Project Code Setup Guide

April 14, 2024

1 Introduction

This document provides all the necessary instructions and steps required to run the code submitted by our team. Please follow the steps below to ensure that you can replicate our experiment results.

2 Environment Setup

This section describes the setup of the ‘comp0090-cw1-pt‘ Python environment used for our experiments. The environment is created and managed using Conda, with additional packages installed via pip.

2.1 Conda Environment

To create and activate the environment, use the following commands:

```
conda create -n comp0090-cw1-pt -c pytorch python=3.10 pytorch=2.1.2 \
torchvision=0.16.2 cudatoolkit=12.1

conda activate comp0090-cw1-pt
```

2.2 Additional Packages

Additional image processing packages required are installed using pip:

```
pip install opencv-python
pip install imgaug
```

3 Project Structure

- **Python Scripts**

  `main.py` Main script to conduct our experiments.

`data_utils.py` Handles data loading and preprocessing tasks, ensuring the data is correctly formatted and ready for both training and test phases.

`data_augmentation.py` Contains code for augmenting the pre-training datasets.

`models.py` Defines the masked auto encoder network models used for segmentation.

`losses.py` Provides definitions for the loss functions used to train the models.

`metrics.py` Defines metrics used to evaluate the performance of the models during and after training.

`utils.py` Includes utility functions that are used across the project, such as visualization tools to analyze model performance and training results.

- **Directories**

`datasets/` This directory stores the origin datasets and augmented data prepared for use in training and testing the models.

`experiments/` Contains separate code files for each experiment conducted, including:

  - `compare_pretrained_with_baseline`
  - `compare_pretrained_model_finetuning_sizes`
  - `compare_data_similarity_for_segmentation`

Each script is dedicated to a specific analysis or comparison.

`Images/` Holds visualization images that depict the results of experiments.

4 Running Code & Reproducing the Experiments

Here are the steps to run our project code. The experiment scripts are stored in 'experiments/'. There are two methods you can adopt to replicate these experiments:

- Run the corresponding experiment scripts directly, following the instruction in Section 4.2, 4.3 and 4.4.
- Run the script 'main.py' by passing the experiment name as an argument, following the instruction in Section 4.5.

Ensure you have set up the environment as described in the previous section.

4.1 Pre-train data download

We have uploaded the original pre-training data to Google Drive. If you want to preview and use it, please click the link to download. This link has been set to public access: data[1]
And due to excessive data, data enhancement is also very time-consuming, so we uploaded the enhanced data to Google Drive:aug_data[2]
In addition, the pre-training dataset for OEQ's experiment, also referred to in 4.4, is linked below: flower[3]

4.2 compare_pretrained_with_baseline.py

The purpose of this experiment is to compare the framework with a baseline model trained on the same finetuning data, using fully supervised methods. Only a pretrained dataset needs to be prepared, located at ‘/datasets/data‘ to directly run the code.

```
python compare_pretrained_with_baseline.py
```

4.3 compare_pretrained_model_finetuning_sizes.py

The purpose of this experiment is to compare the benefit of the pretrained segmentation model, using different finetuning dataset sizes. Only a pretrained dataset needs to be prepared, located at ‘/datasets/data‘ to directly run the code.

```
python compare_pretrained_model_finetuning_sizes.py
```

4.4 compare_data_similarity_for_segmentation.py

The purpose of this experiment is to explore how similar the pretraining and finetuning/test data need to be for a better segmentation model. A pretrained dataset related to pets is prepared, located at ‘/datasets/data‘, and another unrelated pretrained dataset, located at ‘/datasets/102flowers‘, needs to be prepared to run the code.

```
python compare_data_similarity_for_segmentation.py
```

[1] https://drive.google.com/file/d/1QHC18el6TemZBouOXOumJ-WEivq4blxN/view?usp=sharing

[2] https://drive.google.com/file/d/1–AAWSQTVH5opJy9vIfGxLaUdpAPB4If/view?usp=sharing

[3] https://www.robots.ox.ac.uk/ vgg/data/flowers/102/102flowers.tgz

4.5 Run experiments through main.py

With the related dataset prepared, the three aforementioned experiments can be run with the unified entrance 'main.py' by passing the experiment name as an argument, choosing from one of the predefined experiments: 'compare_baseline', 'compare_data_size_ft' or 'compare_data_sim' listed below, corresponding to the experiments in Section 4.2, 4.3 and 4.4 respectively.

```
python main.py compare_baseline
python main.py compare_data_size
python main.py compare_data_sim
```