Updated 05/02/2025

Sign up

TensorFlow training

4 days (28 hours)

Presentation

Our Tensorflow training course will enable you use the TensorFlow framework, one of the most widely used tools for machine learning and data analysis.

This course will introduce you to the basics of TensorFlow and its applications in solving complex problems. Throughout the course, you'll have the opportunity to put your knowledge into practice through practical exercises and concrete examples.

You'll learn how to deploy TensorFlow models on different platforms, such as mobile and web applications, and how to integrate them into existing systems.

We then look at cost functions, optimizers and various regularization methods to improve model performance.

As with all our training courses, this one will introduce you to the latest stable version of TensorFlow 2.18, released in July 2024, coupled with Python 3.13.

Objectives

- Discover what's new in v2 and be able to install and use TensorFlow independently
- Understand the concept of Deep Learning and be able to use it with TensorFlow
- Handle large volumes of data using best practices with TensorFlow
- Configuring a machine to run deep learning models
- Create a data pipeline with data augmentation
- Using TensorFlow's pre-built estimators on a dataset
- Understanding the theory behind the different neural network blocks
- Building a neural network with TensorFlow
- Understanding neural network training theory

- Start network training
- Monitor and fine-tune training with Tensorboard and callbacks
- Use pre-trained networks to apply them to a specific case
- Apply neural networks to different applications

Target audience

- Data Analyst / Data Scientist / Data Engineer
- Developers, Big Data Architects, Lead Developers

Prerequisites

- Knowledge of Python ideally, or recent programming language
- Knowledge of mathematics (e.g. gradient, mean/standard deviation), ideally at Baccalaureate+2 engineering school level
- Test My Knowledge

SOFTWARE REQUIREMENTS

- TensorFlow can be installed with GPU compatibility on both Windows and Linux.
- Or for those of you who don't have a powerful enough PC, or don't want to go that extra mile. In case of complexity, you can run the whole course on Google Colab (which provides GPUs free of charge, provided you have an account).

RECOMMENDED READING BEFORE AND AFTER THE COURSE

I recommend the following resources to help you deepen your knowledge of TensorFlow :

- Google's TensorFlow blog, with detailed information on the latest versions of TensorFlow and tips on how to use it.
- Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélien Géron
- Google's TensorFlow YouTube channel, with video tutorials on the principles machine learning and TensorFlow
- TensorFlow developer forums to chat with other users and ask questions. questions about the Framework
- Jason Brownlee's blog, with tips and code examples for machine learning and TensorFlow

TensorFlow version 2 training program

Day 1 morning: Manipulating vectors and tensors with TensorFlow

- Installation
  - Installing Tensorflow
  - Google Collab
- Using Tensorflow as a Numpy
  - Tensors and operations
  - Variables
  - Use cases
  - Huber Loss
  - Image import

Afternoon: Building a data pipeline

- Tensorflow Data API
  - Dataset concept
  - Linking transformations
  - Data shuffle
  - Interleaving
  - Creation of a complete CSV reading pipeline
  - Optimizations
- Tensorflow pre-treatment PLC
  - Usefulness
  - Numeric columns
  - Category columns
  - Use as a layer

Day 2 a.m.: Introduction to neural networks with Keras

- Keras
  - From biology to artificial neurons
  - The biological neuron
  - Logical calculations with neurons
  - The perceptron
  - MLPs and backpropagation
  - Regression MLPs
  - Classification MLPs
- Implementing MLPs with Tensorflow
  - Saving and restoring a model
  - Using callbacks

Afternoon: Training Neural Networks

- The "Vanishing Gradient" problem
  - Reuse of pre-trained models
  - Faster optimizers

- Putting it into practice
  - Creating an image classifier with the Sequential API
  - Creating a regression with the Sequential API
  - Creating complex models with the Functional API
  - Optimizing hyperparameters

Day 3 morning: CNN

- Computer vision
- The architecture of the visual cortex
- Convolution layers
- Pooling layers
- The different architectures of convolutional neural networks (CNN)
- Data Augmentation

Afternoon: Practical work on Models

- Implementing a simple CNN on TensorFlow
- Use of pre-trained models
- Pre-trained models for Transfer Learning
- Object classification and location

Day 4 morning: Application cases

- Sequential data processing with RNNs and CNNs
  - Neurons and recurrent layers
  - Train an RNN
  - Predicting a time series
- Putting Natural Language Processing with RNNs into practice
  - How do you prepare text data?
  - How do you separate the dataset into multiple windows?
  - Generate fake Shakespear
  - Text classification: Sentimental Analysis

Afternoon: Autoencoders

- Representation Learning and Generative Learning with Autoencoders and GANs
  - Creating a PCA with a part of AutoEncoder
  - Building a "Stacked Autoencoder" with TensorFlow
  - Generate fake clothes with GANs

Tensorflow IoT Add-on Module - 2 days

(for team training only)

TensorFlow applied to the IoT

- Why use an embedded system?
  - IoT and AI (Theory)
    - State of the art: concrete examples and possible applications of AI to IoT
    - The two main architectures
      - Cloud AI (Theory)
      - Examples with Google Cloud
    - On device inference (Theory)
    - Examples of applications on Raspberry Pi

TENSORFLOW

- A brief review
- Constraint
  - From Python to C++ (Theory)
    - Python 3.X and any version of C++, using Bazel
  - Hardware (Theory)
    - Raspberry (Ubuntu 16.04 or 18.04) ARM64
  - VPU / TPU
    - Intel Movidius Neural Compute Stick

Introducing TensorFlow Lite Optimization

- Initialize an environment (Theory + Exercises)
  - R1.14 (stable) or R2.0 (stable/preview)
- Workflow
  - Overview (Theory)
  - Obtaining a model (Exercises)
    - Example with InceptionV3
  - Convert (exercises)
    - Using TensorFlow Lite Converter
    - Loading graphics in C++
  - Deploy and apply
    - On an Ubuntu system (16.04 or 18.04)
  - Optimize (Exercises)
    - Quantization
- Performance
  - Best practices (Theory + Exercises)
    - Eliminate redundancy
    - Playing with multi-threading

TensorFlow Advanced Complementary Module - 4 days

(for team training only)

What's new in TensorFlow 2.x

- Easy model building with Keras API
- Deploy robust models in production on any platform
- Cleaning up obsolete APIs and reducing duplication
- Immediate iteration and intuitive debugging
- New extensions
  - Tensors Ragged
  - TensorFlow Probability
  - Tensor2Tensor
- TensorFlow 1 Compatibility Module
  - tensorflow.compat.v1

Understanding Deep Learning

- Deep Learning: The Artificial Intelligence Revolution
- The limits of Machine Learning
- Advantages of Deep Learning over Machine Learning
- Reasons to switch to Deep
- Real-life examples and applicable scenarios
- The Math behind Machine Learning: Linear Algebra
  - Scalars
  - Vectors
  - Matrix
  - Tensors
  - Hyperplanes
- The Math behind Machine Learning: Statistics
  - Probability
  - Conditional Probabilities
  - Posterior Probability
  - Distributions
  - Samples vs Population
  - Resampling Methods
  - Selection Bias
  - Likelihood
- Machine Learning algorithms
  - Regression
  - Classification
  - Clustering
- Reinforcement Learning
- Underfitting & Overfitting
- Optimization
- Convex Optimization

TensorFlow

- Definition: Neural Networks
- Biological Neuron
- Perceptron
- Multi-Layer Feed-Forward Networks
- Learning Neural Networks
- Backpropagation Learning
- Gradient Descent
- Stochastic Gradient Descent
- Quasi-Newton Optimization Methods
- Generative vs Discriminative Models
- Activation Functions
  - Linear
  - Sigmoid
  - Tanh
  - Hard Tanh
  - Softmax
  - Rectified Linear
- Loss Functions
- Loss Function Notation
- Loss Functions for Regression
- Loss Functions for Classification
- Loss Functions for Reconstruction
- Hyperparameters
- Learning Rate
- Regularization
- Momentum
- Sparsity

Understanding neural networks

- Defining Deep Learning
- Defining Deep Networks
- Common Architectural Principals of Deep Networks
- Reinforcement Learning application in Deep Networks
- Parameters
- Layers
- Activation Functions - Sigmoid, Tanh, ReLU
- Loss Functions
- Optimization Algorithms
- Hyperparameters

Convolutional Neural Network (CNN Algorithm)

- Introduction to CNN
- Implementation and architecture of a CNN
- Convolution layers: Pooling layers in a CNN
- Convolution and pooling layers in a CNN
- Understanding and visualizing a CNN
- Transfer learning and development of convolutional neural networks

Recurrent Neural Networks (RNN Algorithm)

- Introduction to the RNN Model
- RNN use cases
- Modelling sequences
- RNNs learning with Backpropagation
- Long Short-Term memory (LSTM)
- Recursive Neural Tensor Network Theory
- Recurrent Neural Network Model

RBM & Autoencoders

- Restricted Boltzmann Machine
- RBM applications
- Collaborative Filtering with RBM
- Introduction to Autoencoders
- Autoencoders applications
- Understanding and using Autoencoders
- Variational Autoencoders
- Deep Belief Network

Companies concerned

This course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

Positioning on entry to training

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the training to come, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

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

Organization

The course alternates theoretical input from the trainer, supported by examples, brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire is used to check correct acquisition.

skills.

Sanction

A certificate will be issued to each trainee who completes the course.