**NAME: Ashish Dnyaneshvar Parulekar**

**STUDENT ID: 1017913240**

**COURSE: Advanced Computer Vision (CPS-667)**

**Assignment 1**

**TITLE: MNIST & CIFAR-100 Dataset Processing using PyTorch**

**Introduction**

The purpose of this project is to acquaint people with utilizing PyTorch to handle picture datasets. Its main objectives are to download, load, partition, visualize, and extract data from two well-known datasets: CIFAR-100 (100 object categories) and MNIST (handwritten digits). For deep learning models to be developed and trained effectively, this procedure is necessary.

**Steps Implemented**

**Step 1: Download the Datasets**

* The MNIST dataset (handwritten digits 0-9) and CIFAR-100 dataset (100 image classes) are downloaded using torchvision.datasets.
* The datasets are stored in the ./data directory.

**Step 2: Load the Datasets into Memory**

* PyTorch loads the datasets into memory automatically, and the script outputs how many training and test samples there are in each dataset.

**Step 3: Print Dataset Information**

* The script prints the dataset properties:
  + Number of training samples
  + Image dimensions (MNIST: 28x28, CIFAR-100: 32x32)
  + Number of classes (MNIST: 10, CIFAR-100: 100)

**Step 4: Split the Training Dataset**

* The training datasets are split into:
  + **MNIST:** 48,000 training samples, 12,000 validation samples
  + **CIFAR-100:** 40,000 training samples, 10,000 validation samples
* PyTorch's random\_split() function is used to achieve this.

**Step 5: Display an Image from Both Datasets**

* The script defines a function disp(dataset, n, dataset\_name) that displays an image from the dataset.
* The matplotlib library is used for visualization.
* The function is called for both MNIST and CIFAR-100.

**Step 6: Print Out a Training Sample’s Label**

* A function display\_label(dataset, n, dataset\_name) is implemented to print the label of an image.
* Labels are retrieved using PyTorch’s dataset class.
* The label of the same image (index 97) is printed for both MNIST and CIFAR-100.

**How to Run the Code**

1. Install dependencies if not already installed:

pip install torch torchvision matplotlib

1. Run the script in a Jupyter Notebook.
2. Ensure that images and labels match in the output.
3. Check the output to make sure the pictures and labels match what should be there.

**Expected Output**

1. **Dataset Information**

MNIST dataset loaded with 60,000 training samples and 10,000 test samples.

CIFAR-100 dataset loaded with 50,000 training samples and 10,000 test samples.

1. **Dataset Properties**

MNIST Dataset: Number of training samples: 60,000 Image dimensions: 28x28 Number of classes: 10

CIFAR-100 Dataset: Number of training samples: 50,000 Image dimensions: 32x32 Number of classes: 100

1. **Training & Validation Split**

MNIST Training set: 48,000 samples, Validation set: 12,000 samples

CIFAR-100 Training set: 40,000 samples, Validation set: 10,000 samples

1. **Image Display**

A screenshot of a computer

AI-generated content may be incorrect.

Two images will be displayed: one from MNIST and one from CIFAR-100.

1. **Label Output**

Sample 9 label: '4 - four' (MNIST)

Sample 9 label: 'elephant' (CIFAR-100)

A screenshot of a computer

AI-generated content may be incorrect.

**Conclusion**

The basic processes of managing datasets in PyTorch, such as downloading, loading, dividing, displaying, and extracting data from the MNIST and CIFAR-100 datasets, are well demonstrated in this project. Users that adhere to this approach acquire crucial knowledge in getting datasets ready for deep learning applications. Effective model training and evaluation are made possible by the structured methodology, which guarantees seamless data management.