

Mise en Production

Modèle Préconisé

Au vu des performances du modèle par CNN, il est très fortement recommandé de déployer ce modèle. Pour ce faire, vous trouverez ci-dessous les prérequis en terme de hardware et software avant de voir comment l'utiliser.

Hardware

Ce modèle demandant beaucoup de calculs, il est fortement recommandé d'avoir une carte graphique de marque NVidia. En fonction de la mémoire graphique de celle-ci, vous pourrez traiter plus ou moins d'images à chaque traitement. Si vous n'avez pas de carte graphique, le temps de calcul sera beaucoup plus long et vous pourrez traiter que quelques images à chaque fois.

Software

Commun à une architecture CPU/GPU

Dans un premier temps vous aurez besoin d'une version de Python 3 (dans mon cas, j'ai la version 3.6.3 avec Anaconda). En plus de ceci, vous aurez besoin de:

- Numpy (1.13.3+)
- Keras (2.0.9+)
- Scikit-Learn (0.19.1+)
- Tensorflow ou Theano (dans mon cas, j'ai Tensorflow 1.4.0)

Dans le cas où vous avez un GPU, il faut prendre Tensorflow-gpu

Uniquement pour les configurations à GPU Nvidia

Afin de faire fonctionner Tensorflow sur la carte graphique, vous aurez besoin de Cuda (8+) disponible sur <https://developer.nvidia.com/cuda-downloads> ainsi que CudNN disponible sur <https://developer.nvidia.com/cudnn>. Vous devrez remplacer dans l'installation de Cuda les fichiers présents dans l'archive de CudNN.

Une fois cela fait, vous devrez ajouter au %PATH%, les chemins pour CUDA. Cela permettra à Tensorflow de faire les calculs sur le GPU.

Utilisation

Pour utiliser ce modèle, vous devez laisser les fichiers dans le même dossier. Cela inclut:

- Classifieur.py (script qui gère le modèle)
- Final_classifieur.h5 (le modèle pour Keras du Classifieur)
- Label_binarizer.pkl (gère la conversion index -> Race du chien)

Ensuite vous devrez appeler la commande `python classifieur.py img1 ... imgN` avec imgX le chemin vers l'image à classer. Le modèle fournira la prédiction pour chaque image et passera si l'image n'existe pas.