

Notes de cours Initiation Jupyter - DL - Python

Thibaut Marmey

October 24, 2018

Contents

| | | |
|----------|--|----------|
| 1 | Initiation à Jupyter Notebook | 1 |
| 1.1 | Importer des bibliothèques au kernel | 1 |
| 1.2 | Généralité | 2 |
| 1.2.1 | RGB Image Representation | 2 |
| 2 | Initiation au Deep-learning | 2 |
| 2.1 | Général | 2 |
| 2.2 | CADL session | 3 |
| 2.2.1 | Session-0 | 3 |

1 Initiation à Jupyter Notebook

1.1 Importer des bibliothèques au kernel

- *import libraryName*
- S'il y a un problème de librairie : *pip install libraryName*
 - *matplotlib*
 - *image*
- *os* : lire, écrire, ouvrir, manipuler des fichiers
 - *os.mkdir()*
 - *os.listdir()*
 - *[a for a in os.listdir('nameFolder') if '.jpg' in a or '.jpeg' in a]* (spécifier des fichiers)
 - *var.endswith('.extension/smith')* (renvoie True/False)
 - *os.path.join(path, *paths)* : Join one or more path components intelligently. The return value is the concatenation of *path* and any members of **paths*.
- *urllib.request* : interagir avec les URLs
- *ssl* (Secure Sockets Layer) : créer une connection sécurisé (crypté) entre un client et un serveur.

- *matplotlib.pyplot* as *plt* : visualisation and chargement d'images
On peut ainsi faire référence à *matplotlib.pyplot* seulement avec *plt*. C'est une pratique courante.
 - *%matplotlib inline* : la visualisation se fera dans notebook et non pas avec une fenêtre pop-up (méthode par défaut)
 - *plt.figure()*; *plt.imshow(img)* : permet d'afficher plusieurs images
- *numpy* as *np* : permet de travailler sur des variables numériques plus facilement (images par exemple)
 - *np.random.randint(0, len(listOfFiles))* : retourne aléatoirement des entiers entre 0 et la longueur de la listeOfFiles.
 - *np.mean(img, axis=n)* : renvoie une image dont la valeur des pixels est la moyenne De $R+G+B$ des pixels
 - *data = np.array(imgs)* : pour avoir la dimension batch ou nombre d'images. La *shape* de *data* est alors : $N * H * W * C$
 - * N : nombre d'images dans *data*
 - * H : hauteur ou nombre de lignes
 - * W : largeur ou nombre de colonnes
 - * C : nombre de couleurs dans les images (RGB=3, gray=1)
 - *data = np.concatenate([img[np.newaxis] for img in imgs], axis=0)*
- *scipy* : library used for scientific computing and technical computing. contains modules for optimization, linear algebra, integration, interpolation, special functions, FFT, signal and image processing, ODE solvers and other tasks common in science and engineering.
 - *from scipy.misc import imread*
- *scikit-image* : functions to convert an image (like from RGB to grayscale)

1.2 Généralité

1.2.1 RGB Image Representation

- *plt.imshow(img)*
- *img.shape* (lignes, colonnes, couleurs): dimension de la donnée (image)
- *img.dtype* : donne le nombre de bits utilisé pour coder l'image
 - *uint8* : *unsigned, int*, 8 : pas de signe (toutes les valeurs seront positives), seulement des entiers, codés sur 8 bits (de 0 à 255)

2 Initiation au Deep-learning

2.1 Général

- Cours GitHub Kadenze
- Installation Anaconda
- Lancer anaconda : *anaconda-navigator*
- Créer un environnement virtuel via anaconda : *conda create -n nomFile anaconda*
- Installer TF dans cet env. : *conda install -c conda-forge tensorflow*
- *conda install jupyter notebook*
- Lancer env. virtuel : *source activate tensorflow* (*tensorflow* est le nom donné à l'env. virtuel)
- Lancer Jupyter : *jupyter notebook*
- Tester Tensorflow : *python -c "import tensorflow as tf; print(tf.__version__)"*
- Quitter l'env. virtuel : *source deactivate*

2.2 CADL session

2.2.1 Session-0

- Dataset *Celeb Net*, à peu près 200 000 images de célébrités.
- Généralement on travaille avec des images de 32 bits dans les réseaux de neurones. Changer l'image au format voulu : *img.astype(np.float32)*
- Définir une fonction : *def functionName(args):*
- En DL, les images du dataset doivent avoir la même taille. Il faut donc rogner ces images au même format sans altérer les éléments importants dans celles-ci. Une méthode courante est de rogner l'image en carré en prenant le côté le plus long et le mettre à la même taille que le côté le plus court. Par cette méthode il est possible de redimensionner l'image sans la déformer.
- La convention pour stocker plusieurs images est d'utiliser le batch dimension (de la forme N*H*W*C via *data.shape*) avec *np.array(imgs)* ou *np.concatenate()*.