

Deep Learning avec PyTorch

Cours Pratique de 4 jours - 28h

Réf : DLT - Prix 2024 : 2 390CHF HT

Grâce à sa syntaxe simple et intuitive, PyTorch, bibliothèque logicielle Python, est considérée comme plus aisée à apprendre que les autres frameworks deep learning. Sa large communauté permet d'obtenir une documentation utile à tous les développeurs, même débutants en apprentissage profond et en calcul tensoriel.

OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Savoir manipuler des images et du texte avec PyTorch

Mettre en place un entraînement de réseaux de neurones à partir de zéro ou en utilisant le transfer learning

Connaître les modules de PyTorch pour charger les données

Avoir des notions sur les entraînements distribués

Avoir des notions sur les nouvelles méta-architectures telles que les transformers

LE PROGRAMME

dernière mise à jour : 08/2023

1) Prise en main de PyTorch

- PyTorch et ses principes fondamentaux.
- Installer PyTorch et les composantes associées.
- Comparaison entre les bibliothèques Numpy et PyTorch.
- PyTorch vs Tensorflow.
- Principes du calcul distribué.

Travaux pratiques : Installation de PyTorch. Manipulation de tenseurs et de matrices.

2) Sous-modules de PyTorch pour l'entraînement des réseaux de neurones

- Présentation des sous-modules de Pytorch pour l'entraînement des réseaux de neurones.
- Rappels sur la propagation forward.
- Rappels sur la rétropropagation des gradients.
- Chargement des données.
- Définir un réseau neuronal de convolution avec le package torch.nn, entraîner le modèle, le tester.

Travaux pratiques : Mise en place d'un réseau CNN pour la classification d'images.

3) Transfer learning et utilisation de réseaux pré-entraînés

- Principe du transfer learning.
- Exemples de mise en œuvre de l'apprentissage par transfert.
- Les étapes de la méthode de transfer learning dans les projets de machine learning.
- Utilisation de réseaux pré-entraînés.

Travaux pratiques : Reprise d'exercices précédents, pour améliorer les métriques avec la mise en place du transfer learning.

4) Les méta-architectures pour des projets complexes

- Présentation des méta-architectures.

PARTICIPANTS

Concepteurs-développeurs en machine learning, data scientists, ingénieurs en IA.

PRÉREQUIS

Pratique de Python et du machine learning.

COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques...

Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

MOYENS PÉDAGOGIQUES ET TECHNIQUES

- Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
- À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
- Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Vous avez un besoin spécifique d'accessibilité ? Contactez Mme FOSSE, référente handicap, à l'adresse suivante psh-accueil@orsys.fr pour étudier au mieux votre demande et sa faisabilité.

- Problématique de détection d'objets.
 - Problématique de segmentation d'images.
 - L'architecture réseau UNet : les blocs codeur-décodeur et PyTorch.
- Travaux pratiques : Création d'un modèle UNet simple pour la segmentation d'images. Comparaison avec le transfer learning pour UNet.*

5) Le NLP avec PyTorch et spaCy

- Le traitement automatique du langage naturel (NLP, pour Natural Language Processing).
 - L'intérêt de PyTorch et de spaCy.
 - Principe des pipelines.
 - Processing de texte .
 - Entraînement de réseaux récurrents / biLSTM.
 - Utilisation de PyTorch et spaCy pour le NLP.
- Travaux pratiques : Topic modelling sur des avis de films. Analyse des sentiments sur des tweets.*

6) Transformers et mécanismes d'attention

- Les transformers pour le traitement automatique des langues.
 - Détail des mécanismes d'attention.
 - Le mécanisme d'attention appliqué à une séquence : auto-attention.
 - Fonctionnement des transformers.
- Travaux pratiques : Mise en place d'un modèle de traduction.*

LES DATES

CLASSE À DISTANCE
2024 : 27 mai, 10 sept., 19 nov.