PyTorch: Deep Learning from Basics to Advanced
By Dr. Vishwanath Rao

Overview:
This comprehensive course provides an in-depth understanding of PyTorch, a powerful open-source machine learning framework, from fundamental concepts to advanced techniques. Participants will learn to leverage PyTorch for building and training deep learning models, exploring various applications such as image classification, natural language processing, and reinforcement learning.

This class is for (audience):
- Data scientists and machine learning engineers seeking to master PyTorch for deep learning projects.
- Software developers interested in implementing deep learning algorithms with PyTorch.
- Researchers exploring advanced techniques in neural networks and deep learning.
- Professionals looking to enhance their skills in deep learning frameworks and applications.

Prerequisites:
- Basic knowledge of Python programming language.
- Familiarity with machine learning concepts such as supervised learning and neural networks.
- Some experience with deep learning frameworks is beneficial but not required.

Duration: 5 days (40 hours)

Objectives:
- Gain a solid understanding of deep learning principles and neural networks.
- Learn to use PyTorch for building and training neural networks.
- Explore advanced techniques such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) with PyTorch.
- Understand best practices for model evaluation, optimization, and deployment in PyTorch.
- Gain hands-on experience through coding exercises and projects covering various deep learning applications.

What you will learn:

Day 1: Introduction to PyTorch and Deep Learning Basics

- Overview of PyTorch: features, advantages, and ecosystem.
- Installing PyTorch and setting up the development environment.
- Understanding tensors and basic operations in PyTorch.
- Building and training a simple neural network with PyTorch.
- Introduction to gradient descent and backpropagation.
- Loss functions and optimization algorithms in PyTorch.
- Model evaluation and performance metrics.
- Introduction to overfitting and regularization techniques.
- Introduction to PyTorch data handling utilities: Dataset and DataLoader.
- Hands-on: Implementing a basic neural network for classification tasks.

Day 2: Deep Dive into Neural Networks with PyTorch

- Understanding different types of neural networks: feedforward, convolutional, recurrent, and more.
- Building convolutional neural networks (CNNs) with PyTorch for image classification tasks.
- Implementing transfer learning with pre-trained CNN models.
- Introduction to recurrent neural networks (RNNs) and their applications.
- Building and training RNNs for sequence modeling tasks.
- Handling sequential data with PyTorch: sequences, batches, and padding.
- Exploring attention mechanisms for sequence-to-sequence tasks.
- Introduction to generative adversarial networks (GANs) and their implementation in PyTorch.
- Hands-on: Building a CNN for image classification and an RNN for text generation.

Day 3: Advanced Techniques in PyTorch

- Introduction to autoencoders and variational autoencoders (VAEs).
- Implementing autoencoders with PyTorch for dimensionality reduction and anomaly detection.
- Introduction to reinforcement learning and its applications.

- Building reinforcement learning agents with PyTorch and OpenAI Gym.
- Exploring policy gradients and Q-learning algorithms.
- Introduction to deep reinforcement learning architectures: Deep Q-Networks (DQNs), Deep Deterministic Policy Gradient (DDPG), and Proximal Policy Optimization (PPO).
- Implementing deep reinforcement learning algorithms with PyTorch.
- Hands-on: Building a variational autoencoder (VAE) for image generation and implementing a deep reinforcement learning agent.

Day 4: Model Optimization and Deployment

- Techniques for model optimization and performance improvement.
- Introduction to model pruning, quantization, and compression techniques.
- Exploring distributed training with PyTorch using DataParallel and DistributedDataParallel.
- Introduction to model deployment strategies with PyTorch.
- Converting PyTorch models to production-ready formats: ONNX, TorchScript.
- Introduction to model deployment platforms: TorchServe, ONNX Runtime, TensorFlow Serving.
- Deployment considerations: performance, scalability, and monitoring.
- Hands-on: Optimizing and deploying a PyTorch model for inference.

Day 5: Advanced Topics in PyTorch

- Introduction to PyTorch Lightning: a lightweight PyTorch wrapper for high-performance training.
- Exploring PyTorch Hub: a repository for pre-trained models and components.
- Introduction to PyTorch Geometric for graph neural networks (GNNs).
- Implementing graph neural networks with PyTorch Geometric.
- Exploring advanced optimization techniques: learning rate scheduling, momentum, and adaptive optimizers.
- Introduction to PyTorch extensions and custom operators with C++.
- Best practices for organizing PyTorch projects and code.
- Hands-on: Exploring PyTorch Lightning, PyTorch Hub, and implementing a graph neural network with PyTorch Geometric.