Introduzione

Istruzioni per l’installazione di TensorFlow (framework open source per machine learning) necessario per eseguire la fase di training e per il successivo riconoscimento di un oggetto specifico.

**Tensorflow è stato sviluppato e risulta essere funzionante su sistemi operativi:**

**macOS 10.12.6 (Sierra) o successivi**

**Ubuntu 16.04 o successivi**

**Windows 7 o successivi**

Per l’installazione di TensorFlow-GPU è necessaria una macchina con scheda grafica NVIDIA, su macchina virtuale non dovrebbe essere possibile installare questa versione.

Per l’installazione di TensorFlow-CPU invece non vi sono limitazioni hardware, tuttavia i sistemi operativi differenti da quelli citati sopra non dovrebbero permettere l’installazione.

La seguente guida mostra come eseguire l’installazione in ambiente Linux, nello specifico testato e funzionante su Ubuntu 18.04 (GPU e CPU )e 16.04 (solo CPU).

E’ possibile anche l’uso di un ambiente virtuale come virtual env o Anaconda, ulteriori informazioni e guide per installazione su altri sistemi operativi, sono reperibili alla pagina di tensorflow: <https://www.tensorflow.org/install/>

1. [Installazione TensorFlow-GPU…………………………………………………………………………………………………..2](#TensorFlowGPU)
2. [Installazione TensorFlow-CPU…………………………………………………………………………………………………..5](#TensorFlowCPU)

**Installazione TENSORFLOW-GPU 1.8 su Ubuntu 18.04**

**Requisito OS: Ubuntu 18.04 NON VIRTUALIZZATO**

**Per l’installazione di TensorFlow-GPU è necessaria una GPU Nvidia e l’installazione di CUDA. Alla seguente pagina è possibile verificare se la propria GPU supporta CUDA, è necessaria una capacità computazionale superiore o uguale a 3.0.** <https://developer.nvidia.com/cuda-gpus>

1. Rimuovere tutti gli eventuali driver Nvidia:

**sudo apt-get purge nvidia\***

1. Aggiungere driver grafici PPA:

**sudo add-apt-repository ppa:graphics-drivers**

**sudo apt-get update**

1. Installare driver Nvidia:

**sudo apt-get install nvidia-390**

eseguire **sudo** **reboot**

**ATTENZIONE!** In questa fase potrebbe essere richiesta una password per il security boot, in questo caso leggere attentamente le istruzioni mostrate a schermo: abilitare la password, al reboot attivare la modalità MOK (machine owner key), vedere la chiave ed eseguire il reboot da essa, in questo modo verrà autorizzato l’uso dei driver di terze parti installati. (Potrebbero esservi variazioni a seconda del BIOS).

1. Check installazione:

**nvidia-smi**

**lsmod | grep nvidia**

**lsmod | grep nouveau**

**Se viene mostrato un messaggio di errore o risulta abilitato nouveau è necessario disabilitare nouveau altrimenti proseguire al** [passo 10](#GPU10)

**Rimozione nouveau:**

1. Creare file: **/etc/modprobe.d/blacklist-nouveau.conf**
2. Inserire il seguente contenuto:

**blacklist nouveau**

**options nouveau modeset=0**

1. Rigenerare l’initramfs del kernel:

**sudo update-initramfs -u**

1. Reboot del sistema: **sudo reboot**
2. Se il sistema è a risoluzione minore è ok, controllare comunque con

**lsmod | grep nouveau**

**Tornare al passo 3 (se ricapitassero gli stessi problemi ripetere dal passo 1)**

**INSTALLAZIONE CUDA**

1. Andare al link seguente:

<https://developer.nvidia.com/cuda-90-download-archive>

Scaricare la versione:

**Linux -> x86\_64 -> Ubuntu -> 17.04 -> deb**

11. Copiare nel terminale le seguenti righe dalla cartella di download:

**sudo dpkg -i cuda-repo-ubuntu1704-9-0-local\_9.0.176-1\_amd64.deb**

**sudo apt-key add /var/cuda-repo-9-0-local/7fa2af80.pub**

**sudo apt-get update**

**sudo apt-get install cuda**

1. Scaricare ed installare con l’installer predefinito le due patch alla pagina linkata in precedenza.
2. Aggiungere percorsi a PATH, per fare questo scrivere dentro al file bash e copiare in fondo le seguenti righe digitando:

**sudo nano ~/.bashrc**

**export PATH=/usr/local/cuda-9.0/bin${PATH:+:${PATH}}**

**export LD\_LIBRARY\_PATH=/usr/local/cuda-9.0/lib64${LD\_LIBRARY\_PATH:+:${LD\_LIBRARY\_PATH}}**

1. Controllare i due PATH con

**echo $PATH**

**echo $LD\_LIBRARY\_PATH**

1. Se non fossero presenti digitare i comandi 2 e 3 al passo 12 da terminale e ripetere passo 13

**INSTALLARE CUDNN**

1. Andare al seguente link

<https://developer.nvidia.com/cudnn>

1. Eseguire i seguenti step:

**Download Cudnn**

**Login** (potrebbe essere necessario registrarsi)

**Archieved cudnn releases**

**Download cuDNN v7.0.5 (Dec 5, 2017), for CUDA 9.0**

**cuDNN v7.0.5 Library for Linux**

1. Da terminale, nella cartella di download eseguire:

**tar -xzvf cudnn-9.0-linux-x64-v7.tgz**

1. Eseguire i seguenti comandi da terminale per copiare files:

**sudo cp cuda/include/cudnn.h /usr/local/cuda/include**

**sudo cp cuda/lib64/libcudnn\* /usr/local/cuda/lib64**

**sudo chmod a+r /usr/local/cuda/include/cudnn.h /usr/local/cuda/lib64/libcudnn\***

1. Installare TensorFlow, da terminale eseguire:

**pip install tensorflow-gpu**

(potrebbe essere richiesta l’installazione di pip seguire istruzioni su terminale).

1. Testare lanciando python da terminale ed eseguendo:

**import tensorflow as tf**

**hello = tf.constant('Hello, TensorFlow!’)**

**sess = tf.Session()**

**print(sess.run(hello))**

(L’installazione è avvenuta correttamente se eseguento tf.Session() vengono mostrate informazioni relative alla GPU).

21. Installare da terminale le seguenti componenti che risulteranno essere utili in seguito:

**pip install pillow**

**pip install lxml**

**pip install jupyter**

**pip install matplotlib**

**pip install pandas**

**sudo apt install protobuf-compiler**

**Installazione TENSORFLOW-CPU**

1**.** Installare TensorfFlow, da terminale eseguire:

**pip install tensorflow**

(potrebbe essere richiesta l’installazione di pip seguire istruzioni su terminale).

**NOTA** Non eseguire l’aggiornamento di pip alla versione 10.0.1 poiché causerebbe problemi nell’installazione delle componenti successive; nel caso lo si avesse già aggiornato rimuovere l’ultima versione ripristinandolo alla 8.1.1 eseguendo da terminale:

**python -m pip uninstall pip**

2. Testare eseguendo:

Da terminale digitare:

**python**

Quindi inserire:

**import tensorflow as tf**

**hello = tf.constant('Hello, TensorFlow!’)**

**sess = tf.Session()**

**print(sess.run(hello))**

(L’installazione è avvenuta correttamente se viene stampata la scritta Hello, TensorFlow).

3. Installare da terminale le seguenti componenti che risulteranno essere utili in seguito:

**pip install pillow**

**pip install lxml**

**pip install jupyter**

**pip install matplotlib**

**pip install pandas**

**sudo apt install protobuf-compiler**