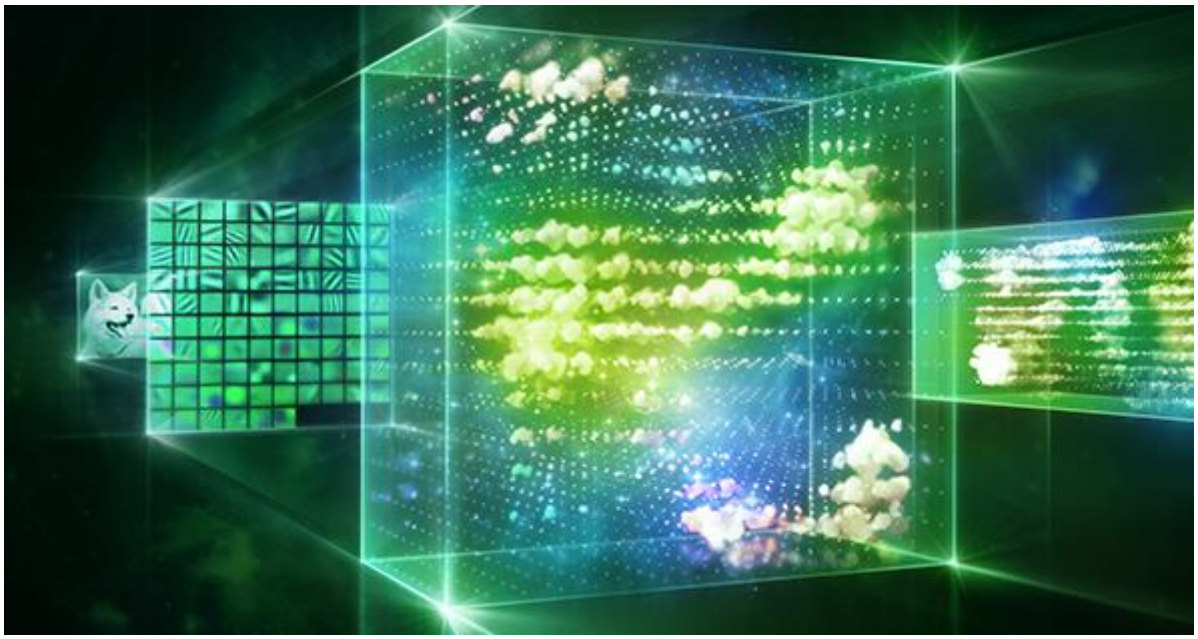
School Name	School of Computing
Semester	AY2020/21 Semester 2
Course Name	DIT (DSDA)
Module Code	ST1504
Module Name	Deep Learning



Assignment 1 (CA1: 40%)

The objective of the assignment is to help you gain a better understanding of deep learning for image classification using Convolutional Neural Networks (CNN). You may also wish to take on the additional task of applying suitable deep learning techniques to real-world data (bonus task).

Guidelines

1. You are to work on the main problem set individually. You may wish to work on the bonus task in groups of 2-3 persons.

2. In this assignment, you will create a CNN for image classification and evaluate the performance of the network. You must perform the necessary steps to improve the model performance.

3. Write a Jupyter notebook including your code and comments and visualizations. In addition, please save a copy of the jupyter notebook as a html file. Create a presentation file for your project. Submit your Jupyter notebook, data and the slides in a compressed package (zip file).

4. Students are required to submit their assignment using the assignment link under the Assignment folder. Please remember to include your student name and student admission number on the top of your jupyter notebook and in the title slide of your PowerPoint presentation.

5. The normal SP's academic policies on Copyright and Plagiarism applies. Please note that you are to cite all sources. You may refer to the citation guide available at: http://eliser.lib.sp.edu.sg/elsr_website/Html/citation.pdf

Submission Details

Deadline: 2020-11-27 23:59H
Submit through: Blackboard

Late Submission

50% of the marks will be deducted for assignments that are received within ONE (1) calendar day after the submission deadline. No marks will be given thereafter.
Exceptions to this policy will be given to students with valid LOA on medical or compassionate grounds. Students in such cases will need to inform the lecturer as soon as reasonably possible. Students are not to assume on their own that their deadline has been extended.

PART A: CONVOLUTIONAL NEURAL NETWORK (100 marks)

Background

Implement an image classifier using a deep learning network. [Hint: You may wish to refer to papers on successful DL architectures such as AlexNet]

Dataset

You are to use the MNIST dataset.

Tasks

1. Write the code to solve the prediction task. You would be using TensorFlow 2.0/Keras, but if you'd prefer to work with some other toolkit such as MXNET or PyTorch, that is fine.
2. Write a Jupyter notebook detailing your implementation, your experiments and analysis. Remember to also save the jupyter notebook as a HTML file after running it. Here are some things to include in your jupyter notebook:

    - How is your prediction task defined? And what is the meaning of the output variable?
    - How is the data represented as features?
    - Did you process the features in any way (e.g. normalization)?
    - How did you select which CNN architecture to use?
    - How do you evaluate the quality of your system?
    - Can you say anything about the errors that the system makes? You may consider a confusion matrix.
    - Please show how you have improved the model from the initial performance, e.g. by hyperparameter turning, by modifying the network architecture, or by introducing regularization using Dropout, etc.
    - Provide a reference section for any papers, online articles, books, publications that you may have referenced.

3. Create a set of slides with the highlights of your Jupyter notebook. Explain the entire deep learning process you went through, including data exploration, data cleaning, feature engineering, and model building and evaluation. Write your conclusions.

Submission requirements for Part A

1. Submit a zip file containing all the project files (Jupyter notebook and HTML file) and the slides (PPTX or pdf).
2. Submit online via the Assignment link.

Evaluation criteria:

Background research	10%
Suitable feature selection/engineering	10%
Suitable model building	20%
Suitable evaluation of model	20%
Model Improvement	20%
Presentation/Demo	10%
Quality of report (Jupyter)	10%

PART B: DEEP LEARNING WITH REAL WORLD DATA (BONUS 20 marks)

Introduction

Use a suitable deep learning architecture on a real-world data set of substantial size. For example, the Grab Safety challenge dataset. You are to work in teams of 2-3 person. Only the team leader needs to submit. Please include the full names of students of all the team members in the notebook, title side, and title screen of video.

Dataset

You are to use the Grab Safety challenge dataset.
[Access URL: https://github.com/yingjia-liu/Msft-Grab-FRS/blob/master/Safety/Safety_Problem_Statement.md]

Tasks

1. Write the code to solve the prediction task. You would be using TensorFlow 2.0/Keras, but if you'd prefer to work with some other toolkit such as MXNET or PyTorch, that is fine too.
2. Write a Jupyter notebook detailing your implementation, your experiments and analysis. Remember to also save the jupyter notebook as a HTML file after running it.
3. Create a set of slides with the highlights of your Jupyter notebook. Explain the entire deep learning process you went through, data exploration, data cleaning, feature engineering, and model building and evaluation. Write your conclusions. Create a video presentation of your slides (i.e. with face and voice over) and save it as a mp4 file.

Submission requirements for Part B

1. Submit a zip file containing all the project files (Jupyter notebook and HTML file) the slides and mp4 video file.
2. Submit a docx file containing the list of specific contributions by each team member in the deliverables for part B (e.g. Did background research into the inertia moment unit sensor).
3. Submit online via the Assignment link.

— End of Assignment —