

# Instructions to run the code of Group 13 - COMP0197

This code is adapted from Meta's implementation of MAE:

<https://github.com/facebookresearch/mae/tree/main>

## Setting Up and Installing

### Dependencies

- Additional pip-installable packages:
  - Tensorboard
  - Matplotlib
  - Timm==0.3.2
    - Go to where timm is installed (most likely  
~/miniconda3/envs/comp0197-cw2-pt/lib/python3.12/site-packages/timm/models  
/layers/helpers.py) and change the code in helpers.py to:

```
""" Layer/Module Helpers

Hacked together by / Copyright 2020 Ross Wightman
"""
from itertools import repeat
import torch
# from torch._six import container_abcs

TORCH_MAJOR = int(torch.__version__.split('.')[0])
TORCH_MINOR = int(torch.__version__.split('.')[1])

if TORCH_MAJOR == 1 and TORCH_MINOR < 8:
    from torch._six import container_abcs
else:
    import collections.abc as container_abcs

# From PyTorch internals
def _ntuple(n):
    def parse(x):
        if isinstance(x, container_abcs.Iterable):
            return x
        return tuple(repeat(x, n))
    return parse

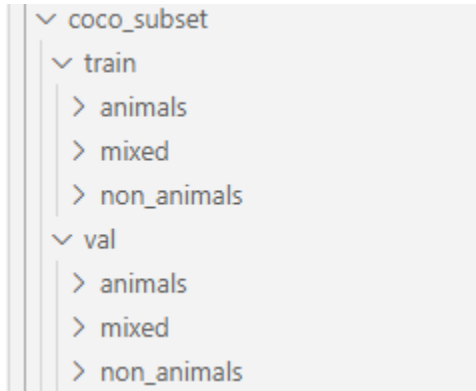
to_1tuple = _ntuple(1)
to_2tuple = _ntuple(2)
```

```
to_3tuple = _ntuple(3)
to_4tuple = _ntuple(4)
to_ntuple = _ntuple
```

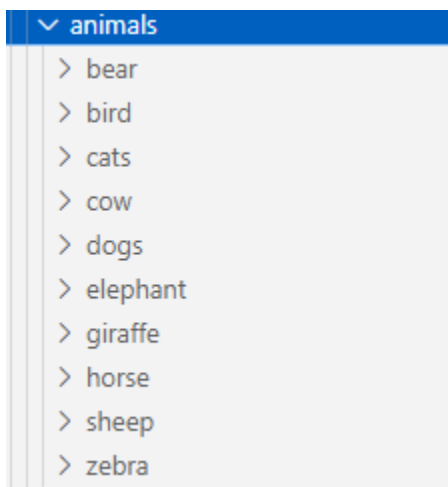
## COMPLETE

### **Pretraining**

**Sorting data:** Download the data in coco\_subset with the following file structure



Each 'animals', 'mixed', and 'non\_animals' subfolder should further split into class labels. E.g. for animals, make sure to have



The data is hosted at:

[https://liveuclac-my.sharepoint.com/:f:/g/personal/ucabns4\\_ucl\\_ac\\_uk/EmZuYDFVlOhEhwaeohhHimsB8ScNteZF3ed\\_X0T85Bs9g?e=O9aWVv](https://liveuclac-my.sharepoint.com/:f:/g/personal/ucabns4_ucl_ac_uk/EmZuYDFVlOhEhwaeohhHimsB8ScNteZF3ed_X0T85Bs9g?e=O9aWVv)

## Pretraining Commands

### Pretraining on Nonanimals Dataset

```
python main_pretrain.py --data_path ./coco_subset/train/ --output_dir  
output_dir_judyNonanimals25k --log_dir output_dir_judyNonanimals25k --epochs 150  
--batch_size 32 --which_subset non_animals
```

### Pretraining on Animals Dataset

```
python main_pretrain.py --data_path ./coco_subset/train/ --output_dir output_dir_judyAnimals  
--log_dir output_dir_judyAnimals --epochs 150 --batch_size 32 --which_subset animals
```

### Pretraining on Mixed Dataset

```
python main_pretrain.py --data_path ./coco_subset/train/ --output_dir output_dir_anabel_mixed  
--log_dir output_dir_anabel_mixed --epochs 150 --batch_size 32 --which_subset mixed
```

There will be a checkpoint for every 20 epochs, i.e. in the output directory folder. These files will be named checkpoint-{epoch}.pth. There will also be a log.txt detailing the training loss for every epoch.

To get the **validation losses for each checkpoint**, run

```
python pretrain_metrics.py --which_subset animals --input_dir output_dir_judyAnimals/  
--dataset_type val/ --output_dir output_dir_pretrain/
```

```
python pretrain_metrics.py --which_subset non_animals --input_dir output_dir_judyNonanimals/  
--dataset_type val/ --output_dir output_dir_pretrain/
```

```
python pretrain_metrics.py --which_subset mixed --input_dir output_dir_judyMixed/  
--dataset_type val/ --output_dir output_dir_pretrain/
```

A log{which\_subset}.txt file will be output into output\_dir\_pretrain, detailing the validation loss for each saved checkpoint.

## **Finetuning**

To get OxfordPet dataset (outside mae folder)

Type the following in terminal (navigate outside of the mae directory):

```
wget -P OxfordPet/ https://thor.robots.ox.ac.uk/~vgg/data/pets/images.tar.gz
```

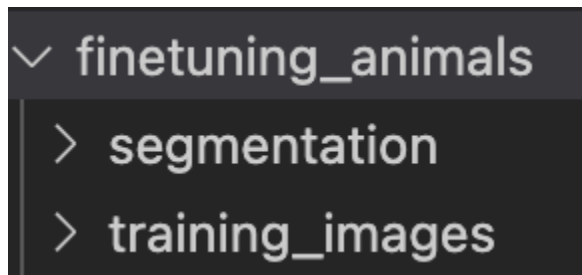
```
wget -P OxfordPet/ https://thor.robots.ox.ac.uk/~vgg/data/pets/annotations.tar.gz
```

The instructions for finetuning follow a similar logic to the pretraining steps.

There will be a checkpoint for every 20 epochs, i.e. in the output directory folder. These files will be named checkpoint-{epoch}.pth. There will also be a log.txt detailing the training loss for every epoch.

The main file “finetune2.py” is called to run fine-tuning. This script relies also on “engine\_finetune2.py” and “models\_mae\_finetune.py”. The former has the boiler code to run 1 epoch of fine-tuning, doing training, evaluation on validation set, and outputting loss, IOU, accuracy and example segmentations on both training and validation sets. The latter has the MAE model that is fine-tuned. Note one must specify which directories to output the model and results to (--output\_dir), as well as the directory where the pre-trained model is (--finetune).

To test this out, a output\_dir\_judyNonanimals/checkpoint-149.pth has been created in <https://drive.google.com/file/d/1QGI4rKoG6TMDyYAOKTydO4MmyEtEhcO/view?usp=sharing> . Make sure it is in output\_dir\_judyNonanimals directory. Also please ensure the output directory you specify in the scripts below has the following subdirectories inside it (assume in this case that finetuning\_animals is the output directory you have specified). The segmentation folder will have segmentation examples on the validation set, whereas training\_images will have segmentations on the training set (both every 5 epochs).



Finetune for Nonanimals

```
python finetune2.py --batch_size 64 --output_dir output_dir_ftNA25k --finetune  
output_dir_judyNonanimals/checkpoint-149.pth
```

Finetune for Animals

```
python finetune2.py --batch_size 64 --output_dir output_dir_sam/finetuning_animals --finetune  
output_dir_judyAnimals/checkpoint-149.pth
```

Finetune for Mixed

```
python finetune2.py --batch_size 64 --output_dir output_dir_sam/finetuning_nonanimals  
--finetune output_dir_judyMixed/checkpoint-149.pth
```

## **ResNet**

Before executing the file *train.py* it is crucial to set the path **in that file** as seen below:

```
def main():  
    # Configuration and hyperparameters  
    root_dir = '/cs/student/projects3/COMP0197/grp3/adl_groupwork/adl_supervised_learning/' # set path to adl_supervised_learning
```

The `root_dir` should be set to the project folder which is:

“adl\_groupwork/adl\_supervised\_learning”

To train, go to `adl_supervised_learning` and run  
`python train.py`

This will generate checkpoint models at each epoch, as well as a `log.txt` file for both training and testing in the `log` folder detailing measured metrics.

## **Plotting**

Navigate to the ‘Plotting’ folder within the `mae` directory

To generate the loss plot for pretraining, run the following command and specify the train and validation paths within the Plotting directory:

```
python plot_pretrain.py --train_log_files [TRAIN LOG PATH 1] [TRAIN LOG PATH 2] [TRAIN LOG PATH 3] --val_log_files [VAL LOG PATH 1] [VAL LOG PATH 2] [VAL LOG PATH 3] --labels Animals  
Non-animals Mixed
```

Example:

```
python plot_pretrain.py --train_log_files  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_judyAnimals/log.txt  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_judyNonanimals/log.txt  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_anabel_mixed/log.txt --val_log_files  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_pretrain/val/log_animals.txt  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_pretrain/val/log_non_animals.txt  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_pretrain/val/log_mixed.txt --labels  
Animals Non-animals Mixed
```

To generate the loss plot for resnet comparison experiment, run the following command within the Plotting directory:

```
python plot_finetune_res.py --log_files [LOG PATH 1] [LOG PATH 2] [LOG PATH 3] --labels "Finetuned"  
"Finetuned-half" "Resnet-50" "MAE Supervised"
```

Example:

```
python plot_finetune_res.py --log_files  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_sam/finetuning_animals/log.txt  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_sam/finetuning_less_data/log.txt  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/adl_supervised_learning/logs/training_log.txt  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_supervised/log.txt --labels  
"Finetuned" "Finetuned-half" "Resnet-50" "MAE Supervised"
```

To generate the loss plot for animal comparison experiment, run the following command within the Plotting directory:

```
python plot_finetune_mix.py --log_files [LOG PATH 1] [LOG PATH 2] [LOG PATH 3] --labels Animals  
Non-animals Mixed
```

Example:

```
python plot_finetune_mix.py --log_files  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_sam/finetuning_animals/log.txt  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_ftNA25k/log.txt  
/cs/student/projects3/COMP0197/grp3/adl_groupwork/mae/output_dir_sam/finetuning_mixed/log.txt --labels  
Animals Non-animals Mixed
```

```
python plot_pretrain.py --train_log_files ../output_dir_judyAnimals/log.txt  
../output_dir_judyNonanimals/log.txt ../output_dir_judyMixed/log.txt --val_log_files  
../output_dir_pretrain/val/log_animals.txt ../output_dir_pretrain/val/log_non_animals.txt  
../output_dir_pretrain/val/log_mixed.txt --labels Animals Non-animals Mixed
```

### **Supervised**

```
python supervised.py --batch_size 64 --output_dir output_dir_supervised --log_dir  
output_dir_supervised
```

### **Running on CPU**

All code can be run on CPU as long as the flag `--device=cpu` is provided