COMP0197 Self-supervised learning

Self supervised learning group work.

Dependencies

A Conda installation is assumed for installation of packages.

If CUDA is available create a new environment

```
conda create -n ssl -c conda-forge -c pytorch -c nvidia python=3.11 pytorch=2.1
    pytorch-cuda=12.1 torchvision=0.16 torchmetrics=1.3
```

The extra packages compared to the standard environment are `pytorch-cuda=12.1` and `torchmetrics=1.3`. To install `pytorch-cuda=12.1` the `nvidia` channel was added with `-c nvidia`

If CUDA is not available create a new environment

```
conda create -n ssl -c conda-forge -c pytorch python=3.11 pytorch=2.1
    torchvision=0.16 torchmetrics=1.3
```

The extra package compared to the standard environment is `torchmetrics=1.3`.

Running the scripts

There are three scripts to run, all in the `unet/` directory. Change the current working directory into `unet/`

```
cd unet
```

When running, the scripts will download the required data into `data/` in the main directory.

Pretrain

To pretrain models

```
python pretrain.py
```

This will save the pretrained models in `saved_models/` and save example images in `example_images/`.

Segmentation train

To perform semantic segmentation training

```
python seg_train.py
```

This will save the segmentation models in `saved_models/` with file names indicating which dataset they came from, the run number and the size of the training set used.

Generate test intersection over union (IoU) results

To generate the test IoU outputs to a csv file

```
python iou_stats.py
```

This will output a csv file in `saved_models/test_ious.csv` containing the test IoU results for each pretraining strategy, run number and training subset. These are the results that are used in the report. `kaggle_seg` indicates Kaggle Dogs vs. Cats pretrain dataset, `synth_seg` indicates synthetic stable diffusion dataset and `no_pretrain` indicates the model with no pretraining.