

# 0\_Dataset

June 1, 2020

## 1 Computer Vision Nanodegree

### 1.1 Project: Image Captioning

---

The Microsoft Common Objects in COntext (MS COCO) dataset is a large-scale dataset for scene understanding. The dataset is commonly used to train and benchmark object detection, segmentation, and captioning algorithms.

You can read more about the dataset on the [website](#) or in the [research paper](#).

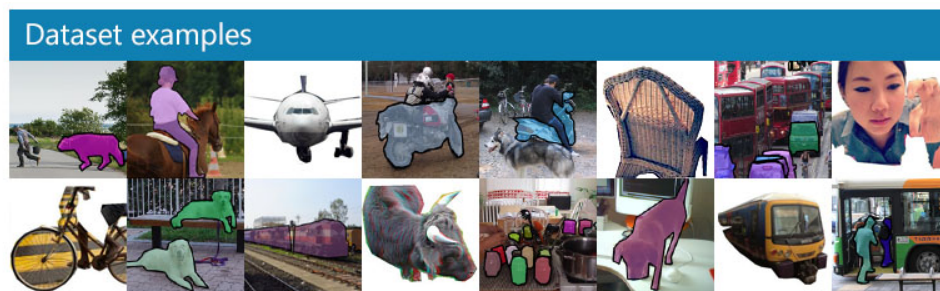
In this notebook, you will explore this dataset, in preparation for the project.

### 1.2 Step 1: Initialize the COCO API

We begin by initializing the [COCO API](#) that you will use to obtain the data.

```
In [1]: import os
import sys
sys.path.append('/opt/cocoapi/PythonAPI')
from pycocotools.coco import COCO

# initialize COCO API for instance annotations
dataDir = '/opt/cocoapi'
dataType = 'val2014'
instances_annFile = os.path.join(dataDir, 'annotations/instances_{}.json'.format(dataType))
coco = COCO(instances_annFile)
```



Sample Dog Output

```

# initialize COCO API for caption annotations
captions_annFile = os.path.join(dataDir, 'annotations/captions_{}.json'.format(dataType))
coco_caps = COCO(captions_annFile)

# get image ids
ids = list(coco.anns.keys())

loading annotations into memory...
Done (t=6.18s)
creating index...
index created!
loading annotations into memory...
Done (t=0.46s)
creating index...
index created!

```

### 1.3 Step 2: Plot a Sample Image

Next, we plot a random image from the dataset, along with its five corresponding captions. Each time you run the code cell below, a different image is selected.

In the project, you will use this dataset to train your own model to generate captions from images!

```

In [2]: import numpy as np
import skimage.io as io
import matplotlib.pyplot as plt
%matplotlib inline

# pick a random image and obtain the corresponding URL
ann_id = np.random.choice(ids)
img_id = coco.anns[ann_id]['image_id']
img = coco.loadImgs(img_id)[0]
url = img['coco_url']

# print URL and visualize corresponding image
print(url)
I = io.imread(url)
plt.axis('off')
plt.imshow(I)
plt.show()

# load and display captions
annIds = coco_caps.getAnnIds(imgIds=img['id']);
anns = coco_caps.loadAnns(annIds)
coco_caps.showAnns(anns)

```

[http://images.cocodataset.org/val2014/COCO\\_val2014\\_000000052376.jpg](http://images.cocodataset.org/val2014/COCO_val2014_000000052376.jpg)



Two women comparing info on their phones in the room  
Two women looking down at their cell phones.  
Two women standing next to each other looking at a phone.  
A couple of people standing with a cell phone.  
Two woman showing each other something on their phones

## 1.4 Step 3: What's to Come!

In this project, you will use the dataset of image-caption pairs to train a CNN-RNN model to automatically generate images from captions. You'll learn more about how to design the architecture in the next notebook in the sequence (**1\_Preliminaries.ipynb**).

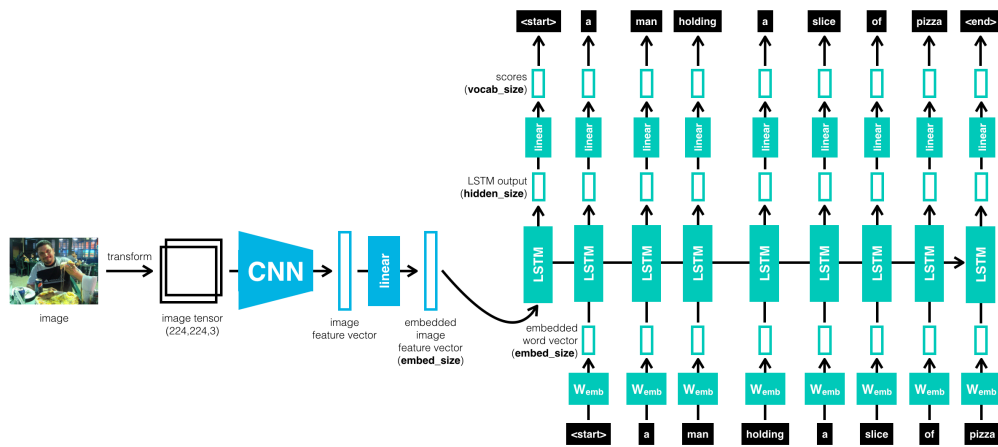


Image Captioning CNN-RNN model