# Downloading, pre-processing, and uploading the ImageNet dataset

This topic describes how to download, pre-process, and upload the ImageNet dataset to use with Cloud TPU. Machine learning models that use the ImageNet dataset include:

- ResNet
- AmoebaNet
- EfficientNet
- MNASNet

ImageNet is an image database. The images in the database are organized into a hierarchy, with each node of the hierarchy depicted by hundreds and thousands of images.

The size of the ImageNet database means it can take a considerable amount of time to train a model. An alternative is to use a demonstration version of the dataset, referred to as *fake_imagenet*. This demonstration version allows you to test the model, while reducing the storage and time requirements typically associated with using the full ImageNet database.

## Pre-processing the full ImageNet dataset

The ImageNet dataset consists of three parts, training data, validation data, and image labels.

The training data contains 1000 categories and 1.2 million images, packaged for easy downloading. The validation and test data are not contained in the ImageNet training data (duplicates have been removed).

The validation and test data consists of 150,000 photographs, collected from flickr and other search engines, hand labeled with the presence or absence of 1000 object categories. The 1000 object categories contain both internal nodes and leaf nodes of ImageNet, but do not overlap with each other. A random subset of 50,000 of the images with labels has been released as validation data along with a list of the 1000 categories. The remaining images are used for evaluation and have been released without labels.

### Steps to pre-processing the full ImageNet dataset

There are five steps to preparing the full ImageNet dataset for use by a Machine Learning model:

1. Verify that you have space on the download target.
2. Set up the target directories.
3. Register on the ImageNet site and request download permission.
4. Download the dataset to local disk or Compute Engine VM.

> **Note:** Downloading to a Compute Engine VM using Cloud Shell takes considerably longer than downloading to your local machine (approximately 40 hours versus 7 hours). However, if you download to your local machine, you then need to copy the files to a Compute Engine VM to pre-process them and upload them to Cloud Storage before using them to run your model. Copying all of the training and

validation files from you local machine to the VM takes about 13 hours. The least complicated, and recommended, approach is to download to the VM.

5. Run the pre-processing and upload script.

## Verify space requirements

Whether you download the dataset to your local machine or to a Compute Engine VM, you need about 300GB of space available on the download target. On a VM, you can check your available storage with the `df -ha` command.

**Note:** If you use `ctpu up` to set up your VM, it will allocate 250GB by default.

You can increase the size of the VM disk using one of the following methods:

- Specify the `--disk-size-gb` flag on the `ctpu up` command line with the size, in GB, that you want allocated.
- Follow the Compute Engine guide to add a disk to your VM.
  - Set **When deleting instance** to **Delete disk** to ensure that the disk is removed when you remove the VM.
  - Make a note of the path to your new disk. For example: `/mnt/disks/mnt-dir`.

## Set up the target directories

On your local machine or Compute Engine VM, set up the directory structure to store the downloaded data.

1. Create and export a home directory for the ImageNet dataset.

   Create a directory, for example, `imagenet` under your home directory on your local machine or VM. Under this directory, create two sub-directories: `train` and `validation`. Export the home directory as IMAGENET_HOME:

   ```
   export IMAGENET_HOME=~/imagenet
   ```

## Register and request permission to download the dataset

1. Register on the Imagenet website. It will take some time for your registration to be processed, but once it is, go to the download site.

## Download the ImageNet dataset

1. From the download site, go to the Images section on the page and right click on "Training images (Task 1 & 2)". This will give you the URL needed to download the largest part of the training set. Save the URL.

Right click on "Training images (Task 3)" to get the URL for the second training set. Save the URL.

Right click on "Validation images (all tasks)" to get the URL for the validation data set. Save the URL.

If you download the ImageNet files to your local machine, after the download completes, you need to copy the directories on your local machine to the corresponding `$IMAGENET_HOME` directory on your Compute Engine VM. Copying the ImageNet dataset from local host to your VM takes approximately 13 hours.

For example the following command copies all of the files under $IMAGENET_HOME on your local machine to your VM that displays the shell prompt ***username@vm-name***:

```
gcloud compute scp --recurse $IMAGENET_HOME username@vm-name:~/imagenet
```

2. From $IMAGENET_HOME, use `wget` to download the training and validation files using the saved URLs.

The "Training images (Task 1 & 2)" file is the large training set. It is 138GB and if you are downloading to a Compute Engine VM using the Cloud Shell, the estimated time to download is approximately 40 hours. For this particularly large file, you can prepend `nohup` to the command or use screen to avoid having the download terminate if the Cloud Shell disconnects.

```
cd $IMAGENET_HOME \
nohup wget http://image-net.org/challenges/LSVRC/2012/dd31405981ef5f776aa17412e1f0c112
```

This downloads a large tar file: ILSVRC2012_img_train.tar.

From $IMAGENET_HOME on the VM, extract the individual training directories into the `$IMAGENET_HOME/train` directory using the following command. The extraction takes between 1 - 3 hours.

```
tar xf ILSVRC2012_img_train.tar
```

The "Training images (Task 3)" file is 728 MB and takes just a few minutes to download so you do not need to take precautions against losing the Cloud Shell connection.

When you download this file, it extracts the individual training directories into the existing `$IMAGENET_HOME/train` directory.

```
wget http://www.image-net.org/challenges/LSVRC/2012/dd31405981ef5f776aa17412e1f0c112/
```

The "Validation images (all tasks)" file is 6GB, so you might want to use `nohup` or screen to avoid having the download terminate if the Cloud Shell disconnects.

```
wget http://www.image-net.org/challenges/LSVRC/2012/dd31405981ef5f776aa17412e1f0c112/
```

This download takes about 30 minutes. When you download this file, it extracts the individual validation directories into the `$IMAGENET_HOME/validation` directory.

If you downloaded the validation files to your local machine, you need to copy the `$IMAGENET_HOME/validation` directory on your local machine to the `$IMAGENET_HOME/validation` directory on your Compute Engine VM. This copy operation takes about 30 minutes.

Download the labels file. This take just seconds.

```
wget -O $IMAGENET_HOME/synset_labels.txt \
https://raw.githubusercontent.com/tensorflow/models/master/research/inception/incepti
```

If you downloaded the labels file to your local machine, you need to copy it to the `$IMAGENET_HOME`directory on your local machine to `$IMAGENET_HOME` on your Compute Engine VM. This copy operation takes a few seconds.

The training subdirectory names (for example, n03062245) are "WordNet IDs" (wnid). The ImageNet API shows the mapping of WordNet IDs to their associated validation labels in the `synset_labels.txt` file. A synset in this context is a visually-similar group of images.

## Process the Imagenet dataset and, optionally, upload to Cloud Storage

1. Download the `imagenet_to_gcs.py` script from GitHub:

```
wget https://raw.githubusercontent.com/tensorflow/tpu/master/tools/datasets/imagenet_
```

2. If you are uploading the dataset to Cloud Storage, specify the storage bucket location to upload the ImageNet dataset:

```
export STORAGE_BUCKET=gs://bucket-name
```

3. If you are uploading the dataset to Cloud Storage, specify a storage bucket data directory to hold the dataset:

```
(vm)$ export DATA_DIR=$STORAGE_BUCKET/dataset-directory
```

4. Run the script to pre-process the raw dataset as TFRecords and upload it to Cloud Storage using the following command:

> **Note:** If you don't want to upload to Cloud Storage, specify `--nogcs_upload` as another parameter and leave off the `--project` and `--gcs_outpu_path` parameters.

```
python3 imagenet_to_gcs.py \
 --project=$PROJECT \
 --gcs_output_path=$DATA_DIR  \
 --raw_data_dir=$IMAGENET_HOME \
 --local_scratch_dir=$IMAGENET_HOME/tf_records
```

> **Note:** Downloading and preprocessing the data can take 10 or more hours, depending on your network and computer speed. Do not interrupt the script.

The script generates a set of directories (for both training and validation) of the form:

```
${DATA_DIR}/train-00000-of-01024
${DATA_DIR}/train-00001-of-01024
 ...
${DATA_DIR}/train-01023-of-01024
```

and

```
${DATA_DIR}/validation-00000-of-00128
S{DATA_DIR}/validation-00001-of-00128
 ...
${DATA_DIR}/validation-00127-of-00128
```

After the data has been uploaded to your Cloud bucket, run your model and set `--data_dir=${DATA_DIR}`.