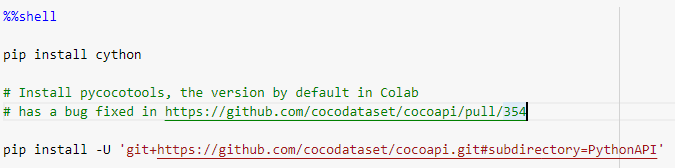
**GUIDELINES FOR IMPLEMENTING SSD**

Clone the repository:

!git clone <https://github.com/ganeshkulkarni98/ssd.git>

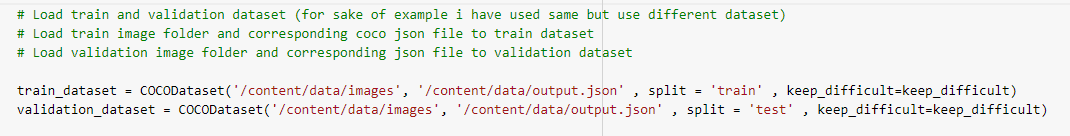
Install Requirements:

!pip3 install -r /content/ssd/requirements.txt



**Upload dataset of COCO dataset format**

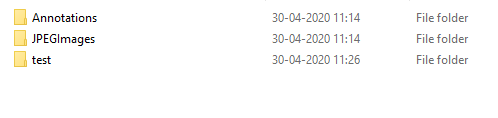
Image and JSON file are two dataset inputs.



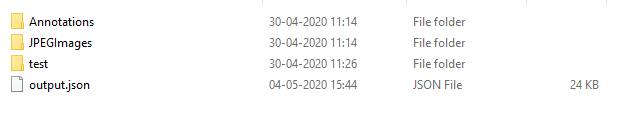
JSON file should be like, {'images': [], 'categories': [], 'annotations': []}

**If Dataset type is pascal VOC,**

**VOC to COCO conversion**



* Above is input in data folder of voc format
* Use following cammand to create json file of coco format
* python voc2coco.py ./data/Annotations ./data/output.json
* coco\_output.json will be created in same data folder.



### Download pretrained weight

VGG is used as backbone in SSD. Last two layers of vgg, fully connected layers are updated.

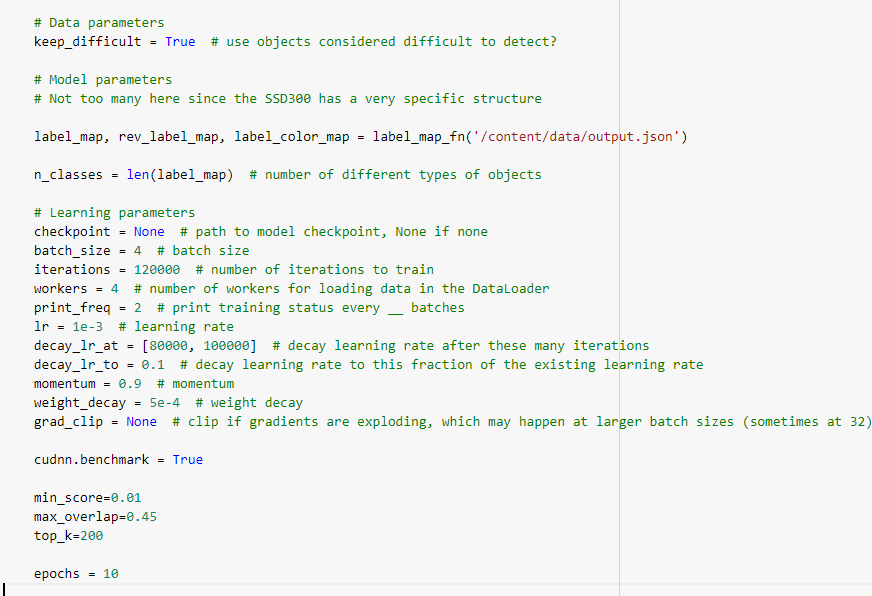
Weight are trained on Imagenet

<https://download.pytorch.org/models/vgg16-397923af.pth>

### Training and Testing of SSD

In order to train SSD, input custom dataset should be in COCO dataset format. Sample of custom dataset for understading is given on this github. JSON file of dataset should be in COCO dataset format format of JSON file should be like, {'images': [], 'categories': [], 'annotations': []}

Load dataset in main function. Input are image folder and json file



to save models stats at each epoch



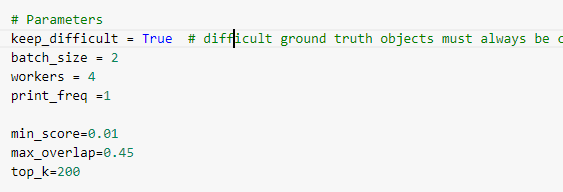
To run train file

!python train.py

### Only Testing of SSD

Provide input image and json file

Assign hyperparameters



This line of code it to test model only.

Input : image folder and json file with hyperparameters

Output : mean average precision and losses

!python test.py

### Prediction using SSD

Input : image folder , jaon file for lable map

Output : list of numpy array annotated images and list of output values of box, label and score in dictionary

!python detect.py

Github link : <https://colab.research.google.com/drive/1sF1o7DTjjARvInYq1WKxJGpsQ9vdpHFF?usp=sharing>