NEURAL NETWORKING AND DEEP LEARNING – ICP 8

PRAVEENA GOLI

700743010

Use Case Description:

LeNet5,

AlexNet,

Vgg16,

Vgg19

1. Training the model

2. Evaluating the model Programming elements:

1. About CNN

2. Hyperparameters of CNN

3. Image classification with CNN In class programming:

1. Tune hyperparameter and make necessary addition to the baseline model to improve validation accuracy and reduce validation loss.

2. Provide logical description of which steps lead to improved response and what was its impact on architecture behavior.

3. Create at least two more visualizations using matplotlib (Other than provided in the source file)

4. Use dataset of your own choice and implement baseline models provided.

5. Apply modified architecture to your own selected dataset and train it.

6. Evaluate your model on testing set.

7. Save the improved model and use it for prediction on testing data

8. Provide plot of confusion matric

9. Provide Training and testing Loss and accuracy plots in one plot using subplot command and history object.

10. Provide at least two more visualizations reflecting your solution.

11. Provide logical description of which steps lead to improved response for new dataset when compared with baseline model and enhance architecture and what was its impact on architecture behavior.

UPLOADED FILES IN GIT HUB FOR THE ABOVE THINGS

GIT HUB LINK: https://github.com/Goli18/NNDL_ICP8.git

VIDEO LINK :
https://github.com/Goli18/NNDL_ICP8/blob/main/NN%26DL_ICP8.pdf%20and%201%20more%20page%20-%20Personal%20-%20Microsoft%E2%80%8B%20Edge%202023-03-29%2023-30-02.mp4