



INVIGOUR ENERGY
OIL & GAS CONSULTANCY

Superman vs Bird (or whatever) for Dummies

Lesson 1 Exercise (way overdue)

By Zaim – A noob

Why Superman and bird?

*It's a Bird... It's a Plane... It's
Superman*



Original Broadway Cast Recording

| | |
|-------------|--|
| Music | Charles Strouse |
| Lyrics | Lee Adams |
| Book | David Newman Robert Benton |
| Basis | <i>Superman</i> by Jerry Siegel and Joe Shuster |
| Productions | 1966 Broadway 1975 ABC TV special 2007 Los Angeles Concert 2010 Dallas 2013 New York City Encores! 2014 London 2015 West End 2016 Germany |

- Who are the Dummies?
- General Steps
- Downloading Images : The Challenge
- How to look for help
- Downloading Images
- Cleanup
- Renaming and renumbering
- Transfer to cloud
- Train your model
- Fine tune your model

- Who are the Dummies?
- General Steps
- Downloading Images : The Challenge
- How to look for help
- Downloading Images
- Cleanup
- Renaming and renumbering
- Transfer to cloud
- Train your model
- Fine tune your model

- Non IT background
- Little coding experience
- May not even know how to transfer files in terminals
- Didn't know the difference between Machine Learning and Deep Learning

- Non IT background
- Little coding experience
- May not even know how to transfer files in terminals
- Didn't know the difference between Machine Learning and Deep Learning

- Lots of logistics nightmare



- Who are the Dummies?
- **General Steps**
- Downloading Images : The Challenge
- How to look for help
- Downloading Images
- Cleanup
- Renaming and renumbering
- Transfer to cloud
- Train your model
- Fine tune your model

cnnsuperbird

cnn fast ai using superman bird data set I downloaded from google

General steps

1. Download superman and bird images from google using someone's python scripts
2. Delete bad images
3. Start formatting the filenames using a simple python script I wrote
4. Separate the files into training and validation folders
5. upload to this github
6. clone back to my google cloud
7. run the jupyter notebook
8. experiment with batch size since the images are not a lot
9. also experiment with data augmentation, and the settings for the learn including annealing and differential learning rate

<https://github.com/zaimawang/cnnsuperbird/blob/master/README.md>

In the process I learned to create my own github



- Who are the Dummies?
- General Steps
- **Downloading Images : The Challenge**
- How to look for help
- Downloading Images
- Cleanup
- Renaming and renumbering
- Transfer to cloud
- Train your model
- Fine tune your model

Challenge : Download Images

- Manually?



Challenge : Download Images

- Issues with small sample size

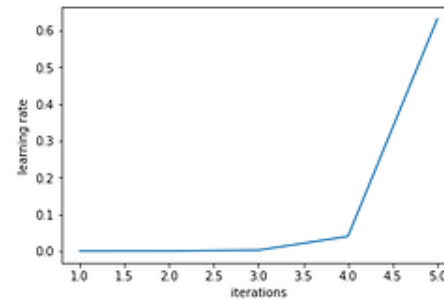


Wiki: Lesson 1

Part 1 v2

3. I used a batch size of 15. Still my learning rate schedule doesn't work

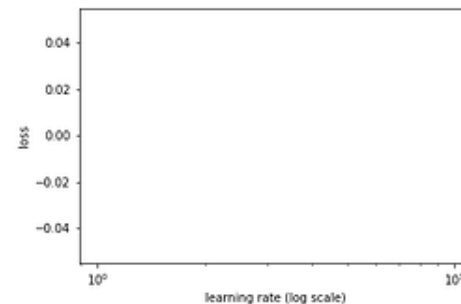
```
In [85]: learn.sched.plot_lr()
```



Note that in the previous plot *iteration* is one iteration (or *minibatch*) of SGD. In one epoch there are 10 iterations.

We can see the plot of loss versus learning rate to see where our loss stops decreasing:

```
In [86]: learn.sched.plot()
```



Issue with small number of images, try minimum of 50



Wiki: Lesson 1

Part 1 v2

1 Reply ▾



Reply



alessa Alessa Bandrabur



financepk

11d

Regarding 3. - your learning rate schedule which still doesn't work.

Both `plot_lr()` and `plot()`, are using the samples from the training dataset.

So this means that for `plot_lr()` you will have `number_of_iterations = training_dataset_size / batch_size = 150/15 = 10`. When I read your first graph I can see that you only have 5-6 iterations. You can double check the sizes by printing these `print(learn.data.bs)` and `print(len(learn.data.trn_y))`

In case of `plot()` method, you want to plot the learning rate against the loss. The two variables have the same length equal to the `number_of_iterations`. But the `plot()` methods cuts by default the first 10 values and the last 5 values. So if you want to use the function as it is, you will need at least 17 iterations in order to plot a 2 points line. Or you can call the function specifying to start from 0 `learn.sched.plot(n_skip=0)` but you will still need a minimum of 7 iterations.

Probably the best/easier would be to decrease the batch size in order to be able to display these graphs.

$bs = 3$

- Who are the Dummies?
- General Steps
- Downloading Images : The Challenge
- **How to look for help**
- Downloading Images
- Cleanup
- Renaming and renumbering
- Transfer to cloud
- Train your model
- Fine tune your model


Download images : How? (1) Search forum.fast.ai

How? Search download imgs at forums.fast.ai


► Advanced Search

+ New Topic

50 results for **download images**Sort by Relevance▼



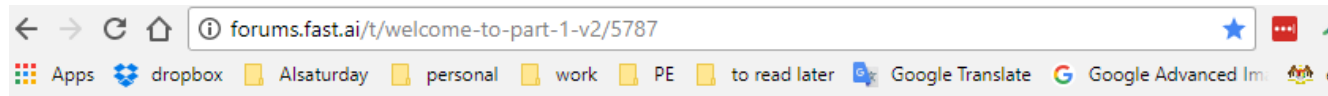
Can not [download IMAGENET images](#) with free (non academic) email address
■ Part 2
Jun '17 - Hi, I am trying to **download** all Imagenet **images** onto my box. But the website is giving me an error after I fill out the form and hit submit request: "We will not approve request..."



License issue of the [images download](#) by google
■ Part 1
Jun '17 - As the assignment suggest(start your own project), collecting **images** from google search and use them to train a model, would this imposed any restrictions for the trained model? If I want to release the collected data(for research purpose), what should I notice?Would I get any troubles from the v...

Download images : How? (2) Use lessons wiki

<http://forums.fast.ai/t/welcome-to-part-1-v2/5787>




Welcome to Part 1 (v2)

Part 1 v2



jeremy  Jeremy Howard

5  Sep '17

Welcome to v2 of Deep Learning Part 1! We are currently doing a "soft launch" of the course videos. Feel free to discuss and share amongst your friends, but don't post the videos to high volume sites like twitter, HN, reddit, etc until we launch officially - since I want new people to get the best possible experience. Here are all the lessons:

- Lesson 1: [Video](#) 2.5k, [Wiki thread](#) 1.8k
- Lesson 2: [Video](#) 776, [Wiki thread](#) 564
- Lesson 3: [Video](#) 397, [Wiki thread](#) 383
- Lesson 4: [Video](#) 292, [Wiki thread](#) 262
- Lesson 5: [Video](#) 195, [Wiki thread](#) 183
- Lesson 6: [Video](#) 163, [Wiki thread](#) 167
- Lesson 7: [Video](#) 201, [Wiki thread](#) 196

And here are threads to help you with specific issues:

- [Paperspace setup](#)

Finally, here's [the introduction thread](#) 411 where you can get to know the other students and introduce yourself.

How to download images. I found a python script



Wiki: Lesson 1

■ Part 1 v2



ecdrid Aditya



financepk

22d

Use this to download images(it works like a charm...
Tried and tested)

github.com 49



[hardikvasa/google-images-download](#)

google-images-download - Python Script to download hundreds of images from 'Google Images'. It is a ready-to-run code!

5 Likes

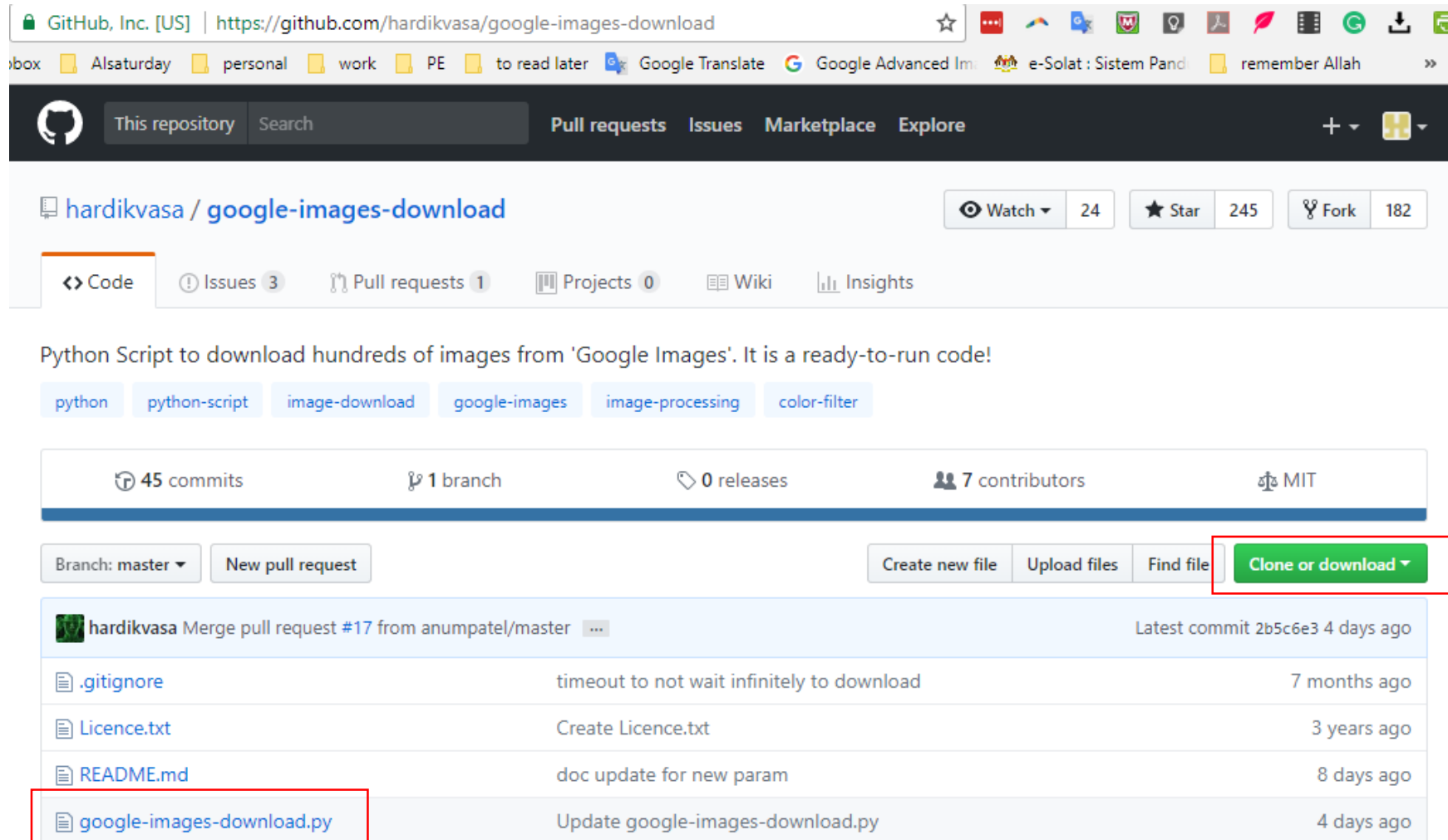


Reply

- Who are the Dummies?
- General Steps
- Downloading Images : The Challenge
- How to look for help
- **Downloading Images**
- Cleanup
- Renaming and renumbering
- Transfer to cloud
- Train your model
- Fine tune your model

Downloading images : Python Program

<https://github.com/hardikvasa/google-images-download>



The screenshot shows the GitHub repository page for `hardikvasa/google-images-download`. The repository is a Python script to download images from Google Images. The page includes a navigation bar with links to Pull requests, Issues, Marketplace, and Explore. The repository statistics show 24 watches, 245 stars, and 182 forks. The repository description states: "Python Script to download hundreds of images from 'Google Images'. It is a ready-to-run code!". The repository has 45 commits, 1 branch, 0 releases, 7 contributors, and is licensed under MIT. The file list shows the following files:

| File | Description | Time |
|--|--|--------------|
| <code>.gitignore</code> | timeout to not wait infinitely to download | 7 months ago |
| <code>Licence.txt</code> | Create Licence.txt | 3 years ago |
| <code>README.md</code> | doc update for new param | 8 days ago |
| <code>google-images-download.py</code> | Update google-images-download.py | 4 days ago |

The `google-images-download.py` file is highlighted with a red box. The `Clone or download` button is also highlighted with a red box.

Run the image downloader

C:\> Google Cloud SDK Shell

Go to the directory where the python file is

```
Welcome to the Google Cloud SDK! Run "gcloud -h" to get the list of available commands.
---
C:\Users\ACER\AppData\Local\Google\Cloud SDK>cd ../../../../Downloads/google-images-download-master
C:\Users\ACER\Downloads\google-images-download-master>
```

-k is keyword, -l is limit (how many)

```
C:\Users\ACER\Downloads\google-images-download-master>python google-images-download.py -k "superman" -l 120
```

```
Item no.: 1 --> Item name = superman
```

```
Evaluating...
```

```
Total Image Links = 100
```

```
Total time taken: 15.81716513633728 Seconds
```

```
Starting Download...
```

```
completed ====> 1. 250px-SupermanRoss.png
```

```
completed ====> 2. lead_960.jpg
```

```
completed ====> 3. 1330222e9a6df3b80d8c21ecffc0c600._SX1280_QL80_TTD_.jpg
```

```
completed ====> 4. GalleryChar_1900x900_MOS_52e05e3fe24a61.04593858.jpg
```

```
completed ====> 5. superman-returns.jpg
```

```
completed ====> 6. maxresdefault.jpg
```

```
completed ====> 7. dc-comics-batman-v-superman-superman-half-scale-polystone-statue-prime-1-902664-01.jpg
```

```
completed ====> 8. 71BDjMoIXYL._SL1500_.jpg
```

```
completed ====> 9. 170px-Superman_S_symbol.svg.png
```

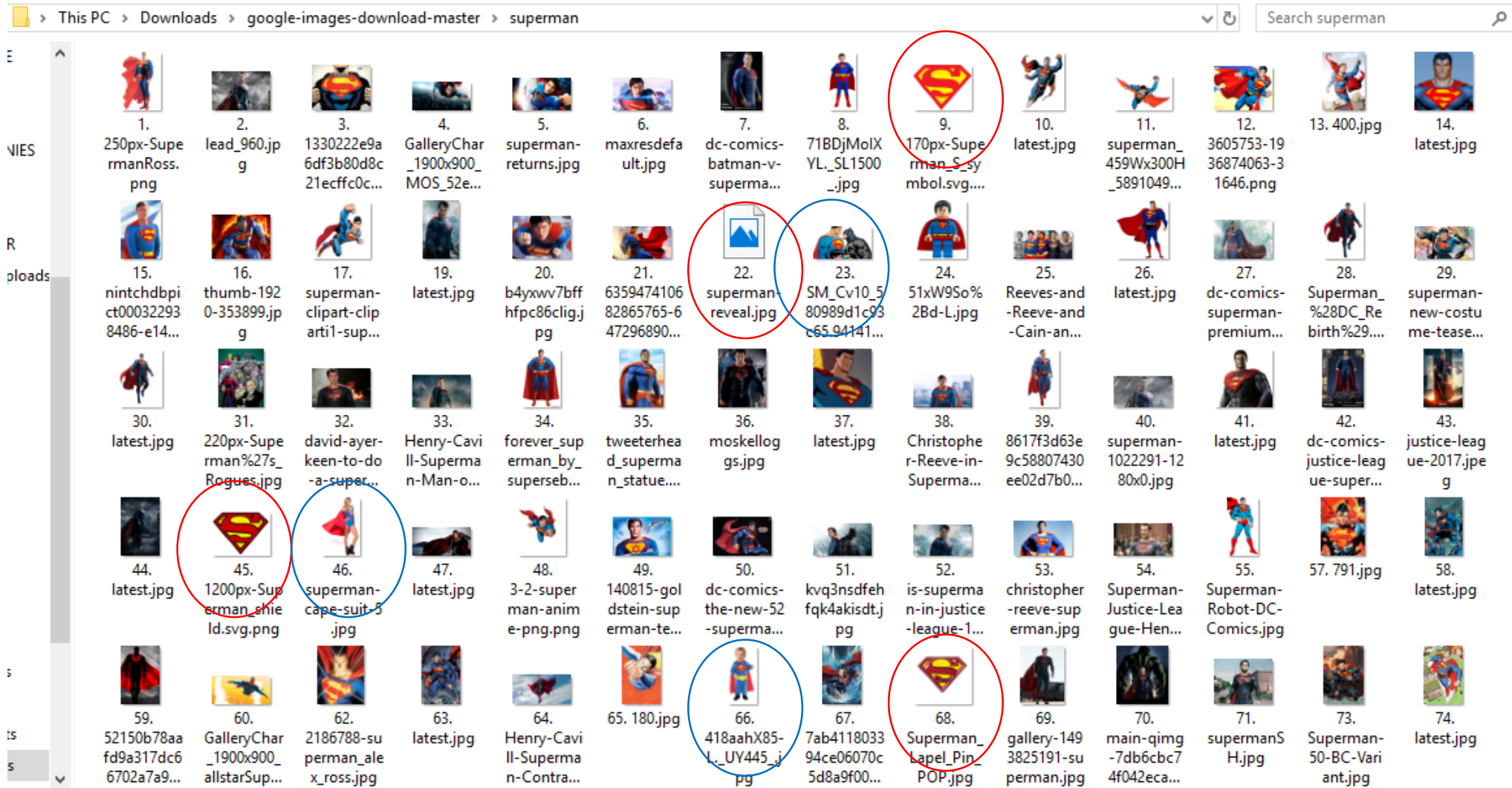
```
completed ====> 10. latest.jpg
```

Somehow the limit is 100, google limit?

Repeat for "bird", "cats" or "dogs" or whatever

- Who are the Dummies?
- General Steps
- Downloading Images : The Challenge
- How to look for help
- Downloading Images
- **Cleanup**
- Renaming and renumbering
- Transfer to cloud
- Train your model
- Fine tune your model

Delete non-relevant data



Repeat for bird

- Who are the Dummies?
- General Steps
- Downloading Images : The Challenge
- How to look for help
- Downloading Images
- Cleanup
- **Renaming and renumbering**
- Transfer to cloud
- Train your model
- Fine tune your model

Renaming Files? Use my python script

<https://github.com/zaimawang/cnnsuperbird>

GitHub, Inc. [US] | <https://github.com/zaimawang/cnnsuperbird>

CodeIssues0Pull requests0Projects0WikiInsightsSettings

cnn fast ai using superman bird data set I downloaded from google

Edit

Add topics

13 commits1 branch0 releases1 contributor

Branch: masterNew pull request

Create new fileUpload filesFind fileClone or download

zaimawang Update README.mdLatest commit 286ea6b 5 days ago

| | | |
|---------------------------|----------------------|------------|
| dataset_sb | Delete test.py | 5 days ago |
| README.md | Update README.md | 5 days ago |
| cnnsuperbird.ipynb | Add files via upload | 5 days ago |
| google-images-download.py | Add files via upload | 5 days ago |
| renumberfile_bird.py | Add files via upload | 5 days ago |
| renumberfile_superman.py | Add files via upload | 5 days ago |
| README.md | | |

Set your path and filename (superman/bird)

```
renumberfile_superman.py x google-images-download.py x
1  # os is a library that gives us the ability to make OS changes
2  import os
3
4
5  def increment_file_numbers(directory, string, integer):
6      # iterate over every file name in the directory
7      print (directory)
8      file_number = startcount
9      for file_name in os.listdir(directory):
10         file_number = file_number + 1
11         extension = ".jpg"
12         file_name2 = objectname+"."
13         # example format : superman.01.jpg
14         new_file_name = '%s/%s%02d%s' % (directory, file_name2, file_number, extension)
15         old_file_name = '%s/%s' % (directory, file_name)
16         print (old_file_name)
17         print (new_file_name)
18         # rename the file!
19         os.rename(old_file_name, new_file_name)
20
21
22  # This is the path to the files
23  # C:\Users\ACER\Downloads\google-images-download-master\superman
24  PATH = os.path.abspath('/Users/ACER/Downloads/google-images-download-master/superman')
25  objectname = "superman"
26  startcount = 0
27
28  # Let's rename the files
29  increment_file_numbers(PATH, objectname, startcount)
30
```

Edit path and filename

```
C:\Users\ACER\Downloads\google-images-download-master>python renumberfile_superman.py
```


Files have been renamed and renumbered

This PC > Downloads > google-images-download-master > data_test_sb > train > superman

Search superman



superman.21.jpg



superman.22.jpg



superman.23.jpg



superman.24.jpg



superman.25.jpg



superman.26.jpg



superman.27.jpg



superman.28.jpg



Move half to a new folder called **train/superman** and another half to **valid/superman**

- Your Ipythonnotebook.ipynb sits in the same folder as superbirddata folder
- superbirddata folder
 - Train
 - superman
 - Bird
 - Valid
 - superman
 - Bird

Similarly for bird do the same

is PC > Downloads > google-images-download-master > data_test_sb > valid > bird



Search bird



bird.86.jpg



bird.87.jpg



bird.88.jpg



bird.89.jpg



bird.90.jpg



bird.91.jpg



bird.92.jpg



bird.93.jpg



bird.94.jpg



bird.95.jpg



bird.96.jpg

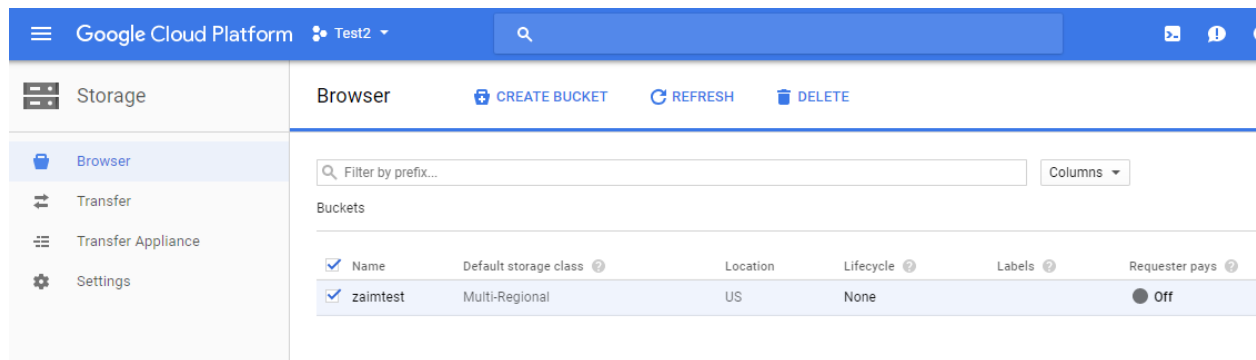


bird.97.jpg

- Who are the Dummies?
- General Steps
- Downloading Images : The Challenge
- How to look for help
- Downloading Images
- Cleanup
- Renaming and renumbering
- **Transfer to cloud**
- Train your model
- Fine tune your model

File transfer method 1 : (bucket and gsutil)

- But images now in pc but need to have it in the cloud
- There are a few ways to do it
 - Upload files to google cloud BUCKET, but create a bucket first (hint: search for bucket)

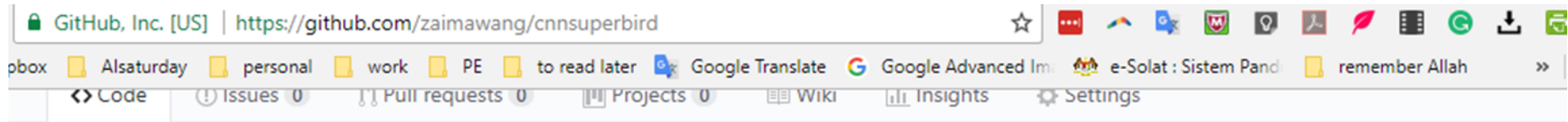


- AND GSUTIL <https://cloud.google.com/storage/docs/gsutil/commands/cp>

```
(fastai) ACER@fastai-instance-1:~/fastai/courses/dll$ gsutil cp -r gs://zaimtest .
```

Your jupyter notebook doesn't have access to the bucket, that is why we gsutil the the bucket to where jupyter notebook can see it

File transfer method 2 : (put in github and then git clone)



cnnsuperbird fast ai using superman bird data set I downloaded from google

Edit

[Add topics](#)

13 commits

1 branch

0 releases

1 contributor

Branch: master ▼








[New pull request](#)


[Create new file](#)

[Upload files](#)

[Find file](#)

[Clone or download ▼](#)

| | |
|--|----------------------------------|
|  zaimawang Update README.md | Latest commit 286ea6b 5 days ago |
|  dataset_sb | Delete test.py 5 days ago |
|  README.md | Update README.md 5 days ago |
|  cnnsuperbird.ipynb | Add files via upload 5 days ago |
|  google-images-download.py | Add files via upload 5 days ago |
|  renumberfile_bird.py | Add files via upload 5 days ago |
|  renumberfile_superman.py | Add files via upload 5 days ago |

 README.md

Git cloning

```
ACER@fastai-instance-1: ~  
(fastai) ACER@fastai-instance-1:~$ git clone https://github.com/zaimawang/cnnsuperbird  
Cloning into 'cnnsuperbird'...  
remote: Counting objects: 223, done.  
remote: Compressing objects: 100% (9/9), done.  
remote: Total 223 (delta 3), reused 0 (delta 0), pack-reused 214  
Receiving objects: 100% (223/223), 56.08 MiB | 12.67 MiB/s, done.  
Resolving deltas: 100% (12/12), done.  
Checking connectivity... done.  
(fastai) ACER@fastai-instance-1:~$ ls  
anaconda3  cnnsuperbird  data  downloads  fastai  
(fastai) ACER@fastai-instance-1:~$
```

Normally you want to git clone under fast/courses/dl1

File transfer method 3: Use your jupyter notebook

<https://stackoverflow.com/questions/34734714/ipython-jupyter-uploading-folder>

1 You could upload a zip file, then start a notebook and use the `zipfile` module from Python to extract it. – Thomas K Jan 12 '16 at 12:01

add a comment

1 Answer

active

oldest

votes

▲ Convert it into a single Zip file and upload that. to unzip the folder use the code down bellow

5

▼

```
import zipfile as zf
files = zf.ZipFile("ZippedFolder.zip", 'r')
files.extractall('directory to extract')
files.close()
```

Files Running Clusters

Upload a zip file to your Jupyter notebook
The zip file is actually your folder

Upload New ↕ ↻

Select items to perform actions on them.

0 / fastai / courses / dl1

Name ↓ Last Modified

A few other methods not discussed here

- Who are the Dummies?
- General Steps
- Downloading Images : The Challenge
- How to look for help
- Downloading Images
- Cleanup
- Renaming and renumbering
- Transfer to cloud
- Train your model
- Fine tune your model

Run your jupyter notebook

jupyter cnnsuperbird Last Checkpoint: Last Sunday at 9:16 PM (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Save New Open Recent Up Down Run Stop Restart Code

```
In [1]: # Put these at the top of every notebook, to get automatic reloading and inline plotting
%reload_ext autoreload
%autoreload 2
%matplotlib inline
```

```
In [2]: # This file contains all the main external libs we'll use
from fastai.imports import *
from fastai.transforms import *
from fastai.conv_learner import *
from fastai.model import *
from fastai.dataset import *
from fastai.sgdr import *
from fastai.plots import *
```

```
In [6]: PATH = "cnnsuperbird/dataset_sb/"
```

Set your path where you have your pictures

Set small batch size due to small sample image

```
jupyter cnnsuperbird Last Checkpoint: Last Sunday at 9:16 PM (autosaved)
```

File Edit View Insert Cell Kernel Widgets Help

In [12]: `# Uncomment the below if you need to reset your precomputed activations
!rm -rf (PATH)temp`

In [13]: `bs=5`

In [14]: `tfms = tfms_from_model(resnet34, sz, aug_tfms=transforms_side_on, max_zoom=1.1)`

In [15]: `arch=resnet34
data = ImageClassifierData.from_paths(PATH, bs, tfms)`

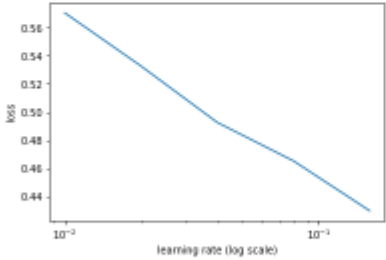
In [16]: `learn = ConvLearner.pretrained(arch, data, precompute=True)`

In [22]: `learn.lr_find()`

Epoch 100% 1/1 [00.03<00.00, 3.11s/it]

[0. 0.51294 669.55872 0.91]

In [23]: `learn.sched.plot()`



In [25]: `learn.sched.plot_lr()`



Why Superman and bird is not exactly the best choice

- Images are too different?
- Easy for cnn to distinguish

Improving accuracy

```
In [26]: learn = ConvLearner.pretrained(arch, data, precompute=True)
learn.fit(0.01, 4)
```

Epoch  100% 4/4 [00:00<00:00, 6.94B/s]

```
[ 0.      0.34434  0.83489  0.99   ]
[ 1.      0.15373  0.83279  0.99   ]
[ 2.      0.09719  0.82931  0.99   ]
[ 3.      0.09749  0.83868  0.99   ]
```

```
In [28]: learn.unfreeze()
```

```
In [29]: lr=np.array([0.0001,0.001,0.01])
```

```
In [32]: learn.fit(lr,3,cycle_len=1,cycle_mult=1)
```

Epoch  100% 3/3 [00:10<00:00, 3.49s/it]

```
[ 0.      0.2933   0.82867  0.99   ]
[ 1.      0.18966  0.8387   0.99   ]
[ 2.      0.16883  0.81245  1.     ]
```

```
In [33]: learn.save('model')
```

```
In [34]: learn.load('model')
```

Correctly Classified

```
In [44]: # 1. A few correct labels at random:  
plot_val_with_title(rand_by_correct(True), "Correctly classified")  
Correctly classified
```

1.78665e-05



0.998975



0.981099



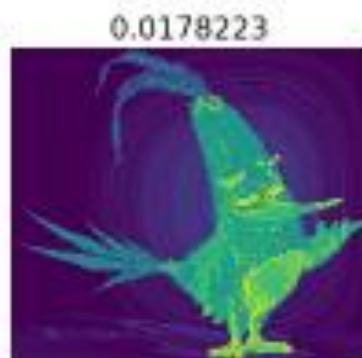
1.71691e-05



Most uncertain prediction

```
In [166]: most_uncertain = np.argsort(np.abs(probs - 0.5))[:4]  
plot_val_with_title(most_uncertain, "Most uncertain predictions")
```

Most uncertain predictions:



Not as accurate model (95%)

```
In [57]: # 2. A few incorrect labels at random  
plot_val_with_title(rand_by_correct(False), "Incorrectly classified")
```

Incorrectly classified

1.0



0.999995



1.0



0.589813



Summary: Some of the things to try

- Batch size
- data augmentation / image transformation
- Differential learning rate
- Multi length cycle, TTA
- Add another category?

Where you can find this file?

- https://github.com/zaimawang/cnnsuperbird/share_cnn_superbird.pdf

THANK YOU