Module 4: Critical Thinking

Pytorch Demo

Ryan Thompson

Colorado State University - Global

CSC 525

Dr. Banerjee

11 May 2025

**PyTorch Convolutional Neural Network on MNIST**

The installation process for PyTorch on a Linux Debian system using the pip package manager was smooth and issue-free. The framework and its dependencies installed correctly without requiring additional configuration or troubleshooting. Notably, the PyTorch installation was straightforward, and the example script executed successfully on the first attempt, confirming the environment was properly set up.

The chosen example is a Convolutional Neural Network (CNN) used for image classification on the MNIST dataset. Upon execution, the script trained the CNN over 14 epochs and evaluated its performance on a test set. The training process was clearly logged, and accuracy improved steadily across epochs. The model achieved high accuracy on the MNIST dataset, which consists of grayscale images of handwritten digits.

The example employs a Convolutional Neural Network, a model particularly effective for visual data due to its ability to automatically extract hierarchical features from input images. CNNs use convolutional layers to scan input data and detect local patterns such as edges and shapes, which makes them well-suited for image classification tasks. CNNs are commonly used in facial recognition, medical image analysis, autonomous driving, and more.

The techniques demonstrated in the example, such as the use of dropout for regularization and ReLU activation for non-linearity are applicable to a portfolio project involving computer vision or time-series data if appropriately adapted. The modular nature of the model and the training pipeline makes it easy to integrate enhancements such as batch normalization or data augmentation.

The dataset used, MNIST, is a classic benchmark in machine learning but relatively simplistic by modern standards. Improvements could include applying more complex datasets like Fashion-MNIST or CIFAR-10 for greater challenge and relevance. Additionally, the model could be expanded with deeper architectures, residual connections, or transferred to mobile inference applications using frameworks like TorchScript.

Finally, the model used is a clear example of supervised learning, where the model learns to map input images to predefined labels through labeled training data. This approach is ideal for classification tasks with clearly labeled datasets, as demonstrated in this project.

**References**

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.<https://doi.org/10.1109/5.726791>

PyTorch. (n.d.). *PyTorch documentation*.<https://pytorch.org/docs/>