Instruction

This file serves as instructions to properly set up the environment to run scripts of the project.

Package installation

First, activate the `conda` environment.

```
conda activate comp0197-cw1-pt
```

Then, install the three packages used in the project, in addition to the packages already included in the `conda` environment:

```
conda install -c huggingface -c conda-forge datasets
conda install segmentation-models-pytorch
conda install matplotlib
```

Pretraining

To pretrain models, simply run the scripts in the `pre-train` directory. Note that downloading dataset could take time.

```
python pre-train/pretrain_cifar.py
python pre-train/pretrain_pets.py
```

We also prepared pre-trained models in `pretrain_model` directory. You can directly use these as foundations for fine-tuning.

Fine-tuning

To fine-tune the models, first make sure the pretrained backbones are placed in `pretrain_model` directory (we have prepared pretrained backbones in it). Then, we need to prepare the fine-tuning dataset.

```
cd fine-tune
mkdir models
mkdir models/model_files
mkdir models/json_result

# Download data
wget http://www.robots.ox.ac.uk/~vgg/data/pets/data/images.tar.gz
wget http://www.robots.ox.ac.uk/~vgg/data/pets/data/annotations.tar.gz

# Unzip data
tar zxvf images.tar.gz
tar zxvf annotations.tar.gz

# Delete unused files in the dataset
cd annotations/trimaps
```

```
find . -type f -name '._*.png' -print
find . -type f -name '._*.png' -exec rm {} +

cd ../../images
find . -type f -name '*.mat' -print
find . -type f -name '*.mat' -exec rm {} +
```

Then, navigate back to the root folder of the project, and we can fine-tune the models. Here, you can choose to fine-tune on the model pretrained with `pet` or `cifar` dataset, or `baseline` to skip pretraining, and what percentage of fine-tuning data to use.

```
cd fine-tune

# Users need to specify which model to run with 3 choices (pet, cifar, baseline)
# Users also need to determine how much data are used for fine-tuning (20, 50, 80)

python fine_tune.py pet 20
# Fine-tune the model pretrained on cats vs dogs, with 20% of finetuning data used

python fine_tune.py cifar 50
# Fine-tune the model pretrained on cifar-10, with 50% of finetuning data used

python fine_tune.py baseline 80
# Fine-tune the model without pretraining, with 80% of finetuning data used
```

The resulted model will be saved in `fine-tune/models`.

Evaluation

To evaluate the model's performance on the test dataset. We first go to the fine-tune folder.

```
cd fine-tune
mkdir results

# Run the evaluate script, which will run tests on all the models files that are previously saved.
python evaluate.py
```