Faysal Aziz

5-Day Course Outline: Mastering PyTorch for Deep Learning Applications

FA-0481

Agentic & Generative AI · Software Development · DevOps, Cloud & Infrastructure · Data & Analytics

Introduction

Discover the power of deep learning with our comprehensive 5/10-day course, Mastering PyTorch for Deep Learning Applications. As artificial intelligence continues to revolutionize industries worldwide, mastering PyTorch, a leading open-source deep learning framework, has become an essential skill for professionals in data science, machine learning, and AI development. This course provides a hands-on approach to learning the latest techniques and architectures in deep learning, empowering you to create cutting-edge applications that can transform businesses, healthcare, transportation, entertainment, and more.

Our carefully designed course outline ensures a solid foundation in key concepts, including various neural network architectures, computer vision, natural language processing, reinforcement learning, model interpretability, optimization, and deployment. Learn from experienced instructors who bring real-world insights and industry case studies to the classroom, preparing you to tackle the most pressing challenges in AI today. By the end of the course, you'll master the skills needed to train data, build models using the PyTorch framework, modify existing models for custom applications, and deploy trained models in C++ environments with ONNX. Don't miss this opportunity to accelerate your career and contribute to the exciting advancements in AI and deep learning.

Learning Outcome

Upon completing the first 5 days of the Mastering PyTorch for Deep Learning Applications course, participants will be able to:

- Understand the fundamentals of deep learning and the PyTorch framework.
- Efficiently work with tensors and use automatic differentiation for gradient computation.
- Implement linear regression, perceptrons, and deep neural networks with PyTorch.

- Build and train convolutional neural networks (CNNs) for image classification tasks.
- Apply transfer learning techniques to fine-tune pre-trained models for custom datasets and tasks.
- Utilize the Detectron2 library to implement object detection models in PyTorch.
- Perform semantic segmentation using popular architectures like U-Net and SegNet.
- Understand and implement transformer-based models for natural language processing tasks.
- Gain experience with generative adversarial networks (GANs) for image synthesis.
- Convert PyTorch models to ONNX format and deploy them in C++ environments.
- Apply the skills and knowledge acquired in the course for practical applications and problem-solving in deep learning projects.

Prerequisites

Before attending the Mastering PyTorch for Deep Learning Applications course, learners should have the following prerequisites:

- Basic programming skills: Learners should be comfortable with programming in Python, including understanding and using variables, loops, conditional statements, functions, and basic data structures like lists and dictionaries.
- Familiarity with libraries: Experience with using Python libraries such as NumPy and Matplotlib for data manipulation and visualization will be beneficial.
- Mathematical background: A basic understanding of mathematics, particularly linear algebra (vectors, matrices, and operations), calculus (derivatives and gradients), and probability and statistics (probability distributions, mean, and standard deviation) is essential for understanding deep learning concepts.
- Machine learning fundamentals: While not strictly necessary, prior knowledge of machine learning concepts, such as supervised and unsupervised learning, training and testing datasets, and evaluation metrics, will help learners grasp deep learning topics more quickly.
- Basic knowledge of neural networks: A general understanding of neural networks, including the concept of neurons, layers, and activation functions, can help learners hit the ground running when diving into deep learning architectures and algorithms.While these prerequisites provide a solid foundation for the course, learners who might not meet all the requirements can still benefit from the course by investing additional time in self-study and practice. Online resources and tutorials can help learners fill knowledge gaps and get up to speed with the necessary background information.

Training outline

1. Day 1: Introduction and Basics

Introduction to PyTorch (1 hour)

- What is PyTorch?

- Comparison with other frameworks (TensorFlow, Keras, etc.)
- PyTorch installation and setup

PyTorch Tensors (2 hours)

- What are tensors?
- Tensor operations and properties
- Tensor gradients and automatic differentiation
- Practical exercises: tensor manipulation

Linear Regression with PyTorch (2 hours)

- Overview of linear regression
- Loss functions and optimization
- Implementing linear regression with PyTorch
- Practical exercise: predicting house prices

2. Day 2: Shallow and Deep Neural Networks

Perceptrons (1.5 hours)

- Introduction to perceptrons
- Activation functions
- Implementing a perceptron with PyTorch
- Practical exercise: binary classification

Deep Neural Networks (2.5 hours)

- Introduction to deep learning
- Feedforward neural networks
- Backpropagation algorithm
- Implementing deep neural networks with PyTorch
- Practical exercise: handwritten digit recognition

3. Day 3: Computer Vision with PyTorch

Image Classification using PyTorch CNNs (2 hours)

- Introduction to convolutional neural networks (CNNs)
- CNN architectures
- Implementing a CNN with PyTorch
- Practical exercise: image classification on CIFAR-10 dataset

Transfer Learning (1.5 hours)

- What is transfer learning?

- Fine-tuning and feature extraction
- Using pre-trained models in PyTorch
- Practical exercise: transfer learning for custom image classification

Object Detection using Detectron2 (1.5 hours)

- Introduction to object detection
- Detectron2 overview
- Implementing object detection with Detectron2 and PyTorch
- Practical exercise: object detection on COCO dataset

4. Day 4: Advanced Computer Vision and NLP

Semantic Segmentation (2 hours)

- Introduction to semantic segmentation
- Popular architectures (U-Net, SegNet, etc.)
- Implementing semantic segmentation with PyTorch
- Practical exercise: semantic segmentation on Cityscapes dataset

Transformers (2 hours)

- Introduction to transformers
- Self-attention mechanism
- Implementing transformers with PyTorch
- Practical exercise: text classification with transformers

5. Day 5: Model Deployment

Generative Adversarial Networks (2 hours)

- Introduction to generative adversarial networks (GANs)
- GAN architectures and training
- Implementing GANs with PyTorch
- Practical exercise: generating synthetic images with GANs

Convert PyTorch model to ONNX and running in C++ (2 hours)

- Introduction to ONNX
- Converting PyTorch models to ONNX format
- Running ONNX models in C++ applications
- Practical exercise: deploy a PyTorch model in a C++ environment

Course Recap and Next Steps (1 hour)

- Review of key concepts and techniques

- Exploring further resources and advanced topics
- Q&A and course feedback

Extended 10-Day Course Outline: Mastering PyTorch for Deep Learning Applications

The extended 10-day course outline provides a comprehensive overview of mastering PyTorch for deep learning applications. This outline covers essential topics in deep learning, including various neural network architectures, computer vision, natural language processing, reinforcement learning, model interpretability, optimization, and deployment. The course also discusses industry applications and case studies to provide participants with practical insights.

Learning Outcome

Upon completing the extended 10-days, participants will be able to:

- Understand the fundamentals of deep learning and the PyTorch framework.
- Efficiently work with tensors and use automatic differentiation for gradient computation.
- Implement various neural network architectures, including feedforward networks, CNNs, RNNs, LSTMs, GRUs, and transformers.
- Build and train models for diverse applications, such as image classification, object detection, semantic segmentation, natural language processing, and reinforcement learning.
- Apply transfer learning techniques to fine-tune pre-trained models for custom datasets and tasks.
- Utilize advanced deep learning concepts like GANs, autoencoders, graph neural networks, and multi-modal learning.
- Interpret and analyze the predictions of deep learning models using interpretability techniques and tools like PyTorch Captum.
- Optimize and prune models to improve performance and reduce computational resources.
- Convert PyTorch models to ONNX format and deploy them in C++ environments.
- Deploy trained models on mobile devices using PyTorch Mobile.
- Gain insights into real-world industry applications and case studies, preparing for the challenges in deploying AI solutions across various domains.
- Collaborate effectively with other AI and machine learning professionals, leveraging the skills and knowledge acquired in the course for practical applications and problem-solving.

Extended Additional Outline

6. Day 6: Recurrent Neural Networks and Sequence-to-Sequence Models

Recurrent Neural Networks (RNNs) (2 hours)

- Introduction to RNNs

- Types of RNNs: Simple RNN, LSTM, and GRU
- Implementing RNNs with PyTorch
- Practical exercise: text generation with RNNs

Sequence-to-Sequence Models (2 hours)

- Introduction to seq2seq models
- Encoder-decoder architecture
- Attention mechanisms
- Implementing seq2seq models with PyTorch
- Practical exercise: neural machine translation

7. Day 7: Reinforcement Learning and Multi-Modal Applications

Introduction to Reinforcement Learning (1.5 hours)

- Overview of reinforcement learning
- Q-learning and Deep Q-Networks (DQN)
- Implementing DQN with PyTorch
- Practical exercise: training an agent to play a simple game

Multi-Modal Deep Learning (2.5 hours)

- Introduction to multi-modal deep learning
- Combining CNNs, RNNs, and transformers for multi-modal tasks
- Implementing multi-modal models with PyTorch
- Practical exercise: visual question answering

8. Day 8: Advanced Topics in Deep Learning

Introduction to Autoencoders (2 hours)

- Overview of autoencoders
- Variational autoencoders (VAEs)
- Implementing autoencoders and VAEs with PyTorch
- Practical exercise: image denoising with autoencoders

Introduction to Graph Neural Networks (2 hours)

- Overview of graph neural networks (GNNs)
- Graph convolutional networks (GCNs)
- Implementing GNNs with PyTorch
- Practical exercise: node classification on a citation network

9. Day 9: Model Interpretability and Optimization

Model Interpretability (2 hours)

- Importance of interpretability in deep learning
- Saliency maps and occlusion analysis
- Using PyTorch Captum library for interpretability
- Practical exercise: analyzing CNN predictions on image classification

Model Optimization and Pruning (2 hours)

- Introduction to model optimization
- Quantization and pruning techniques
- Implementing optimization and pruning with PyTorch
- Practical exercise: optimizing a neural network for deployment

10. Day 10: Advanced Deployment and Industry Applications

Deploying PyTorch Models on Mobile Devices (1.5 hours)

- Introduction to PyTorch Mobile
- Model optimization for mobile deployment
- Implementing PyTorch models in Android and iOS applications
- Practical exercise: deploying a model on a mobile device

Industry Applications and Case Studies (2.5 hours)

- Overview of deep learning applications in various industries
- Case studies: healthcare, finance, autonomous vehicles, and more
- Challenges and future trends in deep learning
- Group discussion: brainstorming ideas for industry-specific applications

Course Recap, Certificates, and Farewell (1 hour)

- Review of key concepts and techniques from the extended course
- Presentation of certificates to participants
- Farewell and networking opportunities

© 2026 Faysal Aziz · Kuala Lumpur