

Google Cloud Platform Quick Start Guide

1. Create VM

Open the Deep Learning VM Deployment Page in the [GCP Marketplace](#). We selected the n1-highmem-4 which utilized 4 vCPUs and includes 26 GB memory. Add a GPU, the NVIDIA Tesla T4 is the least expensive option and will work for our needs. A more expensive option will speed up training and detection times. Verify "Install NVIDIA GPU driver automatically on first startup?" is selected. A minimum 150 GB persistent boot disk size should be selected, however more would be better depending on the dataset being used and how much information you plan on working on at once.

Google Cloud Platform KAB-MachineLearning

New Deep Learning VM deployment

Deployment name *
deeplearning-1

Zone
us-west1-b
GPU availability is limited to certain zones. [Learn more](#)

Machine type

Machine family
GENERAL-PURPOSE COMPUTE-OPTIMIZED MEMORY-OPTIMIZED GPU

Machine types for common workloads, optimized for cost and flexibility

Series
N1
Powered by Intel Skylake CPU platform or one of its predecessors

Machine type
n1-highmem-4 (4 vCPU, 26 GB memory)

vCPU 4 Memory 26 GB

GPUs

The number of attached GPUs affects the VM's maximum number of memory and CPUs. [Learn More](#)

GPU type
NVIDIA Tesla T4
Number of GPUs
1

☐ Enable Virtual Workstation (NVIDIA GRID)

CPU PLATFORM AND GPU

Framework *
TensorFlow Enterprise 2.3 (CUDA 11.0)
Choose the primary machine learning framework you will be using. If the library you would like to use is not listed, choose the base image, which provides core packages.

GPU

☒ Install NVIDIA GPU driver automatically on first startup?
I want to use NVIDIA GPUs with this image. Please fetch NVIDIA GPU drivers from a third-party location and install them on my behalf (requires internet access on the VM).

Access to the Jupyter Lab

☐ Enable access to JupyterLab via URL instead of SSH. (Beta)
Enable this feature to access the running JupyterLab environment through a URL instead of SSH tunneling. Anyone who has been granted the Editor or Owner role in your GCP project can access this URL. This feature is available only in the US, EU and Asia.

Boot Disk

Boot disk type *
Standard Persistent Disk

Boot disk size in GB *
150

Networking

Network interfaces
default default (10.138.0.0/20)
[ADD NETWORK INTERFACE](#)

☒ I accept the [GCP Marketplace Terms of Service](#).

[DEPLOY](#)

Deep Learning VM overview
Product provided by Google Click to Deploy

Click to Deploy gvmc enabled in DeepLearning VMs Usage Fee <small>Google DeepLearning VM does not charge a usage fee.</small>	USD 0.00/mo
Infrastructure fee VM instance: 4 vCPUs + 26 GB memory (n1-highmem-4)	USD 172.72/mo
Standard Persistent Disk: 150GB	USD 7.08/mo
NVIDIA Tesla T4 GPU	USD 255.50/mo
Sustained use discount	- USD 128.47/mo
Estimated monthly total	USD 306.84/mo

Software

Operating System
Debian(10)

Documentation

[Official Documentation](#)
[StackOverflow: Deep Learning VM](#)
[Google Group: Deep Learning VM](#)

Terms of Service

By deploying the software or accessing the service you are agreeing to comply with the [GCP Marketplace terms of service](#) and the terms of applicable open source software licenses bundled with the software or service. Please review these terms and licenses carefully for details about any obligations you may have related to the software or service. To the limited extent an open source software license related to the software or service expressly supersedes the GCP Marketplace Terms of Service, that open source software license governs your use of that software or service.

By using this product, you understand that certain account and usage information may be shared with Google Click to Deploy for the purposes of financial accounting, sales attribution, performance analysis, and support.

Google is providing this software or service "as-is" and any support for this software or service will be provided by Google Click to Deploy under their terms of service.

Google Cloud Platform Quick Start Guide

2. Setup VM

In your instance clone our copy of the YOLOv5 repo which includes our TCP Detection code. Install requirements.txt in a Python $\geq 3.7.0$ environment (3.9.0 if performing hyperparameter evolution training) and PyTorch ≥ 1.7 .

```
git clone https://github.com/SushiTeam2022/KAAB-ML/tree/main/YOLOv5 # clone
cd yolov5
pip install -r requirements.txt #install
```

3. Train

These commands are examples on how to train data using the YOLOv5 machine learning model. These are the settings we used which we found worked best on small litter objects. The data.yaml file will come from your dataset. Setting the batch size to -1 uses YOLOv5's auto-batching feature to maximize the capability of your GPU.

```
python train.py --data data.yaml --cfg yolov5l.yaml --weights " " --batch-size -1 --name "training run name"
```

4. Optional Setup

a. Weights and Biases

Weights and Biases (WANDB) is a website that allows an easy to use interface to track and compare training runs. This is an optional component, but very useful to quickly review data from many different training runs and their parameters. You can either first make an account on <https://wandb.ai> or when first running you will be prompted to create an account. Use your API key, (from <https://wandb.ai/authorize> if you are already a user) to login.

```
pip install wandb
wandb login
```

b. Anaconda Environment

Anaconda allows you to create different environments to develop in so that you can run a program with separate settings than what you may want to run your system on. If the sole purpose of this machine is to run this program Anaconda may not be necessary. For verifying different system configurations prior to actual deployment, Anaconda is recommended. To create an environment with a specific Python version the command is as follows, and remember all "pip install" commands ran in one environment will not transfer to another so the requirements will need to be installed in each environment..

```
conda create --name nameOfYourEnvironment python=3.9
```

The environment must be activated, and deactivated before returning to the normal system state. You will know you are in the environment when the name appears on the left side of your screen.

```
USERNAME: $ conda activate nameOfYourEnvironment
(nameOfYourEnvironment) USERNAME: $
(nameOfYourEnvironment) USERNAME: conda deactivate
```