

Travaux pratiques du Module 3 (Few-Shots Learning), cours MLAI

Installation

Les packages `pytorch` et `torchvision` doivent être installés dans votre environnement de développement Python. Si vous utilisez Anaconda, entrez par exemple les commandes suivantes :

```
conda install pytorch-cpu -c pytorch
pip install torchvision
```

Deadline

Avant le 17 novembre de 2023 vous enverrez à « alejandro.ribes@edf.fr » un Notebook qui contiendra les exercices demandés dans les sections «Few-Shots Learning sur tiny-ImageNet ».

Note 1 : Vous pouvez utiliser Altair (introduit dans le module 1 de ce cours) pour visualiser des courbes.

Note 2 : Les noms des auteurs du Notebook doivent être écrits en haut du document.

Tutoriaux

Vous allez suivre le tutoriel suivant :

1. *Your Own Few-Shot Classification Model Ready in 15mn with PyTorch*
<https://www.sicara.fr/blog-technique/your-few-shot-model-15mn-pytorch>

Ce tutoriel d'Etienne Bennequin présente les concepts de base du Few-Shots Learning. Nous remarquons qu'il utilise sa librairie `easyfsl`. Le code étant en PyTorch, vous pouvez réutiliser ce code et la librairie, ou vous inspirer, pour réaliser le TP. Vous pouvez télécharger directement le Notebook, cela vous évite de copier le code. Vous pouvez aussi l'exécuter sur « Google Colab ».

Si vous avez besoin de réviser les concepts de base de PyTorch, vous pouvez regarder :

2. *Training a classifier*
https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html

Note : Si votre machine dispose d'un GPU compatible CUDA et que vous souhaitez en tirer parti, vous devez indiquer à PyTorch d'utiliser CUDA : les tenseurs et les couches du réseau seront stockés dans la mémoire GPU. Les calculs seront parallélisés. Pour ce faire, utilisez la fonction `cuda()` lors de l'initialisation du réseau :

```
net = Net().cuda()
```

Pendant ces tutoriels, les documentations des bibliothèques à consulter sont là :

- Pytorch : <https://pytorch.org/docs>
- Numpy and Scipy: <https://docs.scipy.org/doc>
- Altair : <https://altair-viz.github.io/>

Few-Shots Learning sur tiny-ImageNet

Le but des exercices est de réaliser une classification d'images « few-shots » provenant de la base ImageNET (<http://www.image-net.org/>). Dans le TP du module 2, vous avez travaillé avec un sous-ensemble de la base tiny-ImageNET, qui est elle-même une petite portion d'ImageNET. Ce base comporte 200 classes et vous avez déjà entraîné un réseaux de neurones en utilisant 40 classes. On appellera ce réseau *tinyNet40*.

Rappel :

Chaque classe contient 500 images RGB 64×64 . Les fichiers source fournis contiennent 2 classes dérivées de `torch.utils.data.Dataset`, qui permettent de parcourir les ensembles d'apprentissage et de test. Pour chaque classe, les 450 premiers exemples sont utilisés pour l'apprentissage, les 50 restants pour les tests. Comme dans le tutoriel « *Training a classifier* », vous avez construit un réseau composé de plusieurs “séquences convolutives” suivies de plusieurs couches denses. On rappelle qu'une séquence convolutive est composée d'une couche de convolution, d'une fonction d'activation non-linéaire et d'un *max-pooling*.

Exercice 1

Vous allez sélectionner 3 classes parmi les 200 (pas utilisés pour entraîner votre réseau). Vous pouvez choisir ces 3 classes comme vous le souhaitez (les premières, les dernières, aléatoirement, une chaque 10, etc). Pour chaque classe sélectionnez 5 images. Vous allez donc avoir un « support set » du type « 3-way 5-shots ». Visualisez cet ensemble d'images et préparez quelques images de « query ».

Exercice 2

Adaptez la classe *PrototypicalNetworks* du tutoriel pour fonctionner avec votre réseau *tinyNet40*. Maintenant, vous allez utiliser *tinyNet40* comme une fonction de distance.

Testez les images de « query ». Préparez un ensemble de « query test » avec 21 images (7 par classe).

Exercice 3 (optionnel)

Répéter l'exercice 2 avec deux problèmes « 5-way 5-shots » et « 8-way 5-shots ». Faites un tableau ou un graphe et comparez leurs précision.

Exercice 4 (optionnel)

Répéter l'exercice 2 avec trois problèmes « 5-way 5-shots », « 5-way 10-shots » et « 5-way 20-shots ». Faites un tableau ou un graphe et comparez leurs précision.