

```
In [1]: import cv2
from glob import glob
import matplotlib.pyplot as plt
from os import listdir
```

```
In [2]: #First step is to read images from the Dataset. In this project i am using Flickr-8k Data set
#In Flickr-8k data set 6000 images are for training,1000 images are for validation and 1000 images are for testing
```

```
In [3]: images_data=glob('C:/Users/mahid/M.Tech Project/Data Set/Images/*')
```

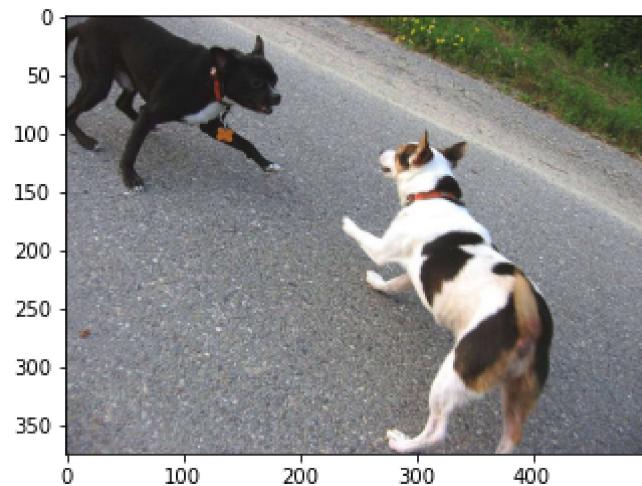
```
In [4]: count=0
for i in images_data:
    if(count<10):
        print(i)
    count+=1
```

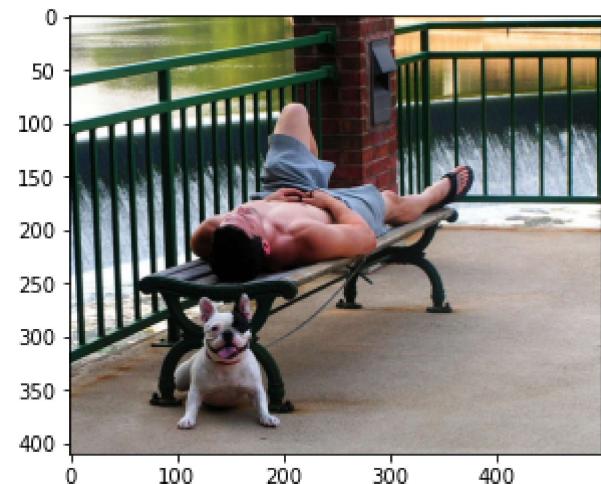
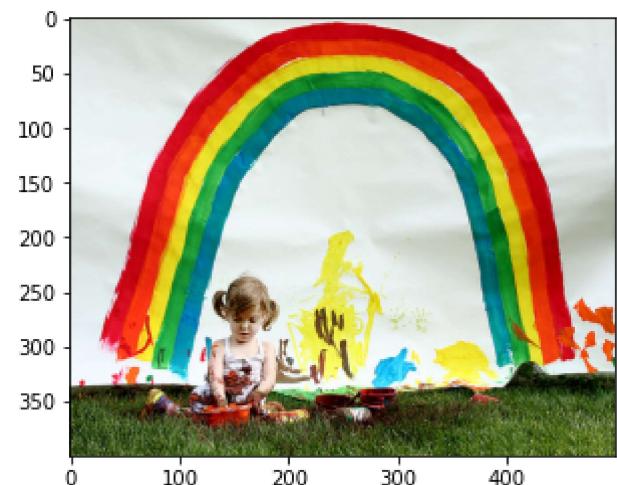
```
C:/Users/mahid/M.Tech Project/Data Set/Images\1000268201_693b08cb0e.jpg
C:/Users/mahid/M.Tech Project/Data Set/Images\1001773457_577c3a7d70.jpg
C:/Users/mahid/M.Tech Project/Data Set/Images\1002674143_1b742ab4b8.jpg
C:/Users/mahid/M.Tech Project/Data Set/Images\1003163366_44323f5815.jpg
C:/Users/mahid/M.Tech Project/Data Set/Images\1007129816_e794419615.jpg
C:/Users/mahid/M.Tech Project/Data Set/Images\1007320043_627395c3d8.jpg
C:/Users/mahid/M.Tech Project/Data Set/Images\1009434119_febe49276a.jpg
C:/Users/mahid/M.Tech Project/Data Set/Images\1012212859_01547e3f17.jpg
C:/Users/mahid/M.Tech Project/Data Set/Images\1015118661_980735411b.jpg
C:/Users/mahid/M.Tech Project/Data Set/Images\1015584366_dfcec3c85a.jpg
```

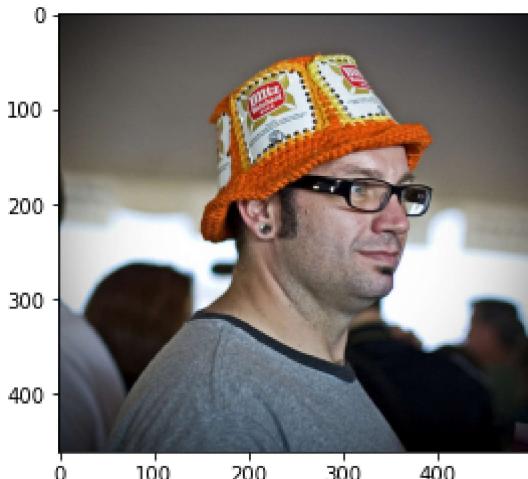
```
In [5]: image_shape=cv2.imread(images_data[0])
print(image_shape.shape)
```

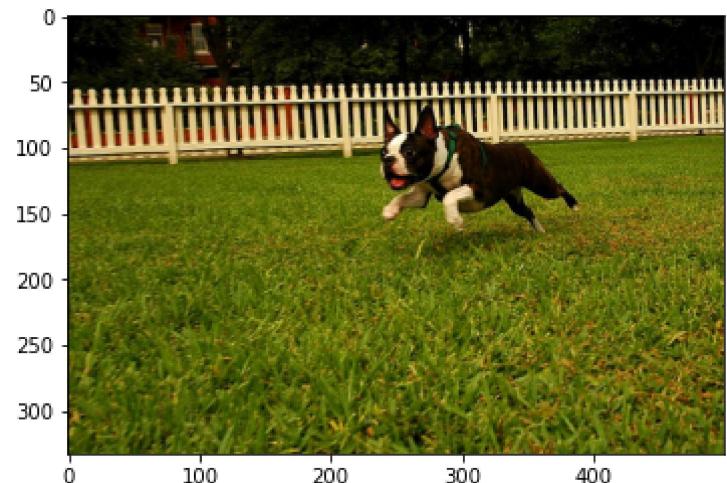
```
(500, 375, 3)
```

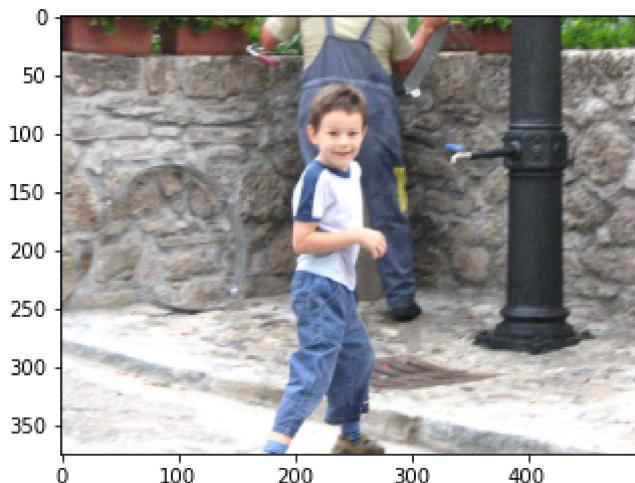
```
In [6]: for i in range(10):
    image=cv2.imread(images_data[i])
    image=cv2.cvtColor(image,cv2.COLOR_BGR2RGB)
    plt.figure()
    plt.imshow(image)
```











```
In [7]: #Now we need to pre process the text data as well but giving unique numbers to each unique word in data  
#Because Neural networks can work only on numerical values
```

```
In [8]: captions_data_path = 'C:/Users/mahid/M.Tech Project/Data Set/Flickr8k_text/Flickr8k.token.txt'
```

```
In [9]: #In captions document the format is Image name #no Caption of that Image name  
#for each image there are 5 different captions indicating similar meaning to the image  
#EX: 1000268201_693b08cb0e.jpg#0 A child in a pink dress is climbing up a set of stairs in an entry way .
```

```
In [10]: captions_data = open(captions_data_path, 'rb').read()
captions_data=captions_data.decode('utf-8').split('\n')
```

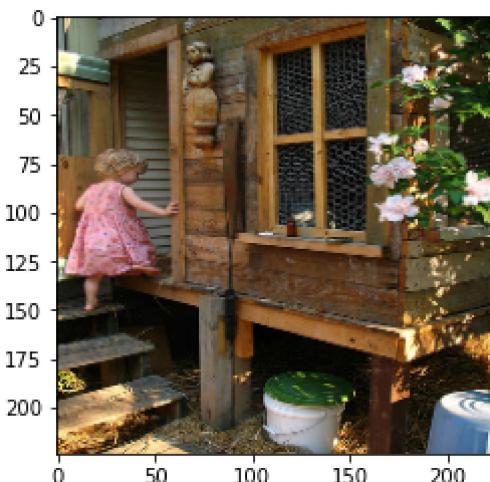
```
In [11]: len(captions_data)
```

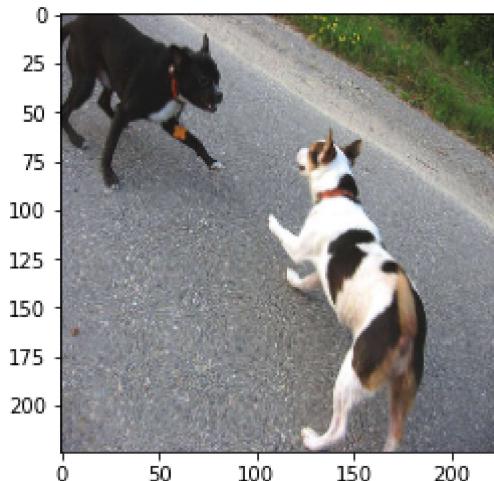
```
Out[11]: 40456
```

```
In [12]: captions_data[0]
```

```
Out[12]: '1000268201_693b08cb0e.jpg#0\tA child in a pink dress is climbing up a set of stairs in an entry way .'
```

```
In [13]: for i in range(2):
    image=cv2.imread(images_data[i])
    image=cv2.cvtColor(image,cv2.COLOR_BGR2RGB)
    image=cv2.resize(image,(224,224))
    image=image.reshape(224,224,3)
    plt.figure()
    plt.imshow(image)
```





```
In [14]: for i in range(10):
    print(captions_data[i])
```

1000268201_693b08cb0e.jpg#0	A child in a pink dress is climbing up a set of stairs in an entry way .
1000268201_693b08cb0e.jpg#1	A girl going into a wooden building .
1000268201_693b08cb0e.jpg#2	A little girl climbing into a wooden playhouse .
1000268201_693b08cb0e.jpg#3	A little girl climbing the stairs to her playhouse .
1000268201_693b08cb0e.jpg#4	A little girl in a pink dress going into a wooden cabin .
1001773457_577c3a7d70.jpg#0	A black dog and a spotted dog are fighting
1001773457_577c3a7d70.jpg#1	A black dog and a tri-colored dog playing with each other on the road .
1001773457_577c3a7d70.jpg#2	A black dog and a white dog with brown spots are staring at each other in the street .
1001773457_577c3a7d70.jpg#3	Two dogs of different breeds looking at each other on the road .
1001773457_577c3a7d70.jpg#4	Two dogs on pavement moving toward each other .

```
In [16]: from pickle import dump
from keras.applications.xception import Xception
from keras.models import Model
```

```
In [18]: model=Xception(include_top=True,weights='imagenet')
lastlayer=model.layers[-2].output
model=Model(inputs=model.input,outputs=lastlayer)
print(model.summary())
```

Downloading data from [https://storage.googleapis.com/tensorflow/keras-applications/xception/xception\\_weights\\_tf\\_dim\\_ordering\\_tf\\_kernels.h5](https://storage.googleapis.com/tensorflow/keras-applications/xception/xception_weights_tf_dim_ordering_tf_kernels.h5)

91889664/91884032 [=====] - 14s 0us/step

91897856/91884032 [=====] - 14s 0us/step

Model: "model"

Layer (type)	Output Shape	Param #	Connected to
<hr/>			
input_1 (InputLayer)	[(None, 299, 299, 3 0 )]	[ ]	
block1_conv1 (Conv2D)	(None, 149, 149, 32 864 )	[ 'input_1[0][0]' ]	
block1_conv1_bn (BatchNormalization)	(None, 149, 149, 32 128 )	[ 'block1_conv1[0][0]' ]	
block1_conv1_act (Activation)	(None, 149, 149, 32 0 )	[ 'block1_conv1_bn[0][0]' ]	
block1_conv2 (Conv2D)	(None, 147, 147, 64 18432 )	[ 'block1_conv1_act[0][0]' ]	
block1_conv2_bn (BatchNormalization)	(None, 147, 147, 64 256 )	[ 'block1_conv2[0][0]' ]	
block1_conv2_act (Activation)	(None, 147, 147, 64 0 )	[ 'block1_conv2_bn[0][0]' ]	
block2_sepconv1 (SeparableConv2D)	(None, 147, 147, 12 8768 )	[ 'block1_conv2_act[0][0]' ]	
block2_sepconv1_bn (BatchNormalization)	(None, 147, 147, 12 512 )	[ 'block2_sepconv1[0][0]' ]	
block2_sepconv2_act (Activation)	(None, 147, 147, 12 0 )	[ 'block2_sepconv1_bn[0][0]' ]	
block2_sepconv2 (SeparableConv2D)	(None, 147, 147, 12 17536 )	[ 'block2_sepconv2_act[0][0]' ]	
block2_sepconv2_bn (BatchNormalization)	(None, 147, 147, 12 512 )	[ 'block2_sepconv2[0][0]' ]	
conv2d (Conv2D)	(None, 74, 74, 128) 8192	[ 'block1_conv2_act[0][0]' ]	

block2_pool (MaxPooling2D)	(None, 74, 74, 128)	0	['block2_sepconv2_bn[0][0]']
batch_normalization (BatchNorm alization)	(None, 74, 74, 128)	512	['conv2d[0][0]']
add (Add)	(None, 74, 74, 128)	0	['block2_pool[0][0]', 'batch_normalization[0][0]']
block3_sepconv1_act (Activatio n)	(None, 74, 74, 128)	0	['add[0][0]']
block3_sepconv1 (SeparableConv 2D)	(None, 74, 74, 256)	33920	['block3_sepconv1_act[0][0]']
block3_sepconv1_bn (BatchNorma lization)	(None, 74, 74, 256)	1024	['block3_sepconv1[0][0]']
block3_sepconv2_act (Activatio n)	(None, 74, 74, 256)	0	['block3_sepconv1_bn[0][0]']
block3_sepconv2 (SeparableConv 2D)	(None, 74, 74, 256)	67840	['block3_sepconv2_act[0][0]']
block3_sepconv2_bn (BatchNorma lization)	(None, 74, 74, 256)	1024	['block3_sepconv2[0][0]']
conv2d_1 (Conv2D)	(None, 37, 37, 256)	32768	['add[0][0]']
block3_pool (MaxPooling2D)	(None, 37, 37, 256)	0	['block3_sepconv2_bn[0][0]']
batch_normalization_1 (BatchNo rmalization)	(None, 37, 37, 256)	1024	['conv2d_1[0][0]']
add_1 (Add)	(None, 37, 37, 256)	0	['block3_pool[0][0]', 'batch_normalization_1[0][0]']
block4_sepconv1_act (Activatio n)	(None, 37, 37, 256)	0	['add_1[0][0]']
block4_sepconv1 (SeparableConv 2D)	(None, 37, 37, 728)	188672	['block4_sepconv1_act[0][0]']
block4_sepconv1_bn (BatchNorma lization)	(None, 37, 37, 728)	2912	['block4_sepconv1[0][0]']

block4_sepconv2_act (Activation)	(None, 37, 37, 728)	0	['block4_sepconv1_bn[0][0]']
block4_sepconv2 (SeparableConv 2D)	(None, 37, 37, 728)	536536	['block4_sepconv2_act[0][0]']
block4_sepconv2_bn (BatchNormalization)	(None, 37, 37, 728)	2912	['block4_sepconv2[0][0]']
conv2d_2 (Conv2D)	(None, 19, 19, 728)	186368	['add_1[0][0]']
block4_pool (MaxPooling2D)	(None, 19, 19, 728)	0	['block4_sepconv2_bn[0][0]']
batch_normalization_2 (BatchNormalization)	(None, 19, 19, 728)	2912	['conv2d_2[0][0]']
add_2 (Add)	(None, 19, 19, 728)	0	['block4_pool[0][0]', 'batch_normalization_2[0][0]']
block5_sepconv1_act (Activation)	(None, 19, 19, 728)	0	['add_2[0][0]']
block5_sepconv1 (SeparableConv 2D)	(None, 19, 19, 728)	536536	['block5_sepconv1_act[0][0]']
block5_sepconv1_bn (BatchNormalization)	(None, 19, 19, 728)	2912	['block5_sepconv1[0][0]']
block5_sepconv2_act (Activation)	(None, 19, 19, 728)	0	['block5_sepconv1_bn[0][0]']
block5_sepconv2 (SeparableConv 2D)	(None, 19, 19, 728)	536536	['block5_sepconv2_act[0][0]']
block5_sepconv2_bn (BatchNormalization)	(None, 19, 19, 728)	2912	['block5_sepconv2[0][0]']
block5_sepconv3_act (Activation)	(None, 19, 19, 728)	0	['block5_sepconv2_bn[0][0]']
block5_sepconv3 (SeparableConv 2D)	(None, 19, 19, 728)	536536	['block5_sepconv3_act[0][0]']

block5_sepconv3_bn (BatchNorma lization)	(None, 19, 19, 728) 2912	['block5_sepconv3[0][0]']
add_3 (Add)	(None, 19, 19, 728) 0	['block5_sepconv3_bn[0][0]', 'add_2[0][0]']
block6_sepconv1_act (Activatio n)	(None, 19, 19, 728) 0	['add_3[0][0]']
block6_sepconv1 (SeparableConv 2D)	(None, 19, 19, 728) 536536	['block6_sepconv1_act[0][0]']
block6_sepconv1_bn (BatchNorma lization)	(None, 19, 19, 728) 2912	['block6_sepconv1[0][0]']
block6_sepconv2_act (Activatio n)	(None, 19, 19, 728) 0	['block6_sepconv1_bn[0][0]']
block6_sepconv2 (SeparableConv 2D)	(None, 19, 19, 728) 536536	['block6_sepconv2_act[0][0]']
block6_sepconv2_bn (BatchNorma lization)	(None, 19, 19, 728) 2912	['block6_sepconv2[0][0]']
block6_sepconv3_act (Activatio n)	(None, 19, 19, 728) 0	['block6_sepconv2_bn[0][0]']
block6_sepconv3 (SeparableConv 2D)	(None, 19, 19, 728) 536536	['block6_sepconv3_act[0][0]']
block6_sepconv3_bn (BatchNorma lization)	(None, 19, 19, 728) 2912	['block6_sepconv3[0][0]']
add_4 (Add)	(None, 19, 19, 728) 0	['block6_sepconv3_bn[0][0]', 'add_3[0][0]']
block7_sepconv1_act (Activatio n)	(None, 19, 19, 728) 0	['add_4[0][0]']
block7_sepconv1 (SeparableConv 2D)	(None, 19, 19, 728) 536536	['block7_sepconv1_act[0][0]']
block7_sepconv1_bn (BatchNorma lization)	(None, 19, 19, 728) 2912	['block7_sepconv1[0][0]']

block7_sepconv2_act (Activation)	(None, 19, 19, 728)	0	['block7_sepconv1_bn[0][0]']
block7_sepconv2 (SeparableConv 2D)	(None, 19, 19, 728)	536536	['block7_sepconv2_act[0][0]']
block7_sepconv2_bn (BatchNormalization)	(None, 19, 19, 728)	2912	['block7_sepconv2[0][0]']
block7_sepconv3_act (Activation)	(None, 19, 19, 728)	0	['block7_sepconv2_bn[0][0]']
block7_sepconv3 (SeparableConv 2D)	(None, 19, 19, 728)	536536	['block7_sepconv3_act[0][0]']
block7_sepconv3_bn (BatchNormalization)	(None, 19, 19, 728)	2912	['block7_sepconv3[0][0]']
add_5 (Add)	(None, 19, 19, 728)	0	['block7_sepconv3_bn[0][0]', 'add_4[0][0]']
block8_sepconv1_act (Activation)	(None, 19, 19, 728)	0	['add_5[0][0]']
block8_sepconv1 (SeparableConv 2D)	(None, 19, 19, 728)	536536	['block8_sepconv1_act[0][0]']
block8_sepconv1_bn (BatchNormalization)	(None, 19, 19, 728)	2912	['block8_sepconv1[0][0]']
block8_sepconv2_act (Activation)	(None, 19, 19, 728)	0	['block8_sepconv1_bn[0][0]']
block8_sepconv2 (SeparableConv 2D)	(None, 19, 19, 728)	536536	['block8_sepconv2_act[0][0]']
block8_sepconv2_bn (BatchNormalization)	(None, 19, 19, 728)	2912	['block8_sepconv2[0][0]']
block8_sepconv3_act (Activation)	(None, 19, 19, 728)	0	['block8_sepconv2_bn[0][0]']
block8_sepconv3 (SeparableConv 2D)	(None, 19, 19, 728)	536536	['block8_sepconv3_act[0][0]']

2D)

block8_sepconv3_bn (BatchNorma lization)	(None, 19, 19, 728) 2912	['block8_sepconv3[0][0]']
add_6 (Add)	(None, 19, 19, 728) 0	['block8_sepconv3_bn[0][0]', 'add_5[0][0]']
block9_sepconv1_act (Activatio n)	(None, 19, 19, 728) 0	['add_6[0][0]']
block9_sepconv1 (SeparableConv 2D)	(None, 19, 19, 728) 536536	['block9_sepconv1_act[0][0]']
block9_sepconv1_bn (BatchNorma lization)	(None, 19, 19, 728) 2912	['block9_sepconv1[0][0]']
block9_sepconv2_act (Activatio n)	(None, 19, 19, 728) 0	['block9_sepconv1_bn[0][0]']
block9_sepconv2 (SeparableConv 2D)	(None, 19, 19, 728) 536536	['block9_sepconv2_act[0][0]']
block9_sepconv2_bn (BatchNorma lization)	(None, 19, 19, 728) 2912	['block9_sepconv2[0][0]']
block9_sepconv3_act (Activatio n)	(None, 19, 19, 728) 0	['block9_sepconv2_bn[0][0]']
block9_sepconv3 (SeparableConv 2D)	(None, 19, 19, 728) 536536	['block9_sepconv3_act[0][0]']
block9_sepconv3_bn (BatchNorma lization)	(None, 19, 19, 728) 2912	['block9_sepconv3[0][0]']
add_7 (Add)	(None, 19, 19, 728) 0	['block9_sepconv3_bn[0][0]', 'add_6[0][0]']
block10_sepconv1_act (Activati on)	(None, 19, 19, 728) 0	['add_7[0][0]']
block10_sepconv1 (SeparableCon v2D)	(None, 19, 19, 728) 536536	['block10_sepconv1_act[0][0]']

block10_sepconv1_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	[ 'block10_sepconv1[0][0]' ]
block10_sepconv2_act (Activati on)	(None, 19, 19, 728) 0	[ 'block10_sepconv1_bn[0][0]' ]
block10_sepconv2 (SeparableCon v2D)	(None, 19, 19, 728) 536536	[ 'block10_sepconv2_act[0][0]' ]
block10_sepconv2_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	[ 'block10_sepconv2[0][0]' ]
block10_sepconv3_act (Activati on)	(None, 19, 19, 728) 0	[ 'block10_sepconv2_bn[0][0]' ]
block10_sepconv3 (SeparableCon v2D)	(None, 19, 19, 728) 536536	[ 'block10_sepconv3_act[0][0]' ]
block10_sepconv3_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	[ 'block10_sepconv3[0][0]' ]
add_8 (Add)	(None, 19, 19, 728) 0	[ 'block10_sepconv3_bn[0][0]', 'add_7[0][0]' ]
block11_sepconv1_act (Activati on)	(None, 19, 19, 728) 0	[ 'add_8[0][0]' ]
block11_sepconv1 (SeparableCon v2D)	(None, 19, 19, 728) 536536	[ 'block11_sepconv1_act[0][0]' ]
block11_sepconv1_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	[ 'block11_sepconv1[0][0]' ]
block11_sepconv2_act (Activati on)	(None, 19, 19, 728) 0	[ 'block11_sepconv1_bn[0][0]' ]
block11_sepconv2 (SeparableCon v2D)	(None, 19, 19, 728) 536536	[ 'block11_sepconv2_act[0][0]' ]
block11_sepconv2_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	[ 'block11_sepconv2[0][0]' ]
block11_sepconv3_act (Activati on)	(None, 19, 19, 728) 0	[ 'block11_sepconv2_bn[0][0]' ]

block11_sepconv3 (SeparableCon v2D)	(None, 19, 19, 728) 536536	['block11_sepconv3_act[0][0]']
block11_sepconv3_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	['block11_sepconv3[0][0]']
add_9 (Add)	(None, 19, 19, 728) 0	['block11_sepconv3_bn[0][0]', 'add_8[0][0]']
block12_sepconv1_act (Activati on)	(None, 19, 19, 728) 0	['add_9[0][0]']
block12_sepconv1 (SeparableCon v2D)	(None, 19, 19, 728) 536536	['block12_sepconv1_act[0][0]']
block12_sepconv1_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	['block12_sepconv1[0][0]']
block12_sepconv2_act (Activati on)	(None, 19, 19, 728) 0	['block12_sepconv1_bn[0][0]']
block12_sepconv2 (SeparableCon v2D)	(None, 19, 19, 728) 536536	['block12_sepconv2_act[0][0]']
block12_sepconv2_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	['block12_sepconv2[0][0]']
block12_sepconv3_act (Activati on)	(None, 19, 19, 728) 0	['block12_sepconv2_bn[0][0]']
block12_sepconv3 (SeparableCon v2D)	(None, 19, 19, 728) 536536	['block12_sepconv3_act[0][0]']
block12_sepconv3_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	['block12_sepconv3[0][0]']
add_10 (Add)	(None, 19, 19, 728) 0	['block12_sepconv3_bn[0][0]', 'add_9[0][0]']
block13_sepconv1_act (Activati on)	(None, 19, 19, 728) 0	['add_10[0][0]']
block13_sepconv1 (SeparableCon	(None, 19, 19, 728) 536536	['block13_sepconv1_act[0][0]']

v2D)

block13_sepconv1_bn (BatchNorm alization)	(None, 19, 19, 728) 2912	[ 'block13_sepconv1[0][0]' ]
block13_sepconv2_act (Activati on)	(None, 19, 19, 728) 0	[ 'block13_sepconv1_bn[0][0]' ]
block13_sepconv2 (SeparableCon v2D)	(None, 19, 19, 1024 752024 )	[ 'block13_sepconv2_act[0][0]' ]
block13_sepconv2_bn (BatchNorm alization)	(None, 19, 19, 1024 4096 )	[ 'block13_sepconv2[0][0]' ]
conv2d_3 (Conv2D)	(None, 10, 10, 1024 745472 )	[ 'add_10[0][0]' ]
block13_pool (MaxPooling2D)	(None, 10, 10, 1024 0 )	[ 'block13_sepconv2_bn[0][0]' ]
batch_normalization_3 (BatchNo rmalization)	(None, 10, 10, 1024 4096 )	[ 'conv2d_3[0][0]' ]
add_11 (Add)	(None, 10, 10, 1024 0 )	[ 'block13_pool[0][0]', 'batch_normalization_3[0][0]' ]
block14_sepconv1 (SeparableCon v2D)	(None, 10, 10, 1536 1582080 )	[ 'add_11[0][0]' ]
block14_sepconv1_bn (BatchNorm alization)	(None, 10, 10, 1536 6144 )	[ 'block14_sepconv1[0][0]' ]
block14_sepconv1_act (Activati on)	(None, 10, 10, 1536 0 )	[ 'block14_sepconv1_bn[0][0]' ]
block14_sepconv2 (SeparableCon v2D)	(None, 10, 10, 2048 3159552 )	[ 'block14_sepconv1_act[0][0]' ]
block14_sepconv2_bn (BatchNorm alization)	(None, 10, 10, 2048 8192 )	[ 'block14_sepconv2[0][0]' ]
block14_sepconv2_act (Activati on)	(None, 10, 10, 2048 0 )	[ 'block14_sepconv2_bn[0][0]' ]

```
avg_pool (GlobalAveragePooling (None, 2048)      0      [ 'block14_sepconv2_act[0][0]' ]
2D)
```

```
=====
Total params: 20,861,480
Trainable params: 20,806,952
Non-trainable params: 54,528
```

---

None

In [19]:

```
from keras.preprocessing.image import load_img
from keras.preprocessing.image import img_to_array
from keras.applications.xception import preprocess_input
```

In [21]:

```
def extract_features(directory):
    count=0
    features = dict()
    for name in.listdir(directory):
        filename = directory + '/' + name
        image = load_img(filename,target_size=(299,299))
        image = img_to_array(image)
        image = image.reshape(1,299,299,3)
        image = preprocess_input(image)
        feature = model.predict(image)
        image_id = name.split('.')[0]
        features[image_id] = feature
        count=count+1
        if(count%500==0):
            print(count)
    return features

# extract features from all images
directory = 'C:/Users/mahid/M.Tech Project/Data Set/Images/'
features = extract_features(directory)
print('Extracted Features: %d' % len(features))
```

500  
1000  
1500  
2000  
2500  
3000

```
3500  
4000  
4500  
5000  
5500  
6000  
6500  
7000  
7500  
8000  
Extracted Features: 8091
```

```
In [22]: dump(features, open('C:/Users/mahid/M.Tech Project/Xception_Model/features.pkl', 'wb'))
```

```
import string  
def map_captions(file):  
    captions_map={}  
    for i in file.split('\n'):  
        img_data=i.split()  
        if len(i)<2:  
            continue  
        image_id=img_data[0]  
        image_caption=img_data[1:]  
        image_id=image_id.split('.')[0]  
        image_caption=' '.join(image_caption)  
        if image_id not in captions_map:  
            captions_map[image_id]=[]  
        captions_map[image_id].append(image_caption)  
    return captions_map
```

```
In [2]: def clean_raw_descriptions(descriptions):  
    mytable=str.maketrans(' ', ' ', string.punctuation)  
    for image_id,image_caption in descriptions.items():  
        for i in range(len(image_caption)):  
            caption=image_caption[i]  
            caption=caption.split()  
            caption=[word.lower() for word in caption]  
            caption=[w.translate(mytable) for w in caption]  
            caption=[word for word in caption if len(word)>1]  
            caption=[word for word in caption if word.isalpha()]  
            image_caption[i]=' '.join(caption)
```

In [3]:

```

def vocabulary_extraction(image_captions):
    list_words=set()
    for key in image_captions.keys():
        [list_words.update(d.split()) for d in image_captions[key]]
    return(list_words)

def count_words(captions_dict):
    word_count={}
    for image,captions in captions_dict.items():
        for caption in captions:
            for word in caption.split():
                if word not in word_count:
                    word_count[word] = 0
                else:
                    word_count[word] += 1
    return(word_count)

# save descriptions to file, one per line
def save_descriptions(descriptions,filename):
    lines=[]
    for image_id,image_caption in descriptions.items():
        for caption in image_caption:
            lines.append(image_id + ' ' +caption)
    data='\n'.join(lines)
    file=open(filename, 'w')
    file.write(data)
    file.close()

```

In [4]:

```

file=open('C:/Users/mahid/M.Tech Project/Data Set/Flickr8k_text/Flickr8k.token.txt', 'r')
text=file.read()
file.close()
desc =mapCaptions(text)
print('Total images mapped with captions : %d' % len(desc))
clean_raw_descriptions(desc)
vocabulary=vocabulary_extraction(desc)
word_count=count_words(desc)
print('Vocabulary Size after cleaning : %d' % len(vocabulary))
save_descriptions(desc,'C:/Users/mahid/M.Tech Project/Xception_Model/descriptions.txt')

```

Total images mapped with captions : 8091

Vocabulary Size after cleaning : 8762

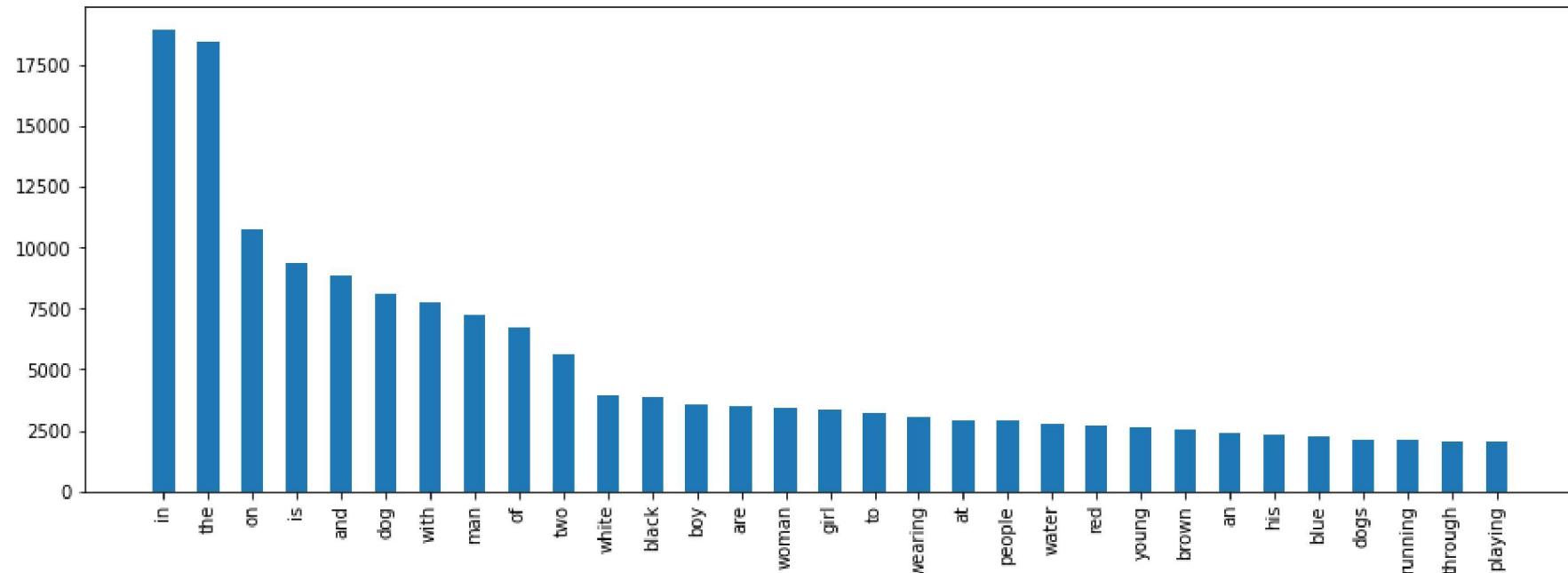
In [6]:

```
sorted_list= sorted(word_count.items(), key=lambda x:x[1],reverse=True)
top_30_items={}
count=30
for word,counter in sorted_list:
    top_30_items[word]=counter
    if(count==0):
        break
    count=count-1
for word in top_30_items:
    print(word,top_30_items[word])
```

in 18969  
the 18415  
on 10743  
is 9342  
and 8851  
dog 8130  
with 7763  
man 7265  
of 6712  
two 5638  
white 3939  
black 3831  
boy 3580  
are 3504  
woman 3402  
girl 3327  
to 3171  
wearing 3061  
at 2914  
people 2886  
water 2782  
red 2671  
young 2629  
brown 2560  
an 2431  
his 2356  
blue 2267  
dogs 2124  
running 2072  
through 2031  
playing 2006

In [7]:

```
import matplotlib.pyplot as plt
keys=top_30_items.keys()
values=top_30_items.values()
plt.figure(figsize=(15,5))
plt.bar(keys,values,width=0.5)
plt.xticks(range(len(values)),keys,rotation=90)
plt.show()
```



In [9]:

```
def load_set(filename):
    dataset=[]
    file=open(filename, 'r')
    text=file.read()
    file.close()
    for line in text.split('\n'):
        identifier = line.split('.')[0]
        dataset.append(identifier)
    return(dataset)
```

In [10]:

```
filename = 'C:/Users/mahid/M.Tech Project/Data Set/Flickr8k_text/Flickr_8k.trainImages.txt'
```

```
train_dataset=load_set(filename)
print('Training Data Set Length is : %d' % len(train_dataset))
```

Training Data Set Length is : 6000

```
In [11]: def add_identifiers(filename,dataset):
    file=open(filename,'r')
    text=file.read()
    file.close()
    captions={}
    for line in text.split('\n'):
        image_data=line.split()
        image_id,image_caption=image_data[0],image_data[1:]
        if image_id in dataset:
            if image_id not in captions:
                captions[image_id]=[]
            modified= 'startseq '+' '.join(image_caption)+' endseq'
            captions[image_id].append(modified)
    return captions
```

```
In [12]: trainCaptions=add_identifiers('C:/Users/mahid/M.Tech Project/Xception_Model/descriptions.txt',train_dataset)
print('Total image Captions read : train %d' % len(trainCaptions))
```

Total image Captions read : train 5999

```
In [13]: for i in range(10):
    print(trainCaptions[train_dataset[i]])
```

```
['startseq black dog is running after white dog in the snow endseq', 'startseq black dog chasing brown dog through snow endseq',
'startseq two dogs chase each other across the snowy ground endseq', 'startseq two dogs play together in the snow endseq', 'starts
eq two dogs running through low lying body of water endseq']
['startseq little baby plays croquet endseq', 'startseq little girl plays croquet next to truck endseq', 'startseq the child is pl
aying croquette by the truck endseq', 'startseq the kid is in front of car with put and ball endseq', 'startseq the little boy is
playing with croquet hammer and ball beside the car endseq']
['startseq brown dog in the snow has something hot pink in its mouth endseq', 'startseq brown dog in the snow holding pink hat end
seq', 'startseq brown dog is holding pink shirt in the snow endseq', 'startseq dog is carrying something pink in its mouth while w
alking through the snow endseq', 'startseq dog with something pink in its mouth is looking forward endseq']
['startseq brown dog is running along beach endseq', 'startseq brown dog wearing black collar running across the beach endseq', 's
tartseq dog walks on the sand near the water endseq', 'startseq brown dog running on the beach endseq', 'startseq the large brown
dog is running on the beach by the ocean endseq']
['startseq black and white dog with red frisbee standing on sandy beach endseq', 'startseq dog drops red disc on beach endseq', 's
```

tartseq dog with red frisbee flying in the air endseq', 'startseq dog catching red frisbee endseq', 'startseq the black dog is dropping red disc on beach endseq']  
['startseq cyclist wearing red helmet is riding on the pavement endseq', 'startseq girl is riding bike on the street while wearing red helmet endseq', 'startseq person on bike wearing red helmet riding down street endseq', 'startseq woman wears red helmet and blue shirt as she goes for bike ride in the shade endseq', 'startseq person in blue shirt and red helmet riding bike down the road endseq']  
['startseq man dressed in purple shirt and red bandanna smiles at the people watching him endseq', 'startseq man on the street wearing leather chaps and chainmail codpiece endseq', 'startseq man wearing purple shirt and black leather chaps poses for the camera endseq', 'startseq man dressed in leather chaps and purple shirt stands in front of onlookers endseq', 'startseq there is man in purple shirt leather chaps and red bandanna standing near other men endseq']  
['startseq boy wearing red tshirt is running through woodland endseq', 'startseq child runs near some trees endseq', 'startseq young boy is dancing around endseq', 'startseq young boy with red short sleeved shirt and jeans runs by some trees endseq', 'startseq the little boy in the red shirt stops to smile for the camera endseq']  
['startseq girl in white dress endseq', 'startseq little girl in white is looking back at the camera while carrying water grenade endseq', 'startseq smiling young girl in braids is playing ball endseq', 'startseq young girl wearing white looks at the camera as she plays endseq', 'startseq the girl is holding green ball endseq']  
['startseq skier in yellow jacket is airborne above the mountains endseq', 'startseq skier jumps high in the air with view of the mountains endseq', 'startseq skiing man in fluorescent jacket jumps very high and it looks as though he is flying endseq', 'startseq someone is high in the air doing ski jump endseq', 'startseq the skier in the green jacket and white pants appears to almost fly into the sky endseq']

In [14]:

```
from pickle import load
def load_features(filename, dataset):
    features={}
    image_features=load(open(filename,'rb'))
    for i in dataset:
        if i in image_features:
            features[i]=image_features[i]
    return features
```

In [15]:

```
train_image_vector=load_features('C:/Users/mahid/M.Tech Project/Xception_Model/features.pkl',train_dataset)
print('Total Features for images obtained : =%d' % len(train_image_vector))
```

Total Features for images obtained : =5999

In [16]:

```
c=0
for i,j in train_captions.items():
    print(str(i))
    print(str(train_captions[i]))
    c=c+1
    if(c==10):
        break
```

1000268201\_693b08cb0e

[ 'startseq child in pink dress is climbing up set of stairs in an entry way endseq', 'startseq girl going into wooden building end seq', 'startseq little girl climbing into wooden playhouse endseq', 'startseq little girl climbing the stairs to her playhouse end seq', 'startseq little girl in pink dress going into wooden cabin endseq' ]

1001773457\_577c3a7d70

[ 'startseq black dog and spotted dog are fighting endseq', 'startseq black dog and tricolored dog playing with each other on the r oad endseq', 'startseq black dog and white dog with brown spots are staring at each other in the street endseq', 'startseq two dog s of different breeds looking at each other on the road endseq', 'startseq two dogs on pavement moving toward each other endseq' ]

1002674143\_1b742ab4b8

[ 'startseq little girl covered in paint sits in front of painted rainbow with her hands in bowl endseq', 'startseq little girl is sitting in front of large painted rainbow endseq', 'startseq small girl in the grass plays with fingerpaints in front of white can vas with rainbow on it endseq', 'startseq there is girl with pigtails sitting in front of rainbow painting endseq', 'startseq youn g girl with pigtails painting outside in the grass endseq' ]

1003163366\_44323f5815

[ 'startseq man lays on bench while his dog sits by him endseq', 'startseq man lays on the bench to which white dog is also tied en dseq', 'startseq man sleeping on bench outside with white and black dog sitting next to him endseq', 'startseq shirtless man lies on park bench with his dog endseq', 'startseq man laying on bench holding leash of dog sitting on ground endseq' ]

1007129816\_e794419615

[ 'startseq man in an orange hat starring at something endseq', 'startseq man wears an orange hat and glasses endseq', 'startseq ma n with gauges and glasses is wearing blitz hat endseq', 'startseq man with glasses is wearing beer can crocheted hat endseq', 'sta rtseq the man with pierced ears is wearing glasses and an orange hat endseq' ]

1007320043\_627395c3d8

[ 'startseq child playing on rope net endseq', 'startseq little girl climbing on red roping endseq', 'startseq little girl in pink climbs rope bridge at the park endseq', 'startseq small child grips onto the red ropes at the playground endseq', 'startseq the sm all child climbs on red ropes on playground endseq' ]

1009434119\_febe49276a

[ 'startseq black and white dog is running in grassy garden surrounded by white fence endseq', 'startseq black and white dog is run ning through the grass endseq', 'startseq boston terrier is running in the grass endseq', 'startseq boston terrier is running on l ush green grass in front of white fence endseq', 'startseq dog runs on the green grass near wooden fence endseq' ]

1012212859\_01547e3f17

[ 'startseq dog shakes its head near the shore red ball next to it endseq', 'startseq white dog shakes on the edge of beach with an orange ball endseq', 'startseq dog with orange ball at feet stands on shore shaking off water endseq', 'startseq white dog playing with red ball on the shore near the water endseq', 'startseq white dog with brown ears standing near water with head turned to one side endseq' ]

1015118661\_980735411b

[ 'startseq boy smiles in front of stony wall in city endseq', 'startseq little boy is standing on the street while man in overalls is working on stone wall endseq', 'startseq young boy runs across the street endseq', 'startseq young child is walking on stone pav ed street with metal pole and man behind him endseq', 'startseq smiling boy in white shirt and blue jeans in front of rock wall wi th man in overalls behind him endseq' ]

1015584366\_dfcec3c85a

[ 'startseq black dog leaps over log endseq', 'startseq grey dog is leaping over fallen tree endseq', 'startseq large black dog lea ps fallen log endseq', 'startseq mottled black and grey dog in blue collar jumping over fallen tree endseq', 'startseq the black d og jumped the tree stump endseq' ]

```
In [17]: def split_captions(captions):
    caption=[]
    for i in captions.keys():
        [caption.append(j) for j in captions[i]]
    return caption
```

```
In [18]: from keras.preprocessing.text import Tokenizer
```

```
In [19]: def token_convert(captions):
    lines=split_captions(captions)
    tokenizer=Tokenizer()
    tokenizer.fit_on_texts(lines)
    return tokenizer
```

```
In [20]: tokenizer=token_convert(train_captions)
vocabulary_size=len(tokenizer.word_index)+1
print('Vocabulary Size: %d' % vocabulary_size)
```

Vocabulary Size: 7578

```
In [21]: lines=split_captions(train_captions)
max_length=max(len(i.split()) for i in lines)
print('Maximum caption length is:%d' %max_length)
```

Maximum caption length is:34

```
In [22]: for i in range(20):
    print(lines[i])
```

startseq child in pink dress is climbing up set of stairs in an entry way endseq  
startseq girl going into wooden building endseq  
startseq little girl climbing into wooden playhouse endseq  
startseq little girl climbing the stairs to her playhouse endseq  
startseq little girl in pink dress going into wooden cabin endseq  
startseq black dog and spotted dog are fighting endseq  
startseq black dog and tricolored dog playing with each other on the road endseq  
startseq black dog and white dog with brown spots are staring at each other in the street endseq  
startseq two dogs of different breeds looking at each other on the road endseq  
startseq two dogs on pavement moving toward each other endseq

```
startseq little girl covered in paint sits in front of painted rainbow with her hands in bowl endseq
startseq little girl is sitting in front of large painted rainbow endseq
startseq small girl in the grass plays with fingerpaints in front of white canvas with rainbow on it endseq
startseq there is girl with pigtails sitting in front of rainbow painting endseq
startseq young girl with pigtails painting outside in the grass endseq
startseq man lays on bench while his dog sits by him endseq
startseq man lays on the bench to which white dog is also tied endseq
startseq man sleeping on bench outside with white and black dog sitting next to him endseq
startseq shirtless man lies on park bench with his dog endseq
startseq man laying on bench holding leash of dog sitting on ground endseq
```

In [23]:

```
from numpy import array
```

In [24]:

```
def output_sequences(tokenizer,max_length,caption_list,image):
    inp1=[]
    inp2=[]
    out=[]
    for caption in caption_list:
        sequence=tokenizer.texts_to_sequences([caption])[0]
        for i in range(1,len(sequence)):
            input_sequence,output_sequence=sequence[:i],sequence[i]
            input_sequence=pad_sequences([input_sequence],maxlen=max_length)[0]
            output_sequence=to_categorical([output_sequence],num_classes=vocabulary_size)[0]
            inp1.append(image)
            inp2.append(input_sequence)
            out.append(output_sequence)
    return array(inp1),array(inp2),array(out)
```

In [25]:

```
def data_generator(tokenizer,max_length,captions,images):
    while 1:
        for key,desc in captions.items():
            image=images[key][0]
            input_img,input_seq,output_word=output_sequences(tokenizer,max_length,desc,image)
            yield [[input_img, input_seq], output_word]
```

In [26]:

```
import tensorflow
from keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.utils import plot_model
from keras.models import Model,Sequential
```

```
from keras.layers import Input
from keras.layers import Dense
from keras.layers import LSTM
from keras.layers import Embedding
from keras.layers import Dropout
from keras.layers.merge import add
from keras.callbacks import ModelCheckpoint
from keras.layers import Flatten, Convolution2D, TimeDistributed, Bidirectional, Activation, RepeatVector, Concatenate
from tensorflow.keras.utils import plot_model
from numpy import argmax
from keras.preprocessing.sequence import pad_sequences
from keras.models import load_model
from nltk.translate.bleu_score import corpus_bleu
```

In [27]:

```
def create_model(vocabulary_size,max_length):
    inputs1=Input(shape=(2048,))
    img_model_1=Dropout(0.5)(inputs1)
    img_model_2=Dense(128, activation='relu')(inputs1)
    # sequence model
    inputs2=Input(shape=(max_length,))
    sequence_model_1=Embedding(vocabulary_size,128,mask_zero=True)(inputs2)
    sequence_model_2=Dropout(0.5)(sequence_model_1)
    sequence_model_3=LSTM(128)(sequence_model_2)
    # decoder model
    decoder1=add([img_model_2,sequence_model_3])
    decoder2=Dense(128,activation='relu')(decoder1)
    outputs=Dense(vocabulary_size,activation='softmax')(decoder2)
    # tie it together [image, seq] [word]
    model=Model(inputs=[inputs1,inputs2],outputs=outputs)
    model.compile(loss='categorical_crossentropy', optimizer='adam')
    # summarize model
    print(model.summary())
    return model
```

In [28]:

```
print(vocabulary_size)
print(max_length)
```

7578

34

In [50]:

```
model=create_model(vocabulary_size, max_length)
```

```

epochs=20
steps=len(train_captions)
for i in range(epochs):
    generator=data_generator(tokenizer,max_length,train_captions,train_image_vector)
    model.fit(generator,epochs=1,steps_per_epoch=steps)
    model.save('C:/Users/mahid/M.Tech Project/Xception_Model/caption_model_'+str(i)+'.h5')

```

Model: "model\_2"

Layer (type)	Output Shape	Param #	Connected to
<hr/>			
input_5 (InputLayer)	[(None, 34)]	0	[]
embedding_1 (Embedding)	(None, 34, 128)	969984	['input_5[0][0]']
input_4 (InputLayer)	[(None, 2048)]	0	[]
dropout_3 (Dropout)	(None, 34, 128)	0	['embedding_1[0][0]']
dense_3 (Dense)	(None, 128)	262272	['input_4[0][0]']
lstm_1 (LSTM)	(None, 128)	131584	['dropout_3[0][0]']
add_13 (Add)	(None, 128)	0	['dense_3[0][0]', 'lstm_1[0][0]']
dense_4 (Dense)	(None, 128)	16512	['add_13[0][0]']
dense_5 (Dense)	(None, 7578)	977562	['dense_4[0][0]']
<hr/>			

Total params: 2,357,914

Trainable params: 2,357,914

Non-trainable params: 0

---

None

```

5999/5999 [=====] - 921s 151ms/step - loss: 4.6399
5999/5999 [=====] - 576s 96ms/step - loss: 3.8058
5999/5999 [=====] - 409s 68ms/step - loss: 3.5099
5999/5999 [=====] - 390s 65ms/step - loss: 3.3330
5999/5999 [=====] - 390s 65ms/step - loss: 3.2110
5999/5999 [=====] - 393s 66ms/step - loss: 3.1205
5999/5999 [=====] - 391s 65ms/step - loss: 3.0528
5999/5999 [=====] - 392s 65ms/step - loss: 2.9936

```

```
5999/5999 [=====] - 386s 64ms/step - loss: 2.9512
5999/5999 [=====] - 386s 64ms/step - loss: 2.9143
5999/5999 [=====] - 395s 66ms/step - loss: 2.8766
5999/5999 [=====] - 401s 67ms/step - loss: 2.8498
5999/5999 [=====] - 404s 67ms/step - loss: 2.8246
5999/5999 [=====] - 404s 67ms/step - loss: 2.8030
5999/5999 [=====] - 407s 68ms/step - loss: 2.7811
5999/5999 [=====] - 398s 66ms/step - loss: 2.7657
5999/5999 [=====] - 401s 67ms/step - loss: 2.7513
5999/5999 [=====] - 415s 69ms/step - loss: 2.7366
5999/5999 [=====] - 414s 69ms/step - loss: 2.7239
5999/5999 [=====] - 415s 69ms/step - loss: 2.7136
```

In [29]:

```
def word_mapping(number,tokenizer):
    for word,int_id in tokenizer.word_index.items():
        if int_id==number:
            return word
    return None
```

In [30]:

```
def caption_generate(model,tokenizer,image,max_length):
    caption='startseq'
    for i in range(max_length):
        sequence=tokenizer.texts_to_sequences([caption])[0]
        sequence=pad_sequences([sequence],maxlen=max_length)
        word_id=argmax(model.predict([image,sequence],verbose=0))
        word=word_mapping(word_id,tokenizer)
        if word is None:
            break
        caption += ' ' + word
        if word=='endseq':
            break
    return caption
```

In [31]:

```
def evaluate_model(model,descriptions,photos,tokenizer,max_length):
    actual=[]
    predicted=[]
    for key,desc_list in descriptions.items():
        pred=caption_generate(model,tokenizer,photos[key],max_length)
        references=[i.split() for i in desc_list]
        actual.append(references)
        predicted.append(pred.split())
```

```

print('BLEU-1 Score: %f' % corpus_bleu(actual,predicted,weights=(1.0, 0, 0, 0)))
print('BLEU-2 Score: %f' % corpus_bleu(actual,predicted,weights=(0.5, 0.5, 0, 0)))
print('BLEU-3 Score: %f' % corpus_bleu(actual,predicted,weights=(0.33, 0.33, 0.33, 0)))
print('BLEU-4 Score: %f' % corpus_bleu(actual,predicted,weights=(0.25, 0.25, 0.25, 0.25)))

```

In [32]:

```

filename = 'C:/Users/mahid/M.Tech Project/Data Set/Flickr8k_text/Flickr_8k.testImages.txt'
test_dataset = load_set(filename)
print('Dataset: %d' % len(test_dataset))
test_descriptions=add_identifiers('C:/Users/mahid/M.Tech Project/Xception_Model/descriptions.txt',test_dataset)
print('Captions: test=%d' % len(test_descriptions))
test_features=load_features('C:/Users/mahid/M.Tech Project/Xception_Model/features.pkl',test_dataset)
print('Photos: test=%d' % len(test_features))
for i in range(20):
    print("Model "+str(i)+" : ")
    filename='C:/Users/mahid/M.Tech Project/Xception_Model/caption_model_'+str(i)+'.h5'
    model=load_model(filename)
    evaluate_model(model,test_descriptions,test_features,tokenizer,max_length)

```

Dataset: 2001  
 Captions: test=2000  
 Photos: test=2000  
 Model 0 :  
 BLEU-1 Score: 0.536114  
 BLEU-2 Score: 0.297098  
 BLEU-3 Score: 0.171659  
 BLEU-4 Score: 0.091279  
 Model 1 :  
 BLEU-1 Score: 0.553432  
 BLEU-2 Score: 0.308317  
 BLEU-3 Score: 0.183296  
 BLEU-4 Score: 0.102154  
 Model 2 :  
 BLEU-1 Score: 0.562758  
 BLEU-2 Score: 0.313774  
 BLEU-3 Score: 0.186815  
 BLEU-4 Score: 0.103409  
 Model 3 :  
 BLEU-1 Score: 0.544538  
 BLEU-2 Score: 0.302205  
 BLEU-3 Score: 0.180268  
 BLEU-4 Score: 0.100878  
 Model 4 :  
 BLEU-1 Score: 0.551982

BLEU-2 Score: 0.307602

BLEU-3 Score: 0.182956

BLEU-4 Score: 0.101224

Model 5 :

BLEU-1 Score: 0.551822

BLEU-2 Score: 0.303435

BLEU-3 Score: 0.180403

BLEU-4 Score: 0.101375

Model 6 :

BLEU-1 Score: 0.553160

BLEU-2 Score: 0.307304

BLEU-3 Score: 0.182146

BLEU-4 Score: 0.102504

Model 7 :

BLEU-1 Score: 0.543118

BLEU-2 Score: 0.303725

BLEU-3 Score: 0.183449

BLEU-4 Score: 0.104024

Model 8 :

BLEU-1 Score: 0.550664

BLEU-2 Score: 0.309113

BLEU-3 Score: 0.184504

BLEU-4 Score: 0.102474

Model 9 :

BLEU-1 Score: 0.545490

BLEU-2 Score: 0.300911

BLEU-3 Score: 0.178600

BLEU-4 Score: 0.099388

Model 10 :

BLEU-1 Score: 0.545730

BLEU-2 Score: 0.302502

BLEU-3 Score: 0.181417

BLEU-4 Score: 0.103378

Model 11 :

BLEU-1 Score: 0.545367

BLEU-2 Score: 0.303911

BLEU-3 Score: 0.181813

BLEU-4 Score: 0.100964

Model 12 :

BLEU-1 Score: 0.540151

BLEU-2 Score: 0.299608

BLEU-3 Score: 0.180269

BLEU-4 Score: 0.100953

Model 13 :

BLEU-1 Score: 0.541113  
BLEU-2 Score: 0.300457  
BLEU-3 Score: 0.180851  
BLEU-4 Score: 0.103041  
Model 14 :  
BLEU-1 Score: 0.536692  
BLEU-2 Score: 0.298734  
BLEU-3 Score: 0.179256  
BLEU-4 Score: 0.100713  
Model 15 :  
BLEU-1 Score: 0.545745  
BLEU-2 Score: 0.307808  
BLEU-3 Score: 0.187383  
BLEU-4 Score: 0.107373  
Model 16 :  
BLEU-1 Score: 0.537386  
BLEU-2 Score: 0.298972  
BLEU-3 Score: 0.179304  
BLEU-4 Score: 0.102725  
Model 17 :  
BLEU-1 Score: 0.540022  
BLEU-2 Score: 0.304328  
BLEU-3 Score: 0.185856  
BLEU-4 Score: 0.107739  
Model 18 :  
BLEU-1 Score: 0.531461  
BLEU-2 Score: 0.296724  
BLEU-3 Score: 0.180293  
BLEU-4 Score: 0.103208  
Model 19 :  
BLEU-1 Score: 0.547769  
BLEU-2 Score: 0.308132  
BLEU-3 Score: 0.189317  
BLEU-4 Score: 0.109944