Insights

20.12.2019

Transfer Learning:

Say, you have trained an image classifier model which predicts if an image has a cat or a dog using neural networks. Say, you have used optimization and different variations of models. There will be many layers and weights will change accordingly, deriving a higher accuracy and lower loss function. Now you can use that pre-trained model (accuracy and layers) to predict whether an image has a drone in it or not.

What’s good about it? Say, you have trained an object detection model using 1 million images of an object. That model will learn complex shapes, architecture of objects and various insights on how to detect an object. You can use transfer learning to detect cancerous cell (whether a cell has cancerous cell or not?) in human body. Even if you have only 100 images of human cells. If you tried to train using only 100 images of human cell, then it would have predicted poorly.

When transfer learning makes sense?

Summary:

* Learn from some task A and transfer that knowledge on some task B.
* Task A and B have the same input x
* If you have lot more data in Task A and very few data in Task B, under the assumption that you really want to do well in Task B.
* Low level features of task A will help in learning Task B.

<https://www.youtube.com/watch?v=yofjFQddwHE>

Andrew Ng

Checkout: transfer learning blog

<https://medium.com/@14prakash/transfer-learning-using-keras-d804b2e04ef8>

TensorFlow:

Resources:

1.Go to Tensorflow website>click get started> again get started> Machine Learning Crash course(made by Google :D ).

<https://developers.google.com/machine-learning/crash-course/ml-intro>

2. Tensorflow Git hub

<https://github.com/jtoy/awesome-tensorflow>

3. Paid staff Udemy – Complete Guide to Tensorflow for Deep Learning with Python.

# 4. 30 Amazing Machine Learning Projects for the Past Year (v.2018)

# <https://medium.mybridge.co/30-amazing-machine-learning-projects-for-the-past-year-v-2018-b853b8621ac7>

# Best way to learn something, is to read the Documentation.

22.12.2019

Colab has a sort of documentation for all kinds of charts and it’s available for everyone.

Matplotlib, Seaborn, [Altair](http://altair-viz.github.io) (is a declarative visualization library for creating interactive visualizations in Python, and is installed and enabled in Colab by default.)

# Charting in Colaboratory

A common use for notebooks is data visualization using charts. Colaboratory makes this easy with several charting tools available as Python imports.

<https://colab.research.google.com/notebooks/charts.ipynb>

This guide trains a neural network model to classify images of clothing, like sneakers and shirts. It's okay if you don't understand all the details; this is a fast-paced overview of a complete TensorFlow program with the details explained as you go.

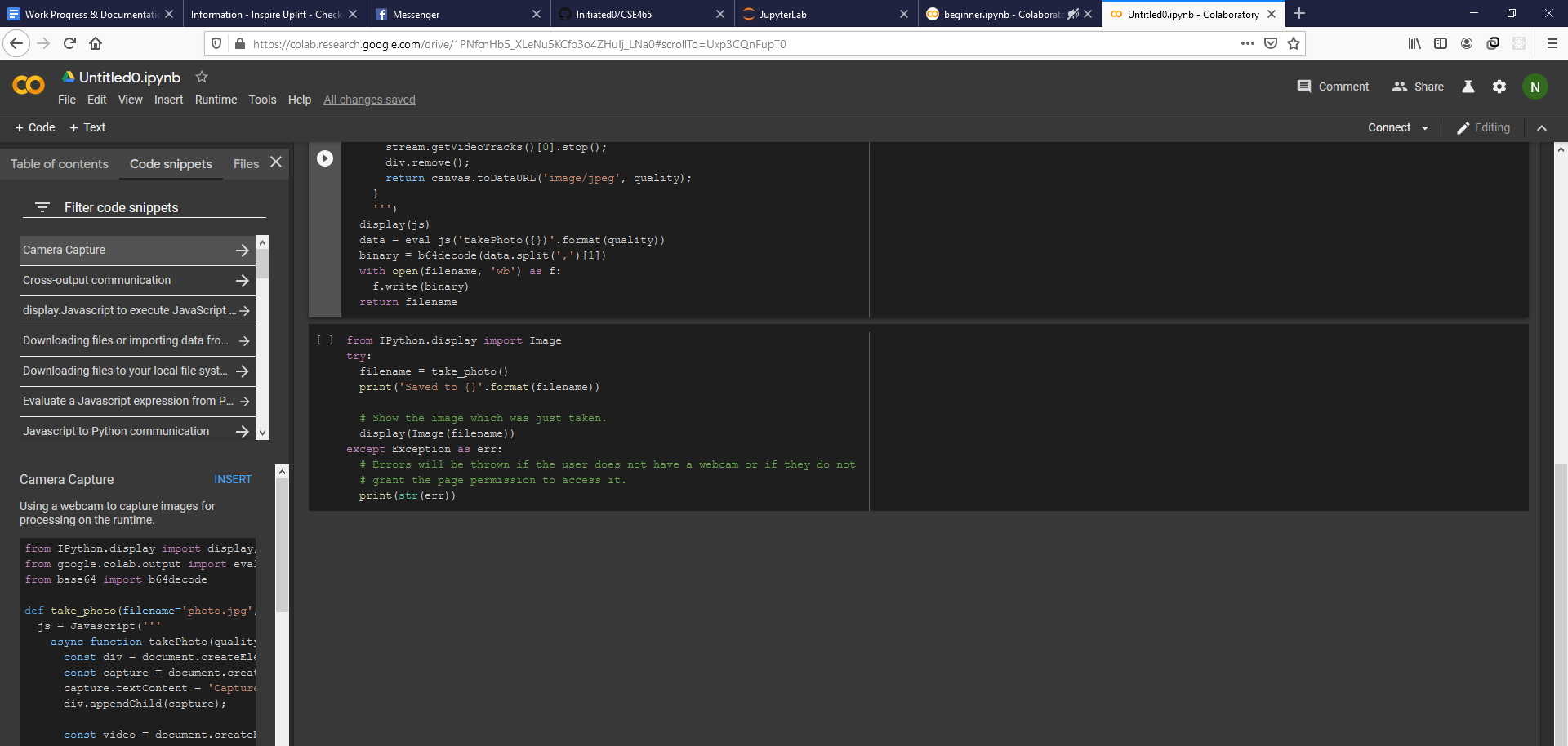
This guide uses [tf.keras](https://www.tensorflow.org/guide/keras), a high-level API to build and train models in TensorFlow.

<https://colab.research.google.com/github/tensorflow/docs/blob/master/site/en/tutorials/keras/classification.ipynb#scrollTo=FbVhjPpzn6BM>

Found a code snippet called Camera Capture” which we can use in our 499 to take input through webcam and process it on runtime and it’s in Colab. :D

Camera Capture

Using a webcam to capture images for processing on the runtime.



https://colab.research.google.com/drive/1PNfcnHb5\_XLeNu5KCfp3o4ZHuIj\_LNa0#scrollTo=Uxp3CQnFupT0

Colab BOSS. A gem like YouTube.

It’s like lego building blocks, there are all kinds of different blocks (codes) out there catering to your every need. You just need the knowledge and dedication to put it together and build a city. :D

-\_-

Using TensorFlow backend.

The default version of TensorFlow in Colab will soon switch to TensorFlow 2.x.  
We recommend you [upgrade](https://www.tensorflow.org/guide/migrate) now or ensure your notebook will continue to use TensorFlow 1.x via the %tensorflow\_version 1.x magic: [more info](https://colab.research.google.com/notebooks/tensorflow_version.ipynb).

/usr/local/lib/python3.6/dist-packages/sklearn/externals/six.py:31: DeprecationWarning: The module is deprecated in version 0.21 and will be removed in version 0.23 since we've dropped support for Python 2.7. Please rely on the official version of six (<https://pypi.org/project/six/>).

"(<https://pypi.org/project/six/>).", DeprecationWarning)

DeprecationWarning -\_-

Just wrote :

%tensorflow\_version 2.x

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

FileNotFoundError Traceback (most recent call last)

[<ipython-input-6-edf221b0e533>](https://localhost:8080/) in <module>()

**81** y = np.asarray(y)

**82** return X,y

---> 83 X\_train, y\_train = get\_data(train\_dir)

**84** #X\_test, y\_test= get\_data(test\_dir) # Too few images

**85**

[<ipython-input-6-edf221b0e533>](https://localhost:8080/) in get\_data(folder)

**9** X = []

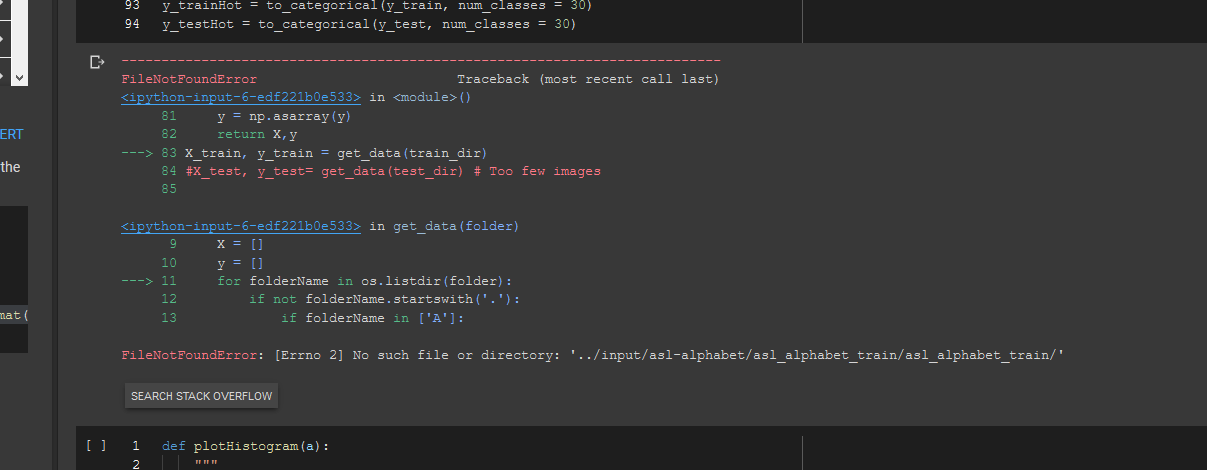
**10** y = []

---> 11 for folderName in os.listdir(folder):

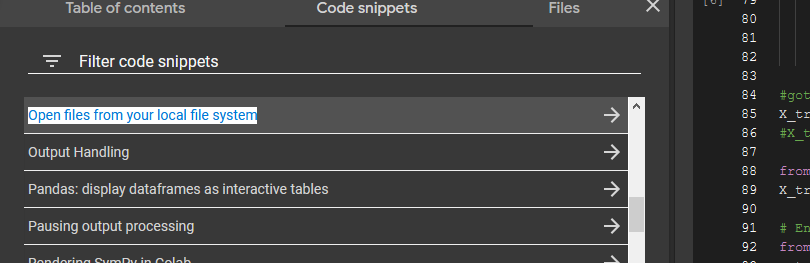
**12** if not folderName.startswith('.'):

**13** if folderName in ['A']:

FileNotFoundError: [Errno 2] No such file or directory: '../input/asl-alphabet/asl\_alphabet\_train/asl\_alphabet\_train/'



Solution in Colab > Code Snippets > Filter code snippets(not quite)



Open files from your local file system

files.upload returns a dictionary of the files which were uploaded. The dictionary is keyed by the file name, the value is the data which was uploaded.

What is a python dictionary? Specifics are very important.

1.Enthought(educational YouTube Channel)

<https://www.youtube.com/watch?v=6AooLxn0f3A>

2. [Corey Schafer](https://www.youtube.com/channel/UCCezIgC97PvUuR4_gbFUs5g) (\*\*\*\*)

407K subscribers

Check Out : <https://realpython.com/lessons/dictionary-python/> (ml website)

Machine Learning Expert’s GitHUB

# Jonathan Hui

<https://github.com/jhui>

khoni ekta

<https://github.com/jhui/deep_learning/blob/master/stanford_cs231n/assignment1/features.ipynb>

465

Start from here

<https://towardsdatascience.com/using-cnns-and-rnns-for-music-genre-recognition-2435fb2ed6af>

or google better