Department of Electrical, Computer, and Software Engineering
University of Auckland
COMPSYS 302 S1 2022 – Python Project (Weighting 40%)

Background

Nowadays, artificial intelligence comes into every aspect of our life. From searching recommendation systems to self-driving cars, the foundation of these applications is Neural Networks. Recently there are a lot of tools and models we can easily use for free, and it becomes core technical skills for most domains. This project will focus on designing and developing an example of AI solution based on a simple design pattern, which can be helpful to your part 4 project.

Objective

We will design and implement a tool that recognizes handwritten digits and English letters. The overall goal of this project is to get familiar with software design, GUI program and usage of basic machine learning skills. Students can find **one** partner to implement this project together. Two students will cooperate by using Github Classroom. Both students should contribute to project design and implementation. The use of Github will also be assessed as well. As you may know, there are lots of codes you can find online. However, we would suggest you **write your own code from scratch except for the recognition methods**.

Project Requirements

All features should be run on GUI

- No command from terminal
- All the procedure and results should be shown via GUI, not PyTorch terminal.

Importing EMNIST dataset (train and test)

- User can import the dataset
- User can see the progress and time left to finish (ie, xx Min, xx Sec left)
- User can stop the progress

Viewer for the dataset

- User can view images of train set or test set
- User can scroll up and down to see the all images (no pagination)
- User can filter out it to see specific character or digit images
- User can see simple statistics of the dataset (the number of each character or digit)

Training a DNN (Deep Neural Network) models using train dataset for classification

- Slide bar to split train dataset into train/validation
- User can select a deep neural networks for training
- User can configure hyper-parameters (batch size, epoch number)
- User can see the training progress (progress bar)

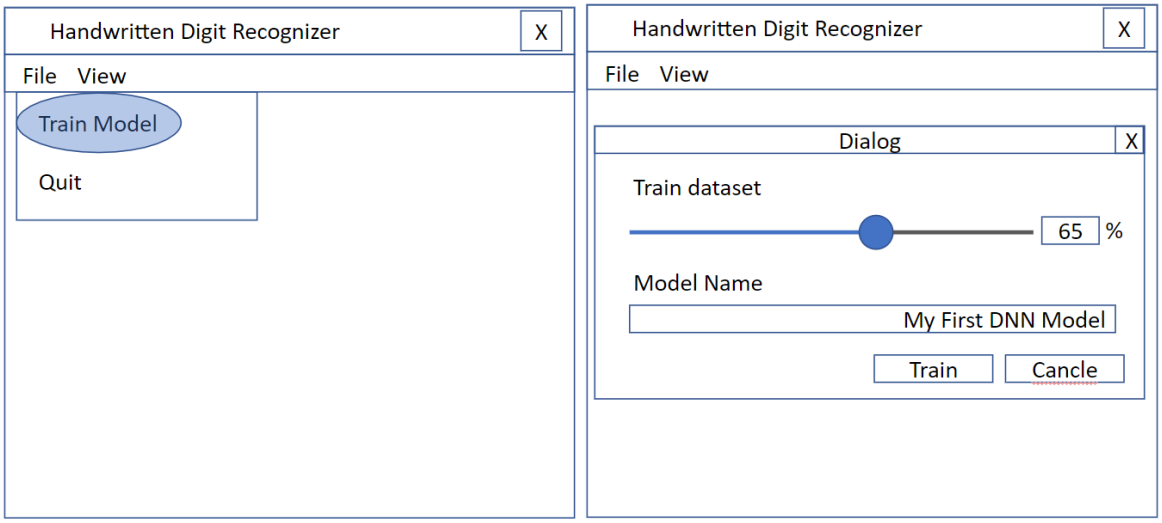
- User can view the training loss and validation accuracy
- A model is saved with training information (train/validation ratio, selected dnn name, batch size, epoch number)
- User can train and save multiple models
- User can stop the training

Prediction

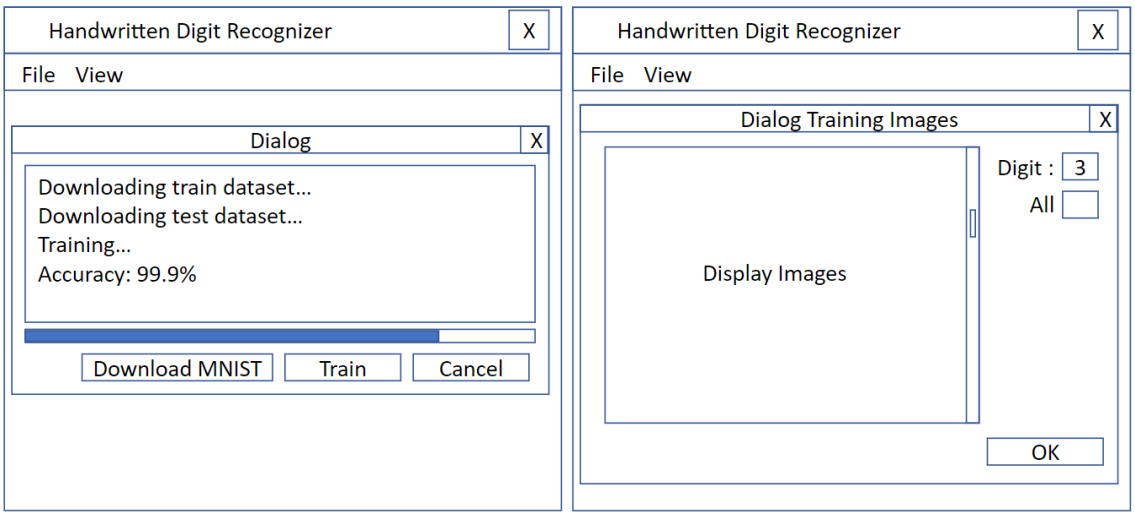
- User can load a trained model among saved models
- User can open test image viewer and choose multiple images for testing
- User can draw a character or digit instead of choosing
- User can see the predicted result (character or digit)
- User can see the accuracy of the prediction

GUI Examples

- These are just examples, so you can design our own GUI.



[For training model]



[For showing progress and results]

Tools / Libraries to use

Python3
For this project, Python3.7 or above is required.

PyQt5 Python GUI
PyQt5 is a GUI widgets toolkit. It is a Python interface for Qt, one of the most powerful, and popular cross-platform GUI libraries. PyQt5 is a blend of the Python programming language and the Qt library. For tutorials, please refer to the following webpage.

- https://wiki.python.org/moin/PyQt/Tutorials

TorchVision for dataset
The EMNIST dataset is a set of handwritten character digits derived from the NIST Special Database 19 and converted to a 28x28 pixel image format and dataset structure that directly matches the MNIST dataset.

- https://pytorch.org/vision/stable/generated/torchvision.datasets.EMNIST.html#torchvision.datasets.EMNIST

PyTorch for DNN
PyTorch is an open-source machine learning library based on the Torch library, used for applications such as computer vision and natural language processing, primarily developed by Facebook's AI Research lab. For the details, please refer to the following website.

- https://pytorch.org/

scikit-image
scikit-image is a collection of algorithms for image processing. But if you are comfortable to use other image manipulation libraries such as PIL or opencv, it is alright to use them. For the details, please refer to the following website.

NumPy
Numpy is the fundamental package for scientific computing with Python. This library should be used for data manipulation, such as an image.

- https://numpy.org/

Code Editor (IDE)
You can use any tools that support Python. Here are the ones you can use, but you can use any tools you want, even not listed here.

- https://code.visualstudio.com/
- https://www.jetbrains.com/pycharm/

Setup Development Environment

Conda and pip will be used to set up the development environment. Detailed instructions will be covered in the lab and/or lecture.

Github

Github Classroom

Each team should create one private repository per group

Directory hierarchy & Code structure

Your python project is recommended to follow a structure as below.

- Name of your project (Repository)
    - requrement.txt (Version of required libraries)
    - README.md (Document for how to install and run your project)
    - main.py (Main code to run your program)
    - scripts (Folder)
        - Your python scripts stay here

Versioning

The use of the Git versioning system via Github is compulsory and will be monitored. Steady progress from all groups is expected throughout the semester, and you should not commit full source code the day before the due date**.**

Requirements

Please provide instructions to run your program correctly.

- Write your instruction on how to install and run your program on README.md,
- Write the details of improvements(or updates) for each version of the program on README.md
- Specify all the dependencies(libraries and versions) on requirement.txt,
- We will check the learning curve to make sure it is learning.

Assessment

This project is completed and assessed in a group as well as individually.

- Submission due: April 24 11:59:59
- [10%] Demo video (Group)
- [10%] Code submission (Group)
- [20%] Final report (Individual)

Demo Video

- https://canvas.auckland.ac.nz/courses/72170/assignments/274848
- Worth 10%
- Type: Group
- Penalty: -1% every 10 mins
- Important points
    - (3%) Presentation
        - Introduction of design
        - Methodology
    - (4%) System development
        - Usefulness/convenience of GUI
        - Key functionalities, such as showing dataset, call method, etc.
        - Completeness
    - (3%) Performance
        - Testing results

Code

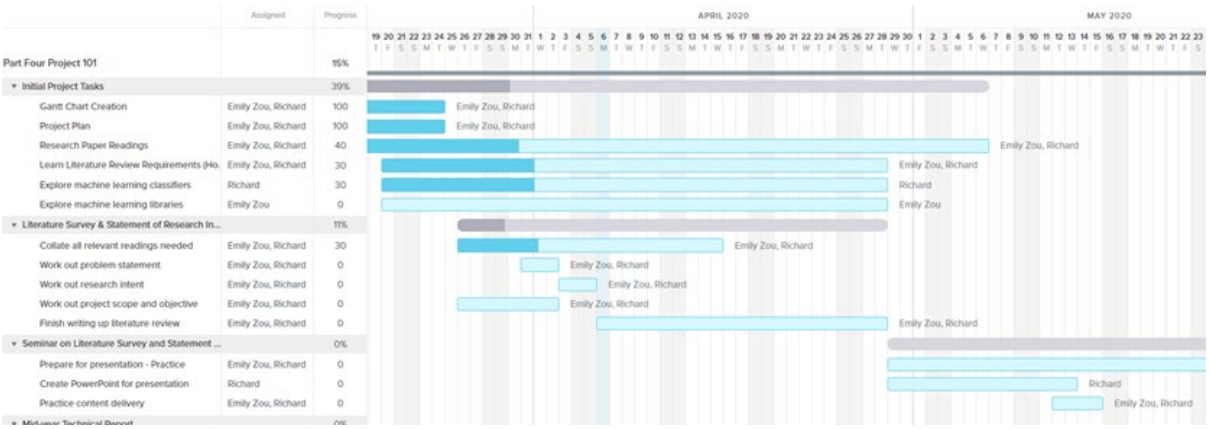
- https://canvas.auckland.ac.nz/courses/72170/assignments/274847
- Worth 10%
- Type: Group
- Penalty: -1% every 10 mins
- Important points
    - You should use Git but must submit the final version as a zip file via CANVAS
- Factors
    - (3%) README file
        - Explain about your project, and versions, etc.
    - (5%) Comments (documentation)
        - Explain your code, including purpose, source (if you get it from somewhere), input, output, etc.
    - (2%) Github history
        - Will check how often you worked on this project.
        - How well you did the version management.

Project Report

- https://canvas.auckland.ac.nz/courses/72170/assignments/274849
- Worth 20%
- Type: Individual
- Penalty: -2% every 10 mins
- Important points

- We combine the design planning document and final report, which were separated in the previous years.
- Your design should be documented in a report which is in the form of a real paper report, which is the same format for your part 4 project reports. There are a few sections that are listed below you need to cover.
- The final report should be max 12 pages. You don't have to write a whole thesis with it. Format should be Doc or pdf.
- You can mention your teamwork and partners here. Please show clear evidence if you complain about your partner.

- Factors
  - (1%) Introduction
    - Imagine you are running a start-up! Image Recognition Group pic.
    - If it is impossible, just show your photo. Individual photos are fine.
    - Can have a logo of your team as well.
    - This is for recognizing who you are.
    - Group name (professional names only!) and group number!
    - Put your group number, UPI, email address
  - (3%) Planning
    - Summary of literature review
    - Setup for your project, such as patterns, environment, tools, etc., and the reason to choose them
    - Gantt chart – schedule and roles



[Example of Gantt chart]

  - (10%) Design Software Architecture
    - Imagine you are writing an application to get fund!
    - Purpose of the system
      - Background
      - Reason to develop your recognizer
      - Benefit to society
      - The goal of your project, i.e. hand-writing recognition
    - Database and methods/model to use
      - Name of database,
      - link of the database you get ==> reference
      - Existing one or your one? How to modify it?
      - How to select and the reason?

            - How does it work?
        - System Diagram for your software
        - Explain each component of your system
        - What is the benefit of your architecture?
    - (4%) Results
        - Describe your results (details of your tool).
        - The statistical analysis: learning result, precision rate, recall rate, f1 for your model.
        - Recognition results
        - Discussion
    - (2%) Conclusion/Future Work
        - Draw your conclusion
        - Future works

Final Note

In this project, you are expected to do your own planning and learning. You'll be responsible for finding out how to do things yourselves. There will be content delivered in lectures to help speed this process up at the beginning, but ultimately it is your responsibility to find out what you will need. If you get stuck and need help, please ask a TA or the lecturer. **Do not leave things to the last minute**. The easiest way to fail this project is to try to start it in the last week (or night) before it's due.

Do not mistake the minimum requirements as the only requirements; if you want to do well in this course, you will need to design your project carefully and manage your time well. Do not hesitate to contact the lecturers and TAs if you are in doubt! They are available to help with concepts, troubleshooting, and debugging, but they will not write your project.

Poorly documented or implemented code, while otherwise functionally correct, may not get you full marks for the project overall. Someone else has to be able to work on your code later! Please develop consistent and good coding practices.

Due to the nature of this project, a certain level of collaboration is required within the class. As a result, it is strongly recommended that you interact with other students in the labs. Additionally, TAs will be there, and you can discuss requirements with them as well.

Academic Integrity Notice

The University of Auckland will not tolerate cheating or assisting others to cheat and views cheating in coursework as a serious offence. The work that a student submits for grading must be the student's work, reflecting his or her learning. Where work from other sources is used, it must be appropriately acknowledged and referenced. This requirement also applies to sources on the world-wide-web. **Do not copy code from other students or the internet** without attribution.