ASSIGNMENT-2

Individual Assignment (100 points)

Instructions:

- Submit the paper review as a word or pdf file.
- Submit elegant code with substantial comments as a Python notebook along with the HTML version.
- You are expected to work independently. Plagiarism in any form will not be tolerated.
- You may take selective help from AI models to solve this assignment.

1. Paper Review – Select and review a technical paper from the list of papers in Group 2 (20)

2. Train two different FFN models using *Keras*: (20)

   - A model to learn the Sine function in the range ($-\pi$, π) and the second to learn another arbitrary but non-trivial function of your choice.
   - Plot the learned mapping for a given range (π, 2π) and compare it with the Sine wave for the first model and a suitable range for the second model.
   - Summarize the model architectures for both models.
   - Neural networks are known to be "Universal Function Approximators". Would you agree with the statement based on this experiment?

3. Build a CNN model using Keras to classify the IP 102 dataset with the given train-test-validation split (60).
   a) Perform exploratory data analysis and data cleaning as necessary.
   b) Filter out (ignore) classes with less than 200 samples in the trainset.
   c) Build a Convolutional Neural Network (CNN) using *Keras*
      - Apply call backs such as Early Stopping and Learning Rate Scheduling
      - Hyperparameter tuning with an optimizer of your choice.
      - Use *Tensorboard* to analyze the model graph and metrics.
   d) Make the following modifications to your architecture and study the impact on model performance.
      - Increase the number of neurons.
      - Add more hidden layers.
      - Add Batch Normalization in one or more layers.
      - Add L1/L2 regularization in one or more layers.
      - Add Dropout in one or more layers.
      - Experiment with at least two types of activation functions.

      Provide comparative loss/accuracy tables and plots to track your experiments and model convergence.
   e) Persist the trained model and perform final evaluation using a model loaded from disk.
   f) Provide a manual calculation for the total number of parameters used in your neural network and verify the results with Keras *summary().*