+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Updated on 05/05/2026

Sign up

Deep Learning Training with PyTorch

3 days (21 hours)

Overview

PyTorch has become the go-to framework for research and industry in Artificial Intelligence. Its flexibility, dynamic graph-based approach, and native integration with Python make it the tool of choice for designing innovative Deep Learning architectures.

Our Deep Learning with PyTorch training will help you master the fundamental concepts of neural networks, from tensor manipulation to the creation of complex architectures such as CNNs or Transformers.

You will learn how to structure your training phases, optimize model convergence, and use advanced tools for debugging and visualization. The course emphasizes hands-on practice to empower you to independently solve real-world problems in computer vision or natural language processing.

By the end of the course, you will be able to design, train, and deploy high-performance deep learning models while adopting best development practices related to the PyTorch ecosystem.

Like all our courses, this one will introduce you to **the latest stable version** of the technology and its new features.

Objectives

- Build neural networks
- Train and evaluate models
- Optimize and debug architectures
- Deploying DL models

Target audience

- Data scientists
- AI engineers

Prerequisites

- Knowledge of Python, ML, and mathematics

Software requirements

- At least 8 GB of RAM, 16 GB recommended
- Access to a GPU (via Google Colab, Kaggle, or a local machine with CUDA)
- Python environment with Jupyter Notebook or VS Code

Curriculum for our Deep Learning with PyTorch course

[Day 1 - Morning]

PyTorch Fundamentals and Tensors

- Understanding the PyTorch ecosystem and its advantages
- Working with Tensors: creation, operations, and broadcasting
- The Autodifferentiation Engine: Autograd
- Device Management (CPU vs. GPU/CUDA)
- Introduction to the torch.nn module
- Hands-on workshop: Implementing a linear regression "from scratch" with Autograd.

[Day 1 - Afternoon]

Building neural networks (MLP)

- Structure of a PyTorch Module: method ____init__and forward
- Activation functions (ReLU, Sigmoid, Softmax)
- Initialization of weights and biases
- Dense layers (Linear layers)
- Architecture of multilayer neural networks
- Hands-on workshop: Creating a Multilayer Perceptron for classifying tabular data.

Training and optimization loops

- Choosing loss functions
- Optimization algorithms (SGD, Adam, RMSprop)

- Training cycle: Forward pass, Loss calculation, Backward pass, Optimizer step
- Tracking performance metrics
- Managing DataLoaders and Datasets
- Hands-on workshop: Training a model from scratch on the MNIST dataset.

[Day 2 - Morning]

Deep Learning for Computer Vision (CNN)

- Principles of Convolutional Neural Networks (CNN)
- Convolution, Pooling, and Dropout Layers
- Famous architectures (ResNet, VGG) and Transfer Learning
- Data augmentation with torchvision
- Fine-tuning pre-trained models
- Hands-on workshop: Image classification using a custom, pre-trained CNN model.

[Day 2 - Afternoon] Optimization

and Regularization

- Preventing Overfitting
- Batch normalization techniques
- Learning Rate Scheduling Strategies
- Using TensorBoard or Weights & Biases for visualization
- Saving and loading models (state_dict)
- Hands-on Workshop: Optimizing a Complex Model and Visualizing Convergence

Debugging and Profiling

- Identifying gradient issues (vanishing/exploding gradients)
- Checking Tensor dimensions
- Using PyTorch hooks to inspect layers
- Profiling memory and computational performance
- Hands-on workshop: Debugging session on a deliberately faulty architecture.

[Day 3 - Morning]

Introduction to Sequential Models and Transformers

- Recurrent Networks (RNN, LSTM, GRU)
- Attention Mechanisms

- Introduction to the Transformer architecture
- Using the Hugging Face library with PyTorch
- Applications in NLP (Natural Language Processing)
- Hands-on workshop: Sentiment analysis or simple text generation.

[Day 3 - Afternoon] Deployment

and Industrialization

- Model optimization for inference
- Conversion to TorchScript and ONNX
- Introduction to TorchServe for model serving
- Model quantification and pruning
- MLOps Best Practices with PyTorch
- Hands-on workshop: Exporting a model and simulating an inference API.

Final synthesis project

- Defining a business problem
- Architecture selection and design
- Optimized training and rigorous evaluation
- Presentation of results and areas for improvement
- Hands-on workshop: Final group challenge using a real-world dataset.

Target Audience

This training is intended for both individuals and companies, large or small, wishing to train their teams in a new advanced IT technology or to acquire specific business knowledge or modern methods.

Entry-level assessment

The pre-training assessment complies with Qualiopi quality standards. Upon final registration, the learner receives a self-assessment questionnaire that allows us to evaluate their estimated proficiency in various types of technologies, as well as their expectations and personal goals for the upcoming training, within the limits imposed by the selected format. This questionnaire also allows us to anticipate certain connection or internal security issues within the company (intra-company or virtual classroom) that could pose challenges for monitoring and ensuring the smooth running of the training session.

Teaching Methods

Practical Course: 60% Practical, 40% Theory. Training materials distributed in digital format to all participants.

Organization

The course alternates between theoretical input from the trainer, supported by examples and

reflection sessions and group work.

Assessment

At the end of the session, a multiple-choice questionnaire verifies that the skills have been properly acquired.

Certification

A certificate will be issued to each trainee who has completed the entire training program.