



CMLG Course Lab I

Haoteng Yin, Shu Li
CS@Purdue



Outline

- How to use Google Cloud
- Build a GNN with PyG
- Q & A



How to use Google Cloud for computation?

Get Ready:


- Sign up for Google Cloud Platform
- [Google Cloud Platform welcome page](#)
- Creates a project and note the Project ID
- Set up a billing account and Link to your project
- Redeem Google Cloud Education Credit (see email earlier)



Create a VM with attached GPUs

Before you begin

- Read about GPU pricing on Compute Engine to understand the cost to use GPUs on your VMs.
- Read about restrictions for VMs with GPUs.
- Check your GPU quota:
 - New projects have a global GPU quota, which limits the total number of GPUs you can create in any supported zone.

Iowa (us-central1) ▾				
Model	GPUs	GPU memory	GPU price (USD)	Spot price* (USD)
NVIDIA A100 80GB 	1 GPU	80 GB HBM2	\$3.93 per GPU	\$1.25 per GPU



Check your GPU quota

- Use the [regions describe command](#) to ensure that you have sufficient GPU quota in the region where you want to create VMs with GPUs.

```
$ gcloud compute regions describe REGION
```

- If you need additional GPU quota, [request a quota increase](#).
 - When you request a GPU quota, you must request a quota for
 - the GPU types that you want to create in each region
 - an additional global quota for the total number of GPUs of all types in all zones.

Request a GPU quota increase

- In the Google Cloud console, [Go to Quotas](#)
- Click filter_list **Filter table** and select **Service**.
- Choose **Compute Engine API**.
- Choose **Limit Name: GPU-ALL-REGIONS-per-project**.
- Click the checkbox of the region whose quota you want to change.
- Click create **Edit Quotas**.
- Complete the form.
- Click **Submit Request**. (normally within 5 mins)

Filter Service : Compute Engine API Limit name : GPUS-ALL-REGIONS-per-project					
<input checked="" type="checkbox"/>	Service	Quota	Dimensions (e.g. location)	Limit	Current us
<input checked="" type="checkbox"/>	Compute Engine API	GPUs (all regions)		1	

Quota changes

Expand each service card to change individual quotas.

Compute Engine API

Quota: GPUs (all regions)

Current limit: 1

Enter a new quota limit. A limit above 1 will require approval from your service provider. ?

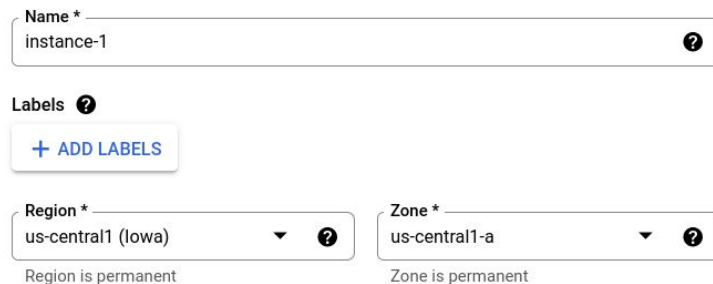
New limit is required.

DONE



Create a virtual machine with attached GPUs

- In the Google Cloud console, [Go to Create an instance](#)
- Specify a **Name** for your VM.
- Select a region and zone that supports the GPU model.
 - To learn about GPU models supported by different regions and zones, see [GPU regions and zones availability](#).



A screenshot of the Google Cloud console form for creating a virtual machine instance. The form includes the following fields and options:

- Name ***: A text input field containing "instance-1". A help icon (?) is visible on the right.
- Labels ?**: A section with a "+ ADD LABELS" button.
- Region ***: A dropdown menu showing "us-central1 (Iowa)". A help icon (?) is visible on the right. Below the dropdown, it says "Region is permanent".
- Zone ***: A dropdown menu showing "us-central1-a". A help icon (?) is visible on the right. Below the dropdown, it says "Zone is permanent".

Machine configuration

Complete the following steps:

- In **Machine family**, select **GPU** and the **Series** will be **N1**.
 - Specify the GPU type and number of GPUs.
- In **Machine type**, select a [N1 machine type](#). Alternatively, you can specify custom machine type settings.
[standard, high RAM/CPU]
- E.g. Nvidia K80, N1-Standard-2: vCPU #2, Memory 7.5 GB

Machine configuration

Machine family

GENERAL-PURPOSE

COMPUTE-OPTIMIZED

MEMORY-OPTIMIZED

GPU

Optimized for machine learning, high performance computing, and visualization workloads

GPU type

NVIDIA Tesla K80

Number of GPUs

1

☐ Enable Virtual Workstation (NVIDIA GRID)

Series

N1

Machine type

n1-standard-2 (2 vCPU, 7.5 GB memory)



vCPU

2

Memory

7.5 GB

CPU platform

Automatic



Choose an operating system image

- To select your operating system, in the **Boot disk** section, click **Change**. This opens the **Boot disk configuration** page.
- On the **Boot disk configuration** page, do the following:
 - On the **Public images** tab, choose a [supported Compute Engine image](#) or [Deep Learning VM image](#).
 - Specify a boot disk size of at least 40 GB.
 - To confirm your boot disk options, click **Save**.

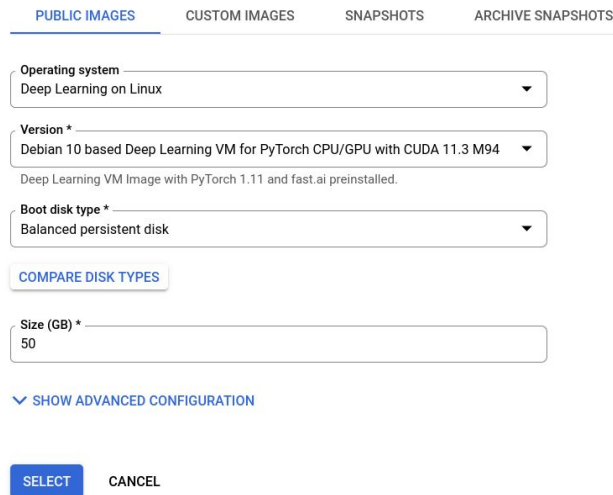


Switch to images with preinstalled CUDA stack

Use a Deep Learning VM image with GPU drivers pre-installed and include packages, such as TensorFlow / PyTorch.

- OS: Deep Learning on Linux
- Version: M94 with PyTorch 1.11 and fast.ai preinstalled
- Disk Size: > 40 GB

[!] You may need to reinstall Nvidia driver after the first initialization of VM.



The screenshot shows the Google Cloud Platform console interface for selecting a VM image. At the top, there are four tabs: "PUBLIC IMAGES" (selected), "CUSTOM IMAGES", "SNAPSHOTS", and "ARCHIVE SNAPSHOTS". Below the tabs, there are three dropdown menus: "Operating system" with "Deep Learning on Linux" selected, "Version *" with "Debian 10 based Deep Learning VM for PyTorch CPU/GPU with CUDA 11.3 M94" selected, and "Boot disk type *" with "Balanced persistent disk" selected. Below these dropdowns is a button labeled "COMPARE DISK TYPES". Underneath that is a "Size (GB) *" input field with the value "50". At the bottom, there is a link that says "SHOW ADVANCED CONFIGURATION" with a downward arrow icon. At the very bottom, there are two buttons: "SELECT" and "CANCEL".

PUBLIC IMAGES CUSTOM IMAGES SNAPSHOTS ARCHIVE SNAPSHOTS

Operating system
Deep Learning on Linux

Version *
Debian 10 based Deep Learning VM for PyTorch CPU/GPU with CUDA 11.3 M94
Deep Learning VM Image with PyTorch 1.11 and fast.ai preinstalled.

Boot disk type *
Balanced persistent disk

COMPARE DISK TYPES

Size (GB) *
50

SHOW ADVANCED CONFIGURATION

SELECT CANCEL



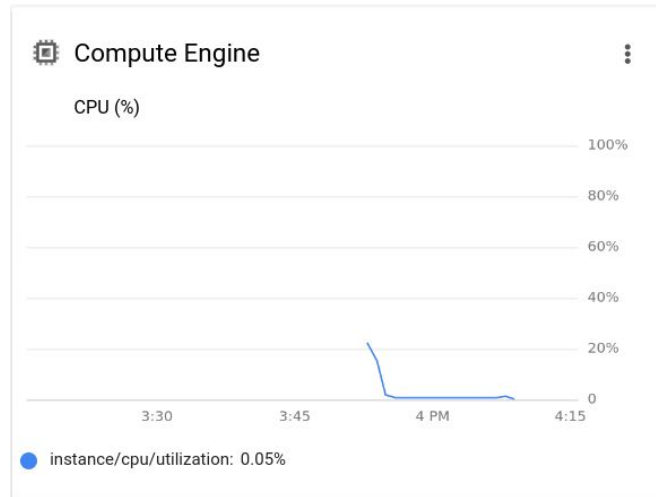
Machine configuration

- Configure other VM settings that you require.
 - Firewall and Network Traffic
 - Change the **Preemptibility / Spot** settings to configure your VM as a preemptible VM.
- To create and start the VM, click **Create**.

Prepare the instance

- Allow a short period of time for the instance to start. After the instance is ready, it's listed on the **VM instances** page with a green status icon.
- Compute Engine grants the user who creates the VM with the roles/compute.instanceAdmin role, and adds that user to the sudo group.

[!] It is highly recommended to keep close eyes for your running instances and monitoring their metrics and maximize their utilization for a lower cost.



Connect to the VM instance

Connect to an instance by using the Google Cloud console and completing the following steps.

- In the Google Cloud console, [Go to VM instances](#)
- In the list of virtual machine instances, click **SSH** in the row of the instance that you want to connect to.

<input type="checkbox"/> Name ^	Zone	Recommendation	Internal IP	External IP	Connect
<input type="checkbox"/>  instance-1	us-east1-b		10.142.0.2 (nic0)	35.231.114.114 	SSH ▾ ⋮

How to start a GCE on Colab

- Colaboratory or 'Colab' allows you to write and execute Python in your browser. It allows you to combine executable code and rich text in a single document.
- Visit [Colab on GCP Marketplace](#) to launch a preconfigured VM purchase flow.
- Similar **Machine Configuration** but do aware the quota for GPU and Boot Disk. But this can **only** work for Colab.



Colab

[Google Colab](#)

A hosted notebook solution for machine learning

LAUNCH

VIEW PAST DEPLOYMENTS

New Colab deployment

Deployment name *

colab-2

Zone

us-central1-a

GPU availability is limited to certain zones. [Learn more](#)

Machine type

Machine family

GENERAL-PURPOSE

COMPUTE-OPTIMIZED

MEMORY-OPTIMIZED

GPU

Optimized for machine learning, high performance computing, and visualization workloads

GPU type

NVIDIA Tesla K80

Number of GPUs

1

☐ Enable Virtual Workstation (NVIDIA GRID)

Series

N1

Machine type

n1-standard-4 (4 vCPU, 15 GB memory)



vCPU

4

Memory

15 GB

Boot Disk

Boot disk type *

Extreme Persistent Disk

Connect Colab to your GCE VM

- Use the link from within GCP's Deployment Manager.
- The link is present in the details of your Colab deployment.
- If you wish to connect from within Colab, from the Connect arrow within Colab select "Connect to a Custom GCE VM".
- Fill in the resulting dialog with the information from your VM configuration and click Connect.

Get started with Colab

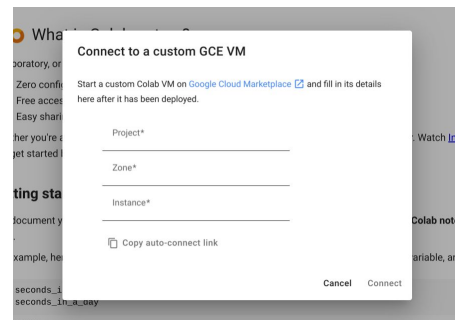
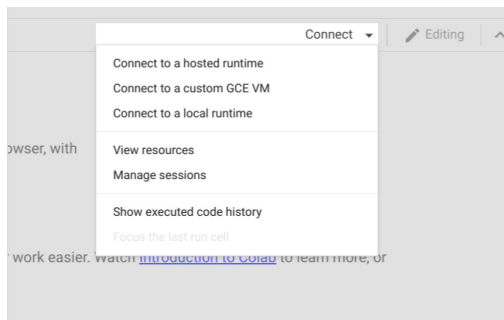
[CONNECT TO VM WITH COLAB](#)

Documentation

- [Google Cloud Platform VM Instances](#)
- [Colaboratory FAQ](#)

Support

[Go to Google Colab support](#)





Build a graph learning pipeline with PyG

PyG is a library built upon PyTorch to easily write and train Graph Neural Networks for a wide range of applications related to structured data.

PyG is both friendly to machine learning researchers and first-time users of machine learning toolkits.

Example: Colab [[Link](#)], Runtime Type: GPU

```
dataset = Planetoid(root='.', name='Cora')

class GCN(torch.nn.Module):
    def __init__(self, in_channels, hidden_channels, out_channels):
        super().__init__()
        self.conv1 = GCNConv(in_channels, hidden_channels)
        self.conv2 = GCNConv(hidden_channels, out_channels)

    def forward(self, x: Tensor, edge_index: Tensor) -> Tensor:
        x = self.conv1(x, edge_index).relu()
        x = self.conv2(x, edge_index)
        return x

model = GCN(dataset.num_features, 16, dataset.num_classes)
```




Q & A