

Look, Ask, Explain: The VQA Challenge

Week 1 Assignment

1. Use **DenseNet-121** (pretrained on ImageNet) in **feature extraction mode** for **CIFAR-10** classification, where you will:
 - Freeze all the convolutional Layers
 - Replace the final classifier layer.
 - Train only the classifier
 - Evaluate the performance using the validation dataset.

Use the CIFAR 10 Dataset, which is a PyTorch built-in dataset.

2. Build and train a **Deep Neural Network (No CNN)** that classifies **MNIST digits**. You will:
 - Load and preprocess the MNIST dataset.
 - Flatten them.
 - Construct a multilayer NN.
 - Train and test the model on the Training and Validation set.

Tips to Improve Accuracy:

- **Early Stopping:** stop training when validation loss stops improving, prevents overfitting, and saves time.
- **Learning rate Scheduling**
- **Dropout:** Add dropout (e.g., $p = 0.2\text{--}0.5$) between layers to reduce overfitting.
- **Data Augmentation (Optional):** MNIST is grayscale, but you can try:
 - Random rotation
 - Random shift

This sometimes helps generalisation. CIFAR-10 is small, so augmentations help a lot.

- **Add Batch Normalisation Before the Classifier:** you can add BatchNorm to the classifier head for Part1, as BN stabilises gradients and usually improves accuracy by 1–3%.

Final Deliverables:

- *.ipynb or .py* code file
- Training & Validation loss curves.
- Final Test Accuracy, Confusion Matrix.
- and some small observations like what worked or what didn't.

So roll up your sleeves — it's time to get your hands dirty and build your own network!

Happy Learning :)