Summative Machine Learning Coursework

Dr Chris G. Willcocks

Last Modified: February 5, 2019

Introduction

This coursework has two parts relating to the CIFAR-100 and CIFAR-10 datasets:

1. Building a classifier [**50 marks**].
2. Generating an image of a Pegasus [**50 marks**].

You are to write a scientific report with a maximum of 1,500 words, detailing which architectural designs were considered, which experiments were conducted, and how the model(s) were evaluated and improved. Also you are required to submit working code (NCC/local .py or Colab .ipynb) for your final solutions.

Part 1: Building a Classifier

Train and build a classifier that generalises well for the CIFAR-100 dataset using the techniques you've learnt in the lectures. For your best architecture, you must submit:

1. A plot of the test accuracy and a plot of the test loss.
2. The value of your best reported test accuracy (e.g. 42%).

Discuss and/or show diagrams of architectures/components you have tried, and explain what worked well and what didn't. You will be marked as follows:

- Discussion of ideas/experiments conducted to improve the architecture [**25 marks**].
- Suitability of final architecture [**10 marks**].
- Best test accuracy [**10 marks**].
- Novelty of ideas/decisions/processes [**5 marks**].

Base code for the classifier as is available here [Classifier Google Colab Link].

Part 2: Generating a Pegasus

Training on data only from CIFAR-10 (not CIFAR-100), attempt to generate a colour image of a Pegasus (a horse with wings). Make a diagram of your final model architecture and discuss its design in your report. You can use any architecture that you like. It is not expected that you will be able to generate a realistic recognisable image of a Pegasus, as this is very difficult task (full marks will be awarded to excellent attempts even if the output is not good). You may submit images of multiple generated Pegasus's, but please highlight your best one that you wish to be marked on.

You will be marked as follows:

- Diagram of final generative model architecture and discussions on its design [**20 marks**].
- Recognisability of output (can I tell this is an image of a Pegasus?) [**10 marks**]
- Realism of output (is the generated image blurry, or does it look like a real image?) [**10 marks**].
- Uniqueness of output (how different is the image from its nearest neighbour in the training/testing dataset?) [**10 marks**].

Base code for generating the pegasus is available here [Pegasus Google Colab Link]. *Note*: you are allowed to significantly deviate from this base code, e.g. interpolating from different labels, using different sampling strategies, or using completely different types of generative architecture.

Constraints

In both of these tasks:

- You can reimplement techniques from academic papers, although it would be preferred if you build your own creative solutions using knowledge learnt from the lectures, and discuss your ideas in the report.
- Do not use pre-trained network weights for either task (transfer learning) - this will score zero.
- Do not use other people's code (existing architectures e.g. on Github). All code is submitted to an automated plagiarism detection service for programming.
- For Part 1, any training on the test set, even if accidental, will be heavily penalised. However this does not matter for Part 2 where you are permitted to train on the test set to generate the Pegasus image.
- Do not paint manual image masks or do any manual image editing. However you are allowed to edit image content automatically in code (e.g. dataset augmentation or image preprocessing).

Submission

Please submit a **.zip** file with your report as a **.pdf**, and two files called 'classifier.py' and 'pegasus.py' (or 'classifier.ipynb' and 'pegasus.ipynb').

Hope you enjoy the coursework - if you are struggling there will be coding workshops on the 7th March to discuss any issues.
