New

Deep Learning avec TensorFlow

Développer, optimiser et déployer des modèles de Deep Learning avec TensorFlow et Keras

Présentiel ou en classe à distance

3 jours (21 h)

Prix inter : 2.490,00 € HT
Forfait intra : 7.690,00 € HT

Réf.: IA421

La formation **Deep Learning avec TensorFlow** permet de développer, entraîner, optimiser et déployer des **modèles de deep learning** avec **TensorFlow** et **Keras**. Elle couvre les réseaux de neurones, les **CNN**, les **LSTM**, les **Transformers**, les API avancées de TensorFlow, ainsi que l'industrialisation des modèles avec **TensorFlow Serving**, Docker et le monitoring en production. Une formation complète pour concevoir des solutions d'IA performantes et prêtes pour un déploiement en environnement professionnel.

A qui s'adresse cette formation ?

Pour qui

- Data scientists, AI engineers

Prérequis

- Connaissances en Python, en Machine Learning et en mathématiques
- Disposez-vous des connaissances nécessaires pour suivre cette formation ? Testez-vous !

Programme

1 - Fondamentaux de TensorFlow

- Découvrir l'écosystème TensorFlow, ses composants, ses cas d'usage et la place de Keras
- Créer et manipuler des tenseurs TensorFlow : types, formes et opérations
- Exécuter des calculs sur CPU et GPU et gérer les périphériques
- Utiliser GradientTape pour la différentiation automatique
- Comprendre le principe de la descente de gradient à travers une régression simple

Atelier

Créer et manipuler des tenseurs : indexation, redimensionnement et opérations

Exécuter des opérations sur GPU et comparer les performances avec le CPU

Calculer des gradients avec GradientTape sur des expressions simples

Mettre en oeuvre manuellement une descente de gradient sur une régression linéaire

2 - Architecture et configuration des réseaux de neurones

- Utiliser l'API Keras : modèles séquentiels et API fonctionnelle
- Comprendre les couches, les fonctions d'activation et les conteneurs
- Concevoir l'architecture d'un réseau de neurones entièrement connecté
- Sélectionner les fonctions de perte et les métriques selon le problème traité

- Compiler un modèle avec un optimiseur, une fonction de perte et des métriques
  Atelier

  Définir une architecture dense avec l'API séquentielle et l'API fonctionnelle

  Choisir les fonctions d'activation, la fonction de perte et les métriques

  Compiler le modèle et examiner sa structure et ses paramètres

  Réaliser des prédictions sur un jeu de données de test

3 - Préparer et entraîner un modèle de Deep Learning

- Construire un pipeline de données performant avec tf.data
- Gérer le chargement, les lots, le mélange et le préchargement des données
- Entraîner un modèle avec la méthode fit et suivre les métriques
- Séparer les données d'entraînement et de validation et analyser les courbes d'apprentissage
- Utiliser les callbacks Keras : checkpoints, arrêt anticipé et journalisation
- Sauvegarder et recharger un modèle TensorFlow
  Atelier

  Construire un pipeline de données d'entraînement avec tf.data

  Entraîner un modèle avec fit et configurer les callbacks

  Suivre la perte et les métriques sur les données d'entraînement et de validation

  Sauvegarder le modèle entraîné et le recharger pour vérifier son fonctionnement

4 - Concevoir des modèles CNN pour la classification d'images

- Comprendre le principe de la convolution : filtres, cartes de caractéristiques et pooling
- Concevoir l'architecture d'un réseau de neurones convolutif (CNN)
- Augmenter les données avec les couches de prétraitement Keras
- Utiliser le transfert d'apprentissage avec les modèles préentraînés de Keras
- Applications Appliquer le réglage fin (fine-tuning) à un modèle existant
  Atelier

  Préparer un jeu d'images avec tf.data et l'augmentation de données

  Construire et entraîner un CNN avec Keras

  Appliquer le transfert d'apprentissage à partir d'un modèle préentraîné

  Comparer un modèle entraîné à partir de zéro avec un modèle affiné

5 - Réseaux LSTM et GRU

- Comprendre les enjeux et les modes de représentation des données séquentielles
- Concevoir des réseaux de neurones récurrents avec Keras, LSTM et GRU
- Comprendre les mécanismes de mémoire et la succession des blocs LSTM
  Atelier

  Préparer un jeu de données destiné à la prédiction de séries temporelles

  Transformer les données séquentielles pour l'entraînement

  Construire et entraîner un réseau récurrent LSTM ou GRU

6 - Architecture Transformer et mécanisme d'attention

- Comprendre le mécanisme d'attention et l'architecture Transformer
- Exploiter des modèles Transformers préentraînés avec KerasNLP
- Choisir une architecture selon la nature de la tâche
  Atelier

  Utiliser un modèle Transformer préentraîné pour une tâche séquentielle

  Configurer les hyperparamètres du modèle

  Comparer les performances des approches LSTM et Transformers

7 - Personnalisation, optimisation et évaluation

- Créer des couches et des modèles personnalisés par sous-classement
- Concevoir des boucles d'entraînement sur mesure avec GradientTape
- Accélérer les calculs avec tf.function et le graphe d'exécution
- Évaluer un modèle à l'aide de métriques et d'une matrice de confusion
- Diagnostiquer le surapprentissage et appliquer des techniques de régularisation
- Suivre les expériences avec TensorBoard
  Atelier

  Créer une couche ou un modèle personnalisé par sous-classement

  Écrire une boucle d'entraînement sur mesure avec GradientTape

  Accélérer l'exécution avec tf.function et mesurer le gain obtenu

  Évaluer le modèle et visualiser l'entraînement avec TensorBoard

8 - Préparer un modèle TensorFlow pour la production

- Comprendre la structure du format SavedModel et les bonnes pratiques d'export
- Optimiser un modèle pour l'inférence grâce à la quantification et à l'élagage
- Utiliser TensorFlow Lite dans les environnements contraints et embarqués
- Comprendre le format ONNX et l'interopérabilité entre les frameworks
- Mesurer la latence et le débit d'un modèle
  Atelier

  Exporter un modèle entraîné au format SavedModel

  Appliquer la quantification et mesurer les gains de taille et de latence

  Convertir un modèle avec TensorFlow Lite pour un environnement contraint

  Comparer les performances d'inférence avant et après l'optimisation

9 - Déployer un modèle dans un service d'inférence

- Comparer les architectures de déploiement : inférence par lots et en temps réel
- Servir un modèle avec TensorFlow Serving
- Exposer le modèle à travers une API d'inférence REST ou gRPC
- Conteneuriser le service d'inférence avec Docker
- Gérer les différentes versions des modèles déployés
  Atelier

  Servir un modèle SavedModel avec TensorFlow Serving

  Exposer le modèle à travers une API REST et tester des requêtes d'inférence

  Conteneuriser le service avec Docker

  Mettre en service une nouvelle version du modèle et vérifier la bascule

10 - Surveiller les performances et la qualité des modèles

- Comprendre les enjeux du monitoring d'un modèle de Deep Learning en production
- Suivre les indicateurs techniques : latence, débit, taux d'erreur et utilisation des ressources
- Surveiller la qualité du modèle et détecter la dérive des données et des prédictions
- Journaliser les prédictions et mettre en place des alertes
- Définir des stratégies de mise à jour et de réentraînement des modèles
  Atelier

  Instrumenter le service d'inférence pour collecter la latence et le taux d'erreur

  Journaliser les prédictions et les données d'entrée

  Détecter une dérive des données dans un flux de prédictions

  Définir des seuils d'alerte et une stratégie de réentraînement

Les objectifs de la formation

- Concevoir et entraîner des réseaux de neurones adaptés à différents cas d'usage
- Évaluer et optimiser les performances de modèles de Deep Learning
- Développer des applications d'IA avec les API avancées de TensorFlow
- Déployer des modèles de Deep Learning en environnement de production
- Monitorer les modèles afin d'assurer leur performance et leur fiabilité dans le temps

Evaluation

- Pendant la formation, le formateur évalue la progression pédagogique des participants via des QCM, des mises en situation et des travaux pratiques. Les participants passent un test de positionnement avant et après la formation pour valider leurs compétences acquises.

Les points forts de la formation

- Une formation complète couvrant l'ensemble du cycle de vie d'un modèle de Deep Learning, de sa conception avec Keras à son déploiement et à son monitoring en production
- Des ateliers pratiques s'appuyant sur l'écosystème TensorFlow (Keras, tf.data, TensorFlow Serving, TensorFlow Lite et TensorBoard) pour concevoir, entraîner, optimiser et déployer des modèles sur des cas d'usage concrets

Dates et villes 2026 - Référence IA421

Dernières places disponibles Session garantie

A distance

du 19 oct. au 21 oct.	du 14 déc. au 16 déc.

Paris

du 19 oct. au 21 oct.	du 14 déc. au 16 déc.