Programming Assignment-2
CSL7590: Deep Learning (Fractal-2)
MTech-ExDCS Program
Due on: July 03, 2023
Max. Marks: 350 + 50 (Bonus)

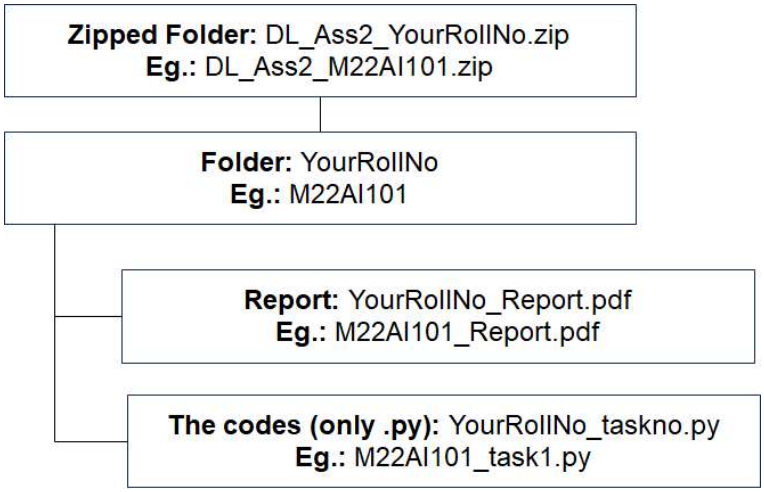
General Instructions:

1. This is an individual activity.
2. Clearly, mention the assumptions you have made, if any.
3. Clearly, report any resources you have used while attempting the assignment.
4. Any submission received in another format or after the deadline will not be evaluated.
5. Make sure to add references to the resources that you have used while attempting the assignment.
6. Plagiarism of any kind will not be tolerated and will result in zero marks.

Submission Guidelines:

1. Prepare separate Python code files for each task and name them YourRollNo_task1.py, YourRollNo_task2.py, and so on.
2. Also, provide your colab file link in the report. Make sure that the file is sharable.
3. Put both the codes and a report in a folder named <YourRollNo>, create a zip folder named DL_Ass2_YourRollNo.zip, and upload to google-classroom. See attached image to get better clarity.
4. **Do not** download the .ipynb file; rename it as .py, and upload it. .ipynb files are not exactly in a readable form, so uploading it will only result in you receiving 0 marks for the same. You have the option to download the .py file in google colab. Use it to get the .py format.
5. Submit a single report depicting all tasks' methods, results, and observations. There is no need to add theory behind the concepts. Preparing a report is mandatory; failing it will lead to non-evaluation of the assignment.
6. **Do not** copy-paste code or screenshots, etc., in the report. The report should look like a technical document, containing plots, tables, etc., whenever necessary.

PLEASE NOTE: NOT ADHERING TO SUBMISSION GUIDELINES WILL RESULT IN 0 MARKS BEING AWARDED TO THE STUDENT.



Task1: Training masked autoencoder on PASCAL VOC 2007 dataset [200 marks + 30 marks (bonus)]

1. Dataset preparation: [5+5 marks]
    a. Download the PASCAL VOC 2007 dataset from here.
    b. Preprocess the dataset by resizing the images to a fixed size and normalizing the pixel values.
2. Dataset split: (Run Task.1.3(a) with both the following splits and choose the best one for Task.1.3(b), Task.1.3(c) and next experiments.)
    a. Use 80-10-10, train-val-test split [10 marks]
    b. Use 70-10-20, train-val-test split [10 marks]
3. Architecture: An autoencoder with three hidden layers with the following bottleneck dimension (on denoising task, add gaussian noise to the input image, and the output should be denoised image)
    a. 256 (run using both the splits in 2, and choose the best one for the further set of experiments) [20 marks]
    b. 128 [10 marks]
    c. 64 [10 marks]
    d. 32 [10 marks]
    e. 16 [10 marks]
4. Choose the best bottleneck dimension, and re-run the autoencoder using masking strategy: (mask the following % of pixels in the image, i.e., set the pixel value to (0,0,0)) [20*4 = 80 marks]
    a. 20%
    b. 40%
    c. 60%
    d. 80%.

5. Plot reconstruction error for every autoencoder model. [10 marks]
6. Evaluation: Report MSE (mean square error), MAE (mean absolute error) for all models. Comment on which metric is more useful in judging the quality . [10 marks]
7. Visualize and compare the original images, masked images, and reconstructed images. [10 marks]
8. Use any other metric of your choice (apart from MSE, MAE) to judge the image quality. **[Bonus: 30 marks]**
9. Use the best split, and best masking strategy (with best encoding dimension) for Task 2.

Task2: Fine-tuning a pre-trained autoencoder on STL-10 dataset [110 marks + 20 marks (bonus)]

1. Dataset: Download the STL-10 dataset from here and pre-process as required. [5+5=10 marks]
2. Use the encoder of the above-pretrained autoencoder (**Refer to task 1.9**) as a feature extractor. [15 marks]
3. Build a downstream task classifier (a MLP) with (100% training samples) and choose the best one among a, and b for Task 2.4. [20 marks]
    a. # hidden_layer = 3
    b. # hidden_layer = 5
4. Use the best from Task 2.3.a and 2.3.b, fine-tune the classifier on the STL-10 dataset with the following % of training samples [5 * 10 marks = 50 marks]
    a. 1%
    b. 10%
    c. 20%
    d. 40%.
    e. 60 %
5. Prepare confusion matrix and AUC-ROC curve to evaluate the model performance (for Task 2.3.(a-b) and Task 2.4.(a-e)). [15 marks]
6. Implement a different architecture than the architecture described in Task.2.3.(a-b), that improves the result. [**Bonus: 20 marks**]

Task3: Report [40 Marks]

1. Report hyperparameters clearly for every model [15 marks]
    a. No. of epochs
    b. Learning rate

    c. Optimizer
    d. Loss function.
    e. Dimensions
2. Report and justify your choice of best (for split, for bottleneck dimension, masking strategy etc.) [15 marks]
3. Observations about results, any additional analysis over results. [10 marks]