**Image Classification using Deep Learning**

**Dataset**

The dataset consists of images in two parts: part1 and part2. The part1 directory contains two subdirectories: train and test. The train directory contains subdirectories for each class and the **test** directory contains unlabeled images. The part2 directory contains only images and annotations are provided in a separate annotations.csv file.

**Environment Setup**

1. Install PyTorch and torchvision by following the instructions on the PyTorch website.
2. Install other necessary packages using pip: pip install pandas ,scikit-learn

**Model Architecture**

I have ResNet-50 pre-trained model as the base model and fine-tune the last fully connected layer for our task. We will add a new fully connected layer with a softmax activation function that outputs the probability distribution over the classes.

**Data Preprocessing**

1. Resize the images to a fixed size of 224x224 pixels.
2. Randomly flip the images horizontally for data augmentation.
3. Convert the images to PyTorch tensors.
4. Normalize the image tensors using the mean and standard deviation of the ImageNet dataset.

**Training**

Train the model using the cross-entropy loss function and the Adam optimizer with a learning rate of 0.001. We will train the model for 50 epochs with a batch size of 16. We will monitor the loss on the training set and the mean average precision on the validation set during training.

**Evaluation**

I tried to evaluate the model on the test set and report the accuracy and other relevant metrics.