyTorch correttamente.

The screenshot shows the PyTorch 'GET STARTED' page. The 'Stable (2.0.0)' version is selected. The 'Windows' operating system is chosen. The 'Conda' package manager is selected. The 'Python' language is chosen. The 'CUDA 11.7' compute platform is selected. The command to run in a conda environment is highlighted in a red box:

```
conda install pytorch torchvision torchaudio pytorch-cuda=11.7 -c pytorch -c nvidia
```

In un environment conda (attivo) digitare il comando suggerito.

( Rif: [PyTorch](#) )



# Dispositivi hardware

## PyTorch per GPU (NVIDIA): verifica installazione

Per verificare l'installazione di PyTorch per GPU:

- ▶ Verificare la GPU sia visibile e rilevata dal sistema operativo.
- ▶ Verifichiamo poi che la GPU risulti disponibile a PyTorch.
- ▶ Contiamo i device che utilizzano CUDA.
- ▶ Verifichiamo le informazioni sul device.

```
(Spettri) C:\Users\18P029-Dell>python
Python 3.8.13 | packaged by conda-forge | (default, Mar 25 2022, 05:59:45) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> torch.cuda.is_available()
True
>>> torch.cuda.device_count()
1
>>> torch.cuda.current_device()
0
>>> torch.cuda.device(0)
<torch.cuda.device object at 0x000001F6EE6F7850>
>>> torch.cuda.get_device_name(0)
'NVIDIA GeForce GTX 1660 Ti'
>>> _
```

( Rif: [torch.cuda](#))

Proviamo?