

TP : Apprentissage par transfert (Transfer Learning) pour la classification d'images

L'apprentissage par transfert constitue une technique puissante en l'apprentissage profond, surtout dans le cadre des tâches de classification d'images.

L'objectif de ce TP est de construire un modèle de classification d'images en utilisant l'apprentissage par transfert pour distinguer différentes classes à partir de la base de données PlantVillage.

NB : Assurez-vous de documenter votre code de manière précise, d'inclure des graphiques illustrant les performances des modèles, et d'interpréter de manière approfondie les résultats obtenus.

Exercice 1: Implémentation des modèles VGG19, ResNet34 et DenseNet121

Implémenter les architectures VGG19, ResNet34, et DenseNet121 en utilisant les frameworks keras/TensorFlow ou PyTorch. Ensuite, exécuter l'entraînement et présenter les résultats obtenus.

Références:

- VGG19 et ResNet 34 : <https://arxiv.org/pdf/1512.03385.pdf>,
- DensNet121 : https://link.springer.com.eresources.imist.ma/chapter/10.1007/978-3-030-51935-3_7

Exercice 2: Utilisation de poids pré-entraînés

Reprendre les modèles de l'exercice 1, mais cette fois-ci utiliser des poids pré-entraînés sur des ensembles de données étendus tels qu'ImageNet.

Retirer la tête de classification originale de chaque modèle pré-entraîné et la remplacer par une nouvelle tête adaptée au nombre de classes de la base de données Planet Village.

Examiner vos constatations et débattre des avantages associés à l'utilisation de poids pré-entraînés.

Exercice 3: Transfert learning avec EfficientNet et ViT

En utilisant le même jeu des données 'PlanetVillage', appliquer la technique de transfert learning pour les modèles EfficientNet et ViT.

Références:

- EfficientNet : <https://arxiv.org/pdf/1905.11946.pdf>
- ViT : <https://arxiv.org/pdf/2010.11929.pdf>

Quels sont les avantages spécifiques des modèles EfficientNet et Vision Transformer (ViT) par rapport aux architectures précédentes en termes de performances en transfert learning pour la classification d'images?