**TUTO ML5**

<https://www.youtube.com/playlist?list=PLRqwX-V7Uu6YPSwT06y_AEYTqIwbeam3y>

1\_ présentation série

2\_ introduction

Différence entre algorithme et MachineLearning : a gauche ou a droite \_ vidéo 3.30-5 :30 \_ 2 schémas

* Algo : on lui donne la règle
* ML se crée une règle a partir des dataset labelisées
* Environnement logiciel : p5 > tensorflow (c++)>tensorflow.js(js) > ml5

Ml5 pas besoin de s’occuper de mémory management

* 3 composants 1/neural networks 2/ model pretrained 3/dataset  
  GPT PreTrained

**3 \_ image classification**

Note : regarder date videos, si avant 2022 sur l’ancienne version du site ml5 et regarder si les exemples ont été maj

Schéma en 4’

MobileNet est un PreTrainedModel créé par un Supervide Learning process avec un dataset

Le Dataset utilisé s’appelle à partir imagenet 15 millions images, >21000 catégories, initié par Fei-Fei Li

A voir/lire

* How we teach computers to understand images : <https://www.youtube.com/watch?v=40riCqvRoMs&t=0s>
* voir aussi wordnet : <https://wordnet.princeton.edu/>
* Excavating AI / The Politics of Images in Machine Learning Training Sets <https://excavating.ai/>

Mobile Net n’utilise qu’une part de ImageNet (subset)

le ILSVRC Imagenet Large Scale Visual Recogniition Challende

https://fr.wikipedia.org/wiki/ImageNet,

Image repo github : <https://github.com/EliSchwartz/imagenet-sample-images>

Pour entrainer à reconnaitre nos propres images, labels : <https://teachablemachine.withgoogle.com/>

Limites du dataset utilisé dans ml5 : <https://github.com/ml5js/ml5-next-gen>

MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications <https://arxiv.org/abs/1704.04861>

pre-trained tensorflow.js models github <https://github.com/tensorflow/tfjs-models>

MobileNet github : <https://github.com/tensorflow/tfjs-models/tree/master/mobilenet>

----------------

Début du code :

Référence ml5 : <https://docs.ml5js.org/#/reference/overview> => voir la liste des modèles supportés

Exp : imageclassifier : <https://docs.ml5js.org/#/reference/image-classifier>

Remarque : tout est calculé dans le browser du pc, donc pas sur un serveur. Bien pour la privacy

ON DEVICE MODEL

SCRIPT

1\_imagenet

2\_doodlenet

3\_transfert learning (training own model)

* https://teachablemachine.withgoogle.com/