COMP0197: Applied Deep Learning
Assessed Component 2 (Group Work – 25%) 2023-24
Assessed Component 3 (Individual Report – 25%) 2023-24

Submission before 16:00 (UK time), 15th April 2024 (subject to change), on Moodle

Project: Self-supervised pretraining for segmentation

1 Introduction

Self-supervised learning is a subfield in machine learning that develops methods utilising potentially large amount of unlabelled data, before training (finetuning) a supervised model. It is a typical example used for pretraining stage of modern foundation models. Recent work using deep neural network (pre-)training strategies such as masked autoencoder and contrastive learning algorithms can be considered.

2 Project

In this project, you will implement and evaluate a self-supervised neural network for semantic segmentation of images (group work), then describe the methods, implementation and results in a report (individual report).

The aim of this project is to assess your ability to reason about design choices in deep learning experiments, to implement deep learning framework, evaluate the developed deep learning models, work in a group and summarise and communicate your work in a technical report.

2.1 The data set

The Oxford-IIIT Pet Data-set will be used as the finetuning and testing data in this project. The data contains a few thousand images with their pixel-level annotations of animals – semantic segmentation labels. In addition, these are categorised into types of animals, which could be used as classification labels but optional for this project.

2.2 The "minimum required project"

The so-called minimum required project (MRP) (70%) for this group work consists of the following points, which are expected to be found in your submission. Some of these points are discussed in more detail to get you started.

- Deciding and justifying which self-supervised segmentation algorithm to use.
    - Use references and citations to support your choice.
- Identify and collect pretraining data, and explain the motivation in the selected pretraining data.
- Implementing a self-supervised segmentation framework for pretraining a segmentation model.
- Designing and implementing a finetuning method using a subset of the Pet data set.
- Designing and conducting experiments for network comparisons, at least:
    - Compare the framework with a baseline model trained on the same finetuning data, using fully supervised methods.
    - Compare the benefit of the pretrained segmentation model, using different finetuning data set sizes.

- Describing implemented methods and conducted experiments.
- Summarising obtained results and drawing conclusions.

2.3 An "open-ended question"

To be awarded the remaining 30% in this project, you need to come up with a new study question to answer, an open-ended question (OEQ). Novelty is encouraged in this part. The group should clearly identify and generate a hypothesis (i.e. the study question), design experiments that produce results to answer this question and analyse the obtained experimental results for quantitative conclusion. This part needs to be built on the MRP and relevant to self-supervised segmentation algorithms. Describe the question, experiment and results clearly and cohesively with the MRP in the report.

Some example study questions are given as follows.

- Does increase unlabelled pretraining data always help?
- How similar the pretraining and finetuning/test data need to be, for a better segmentation model?
- What has been learned from the pretraining data?
- ……

3 Submission and Marking Criteria

Each group will submit one bundle of code in a single zip file and an accompanying instruction pdf file (50 MB file size limit applies). Individual will submit a report in a pdf file. Both should be submitted before the same deadline.

3.1 Code

In this project, you must only use one of the PyTorch or TensorFlow for the entirety of the project. However, you can use up to **three additional pip-installable packages** within the "comp0090-cw1-xx" conda environment, in addition to the available tools in the conda environment.

A "instruction.pdf" file is required for 1) detailing the installed (within the "comp0090-cw1-xx") additional packages, if used, and 2) listing detailed steps to run your code, reproducing all the results included in the report. It is your responsibility to ensure the compatibility of the packages you added and that they can run on a marking environment on Ubuntu 20.04. There is no specific format or style required for how you design, structure and implement your code. However, general good programming practice and readability of the code may contribute towards meeting the requirements in marking criteria below.

3.2 Report

Each group member needs to submit one report (details to follow). Unlike the first individual coursework, the marking is primarily based on the reports. The submitted report must contain the following sections (followed with the suggested content), structured as a research paper. The percentages indicate the maximum marks for each section, each with a 7:3 split for the MRP and OEQ, respectively.

Note: your report must not exceed 6 pages in the LNCS template, excluding references. Attempts to gain extra space or pages, for example by modifying fonts or margins, will be considered as a case of exceeding word/page limit, which will be penalised according to the UCL regulations.

- Title: an interesting and informative title of the study (0%)
- Introduction: background, literature, motivation, the research/study questions to address (20%)

- Methods: the methodological details for the implemented semi-supervised algorithms and other networks, optional novel methodology for the OEQ (25%)
- Experiments: two MRP comparisons, other experiments for the OEQ (25%)
- Results: a summary of results, comparison, quantitative analysis (20%)
- Discussion: interesting findings, unanswered questions, limitations and future directions (5%)
- Conclusion: summarising the study and the results (5%)

3.3 Marking

The marking criteria used is adapted using the departmental project marking criteria (both UG and PG) as reference. To summarise, in descending order of importance:

- Scientific soundness: reasoning and justification of problems, methods and experiments, background reading and understanding, experiment design and quality.
- Technical accuracy: appropriate and correct use of terminology, methodology, data, code and other tools.
- Completeness: objective achieved, conclusion of study question, completeness of report.
- Presentation: writing organisation, clarity, illustrative report, structure, code readability.
- Critical appraisal: conclusive results, informative analysis, future outlook.
- Novelty (only relevant to OEQ): this point can be subjective, and can be particularly related to the background and existing literature.

3.4 Peer assessment

There is no mandatory peer assessment. The same grade on the code from each group will be assigned to all group members, while scores on reports are individually marked. However, cases of negligence or any other academic malpractice, if reported, will be investigated and assessed per UCL regulations, on a case-by-case basis.