

[AI & Machine Learning Products](https://cloud.google.com/products/machine-learning/) (<https://cloud.google.com/products/machine-learning/>)

[Cloud TPU](https://cloud.google.com/tpu/) (<https://cloud.google.com/tpu/>) [Guides](#)

Running MNIST on Cloud TPU (TF 2.x)

Alpha

This product or feature is in a pre-release state and might change or have limited support. For more information, see the [product launch stages](https://cloud.google.com/products/#product-launch-stages) (<https://cloud.google.com/products/#product-launch-stages>).

This tutorial contains a high-level description of the MNIST model, instructions on downloading the MNIST TensorFlow TPU code sample, and a guide to running the code on Cloud TPU.

Note: This information is also covered in the [Cloud TPU quickstart](https://cloud.google.com/tpu/docs/quickstart) (<https://cloud.google.com/tpu/docs/quickstart>).

Disclaimer

This tutorial uses a third-party dataset. Google provides no representation, warranty, or other guarantees about the validity, or any other aspects of, this dataset.

Model description

The [MNIST dataset](http://yann.lecun.com/exdb/mnist/) (<http://yann.lecun.com/exdb/mnist/>) contains a large number of images of hand-written digits in the range 0 to 9, as well as the labels identifying the digit in each image.

This tutorial trains a machine learning model to classify images based on the MNIST dataset. After training, the model classifies incoming images into 10 categories (0 to 9) based on what it's learned about handwritten images from the MNIST dataset. You can then send the model an image that it hasn't seen before, and the model identifies the digit in the image based on what the model has learned during training.

The MNIST dataset has been split into three parts:

- 60,000 examples of training data
- 10,000 examples of test data
- 5,000 examples of validation data

You can find more information about the dataset at the [MNIST database site](http://yann.lecun.com/exdb/mnist/) (<http://yann.lecun.com/exdb/mnist/>).

The model has a mixture of seven layers:

- 2 x convolution
- 2 x max pooling
- 2 x dense (fully connected)
- 1 x dropout

Loss is computed via categorical cross entropy.

This version of the MNIST model uses the Keras API, a recommended way to build and run a machine learning model on a Cloud TPU.

Keras simplifies the model development process by hiding most of the low-level implementation, which also makes it easy to switch between TPU and other test platforms such as GPUs or CPUs.

Note: This model performs training only.

Objectives

- Create a Cloud Storage bucket to hold your dataset and model output.
- Run the training job.
- Verify the output results.

Costs

This tutorial uses billable components of Google Cloud, including:

- Compute Engine
- Cloud TPU
- Cloud Storage

Use the [pricing calculator](https://cloud.google.com/products/calculator/) (https://cloud.google.com/products/calculator/) to generate a cost estimate based on your projected usage. New Google Cloud users might be eligible for a [free trial](https://cloud.google.com/free/) (https://cloud.google.com/free/).

Before you begin

This section provides information on setting up Cloud Storage storage and a Compute Engine VM.

Important: Set up your Compute Engine VM, your Cloud TPU node and your Cloud Storage bucket in the same region/zone to reduce network latency and network costs.

1. Open a Cloud Shell window.

[OPEN CLOUD SHELL](https://console.cloud.google.com/?cloudshell=true) (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/?CLOUDSHELL=TRUE)

2. Create a variable for your project's name.

```
export PROJECT_NAME=project-name
```

3. Configure `gcloud` command-line tool to use the project where you want to create Cloud TPU.

```
gcloud config set project $PROJECT_NAME
```

4. Create a Cloud Storage bucket using the following command:

Note: In the following command, replace *bucket-name* with the name you want to assign to your bucket.

```
gsutil mb -p $PROJECT_NAME -c standard -l us-central1-b -b on gs://bucket-name
```

This Cloud Storage bucket stores the data you use to train your model and the training results.

5. Launch a Compute Engine VM and Cloud TPU using the `ctpu up` command.

```
$ ctpu up --zone=us-central1-b --tf-version=2.1  
[optional flags: --project, --name]
```

Note: If you have more than one project, you must specify the project name with the `--project` flag. If the Compute Engine VM name is not specified with the `--name` flag, it defaults to your username.

6. The configuration you specified appears. Enter `y` to approve or `n` to cancel.

7. When the `ctpu up` command has finished executing, verify that your shell prompt has changed from `username@project` to `username@tpuname`. This change shows that you are now logged into your Compute Engine VM.

Note: If you are not connected to the Compute Engine instance, you can connect by running the following commands, replacing `vm-name` with your VM name and `tpu-name` with your TPU name. The VM and TPU names default to your username:

```
gcloud compute ssh vm-name --zone=us-central1-b
```

```
(vm)$ export TPU_NAME=tpu-name
```

As you continue these instructions, run each command that begins with `(vm)$` in your VM session window.

Run the MNIST TPU model

The MNIST TPU model is pre-installed on your Compute Engine VM. Change to directory that stores the model:

```
(vm)$ cd /usr/share/models/official/vision/image_classification
```

The source code for the MNIST TPU model is available on [GitHub](https://github.com)

(https://github.com/tensorflow/models/blob/master/official/vision/image_classification/mnist_main.py).

You can run the model on a Cloud TPU. Alternatively, see how to [run the model on a local machine](#) (#run-local).

Set up environment variables

Export the following variables. Replace **bucket-name** with your bucket name:

```
(vm)$ export STORAGE_BUCKET=gs://bucket-name
(vm)$ export MODEL_DIR=$STORAGE_BUCKET/mnist
(vm)$ export DATA_DIR=$STORAGE_BUCKET/data
(vm)$ export PYTHONPATH="$PYTHONPATH:/usr/share/models"
```



Run the model on Cloud TPU

1. Run the MNIST model:

```
(vm)$ python3 mnist_main.py \
  --tpu=$TPU_NAME \
  --model_dir=$MODEL_DIR \
  --data_dir=$DATA_DIR \
  --train_epochs=10 \
  --distribution_strategy=tpu \
  --download
```



ParameterDescription

tpu	The name of the Cloud TPU. If not specified when setting up the Compute Engine VM and Cloud TPU, defaults to your username.
model_dir	This is the directory that contains the model files. This tutorial uses a folder within the Cloud Storage bucket. You do not have to create this folder beforehand. The script creates the folder if it does not already exist.
data_dir	This is the directory that contains files used for training.
download	When specified (set to true), the script downloads and preprocesses the MNIST dataset, if it hasn't been downloaded already.

The model runs in under 5 minutes on a v3-8 Cloud TPU and displays output similar to:

```
I1203 03:43:15.936553 140096948798912 mnist_main.py:165]
Run stats: {'loss': 0.11427700750786683, 'training_accuracy_top_1': 0.96576976776123
'accuracy_top_1': 0.9730902910232544, 'eval_loss': 0.08600160645114051}
```

Cleaning up

To avoid incurring charges to your Google Cloud Platform account for the resources used in this tutorial:

1. Disconnect from the Compute Engine instance, if you have not already done so:

```
(vm)$ exit
```

Your prompt should now be `user@projectname`, showing you are in the Cloud Shell.

2. In your Cloud Shell, run `ctpu delete` with the `--zone` flag you used when you set up the Compute Engine VM and Cloud TPU. This deletes both your VM and your Cloud TPU.

```
$ ctpu delete --zone=us-central1-b
```

Important: If you set the TPU resources name when you ran `ctpu up`, you must specify that name with the `--name` flag when you run `ctpu delete` in order to shut down your TPU resources.

3. Run `ctpu status` to make sure you have no instances allocated to avoid unnecessary charges for TPU usage. The deletion might take several minutes. A response like the one below indicates there are no more allocated instances:

```
$ ctpu status --zone=us-central1-b
```

```
2018/04/28 16:16:23 WARNING: Setting zone to "us-central1-b"
No instances currently exist.
  Compute Engine VM:      --
  Cloud TPU:              --
```

4. Run `gsutil` as shown, replacing **bucket-name** with the name of the Cloud Storage bucket you created for this tutorial:

```
$ gsutil rm -r gs://bucket-name
```

Note: For free storage limits and other pricing information, see the [Cloud Storage pricing guide](https://cloud.google.com/storage/pricing) (<https://cloud.google.com/storage/pricing>).

What's next

- Explore the other [Cloud TPU tutorials](https://cloud.google.com/tpu/docs/tutorials) (<https://cloud.google.com/tpu/docs/tutorials>).
- Learn to use the [TPU monitoring tools in TensorBoard](https://cloud.google.com/tpu/docs/cloud-tpu-tools) (<https://cloud.google.com/tpu/docs/cloud-tpu-tools>).
- Verify performance on a large-scale model by [running the ResNet sample](https://cloud.google.com/tpu/docs/tutorials/resnet) (<https://cloud.google.com/tpu/docs/tutorials/resnet>).

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (<https://creativecommons.org/licenses/by/4.0/>), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (<https://www.apache.org/licenses/LICENSE-2.0>). For details, see our [Site Policies](https://developers.google.com/terms/site-policies) (<https://developers.google.com/terms/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.

Last updated January 18, 2020.