

```

from tensorflow.keras.utils import load_img
from tensorflow.keras.utils import img_to_array
from keras.applications.vgg16 import preprocess_input
from keras.applications.vgg16 import decode_predictions
from keras.applications.vgg16 import VGG16


```

Pre-Trained Model as Classifier

```

model = VGG16()
image = load_img('mug.png', target_size=(224, 224))
image = img_to_array(image)
image = image.reshape((1, image.shape[0], image.shape[1], image.shape[2]))
image = preprocess_input(image)
yhat = model.predict(image)
label = decode_predictions(yhat)
label = label[0][0]
print('%s (%.2f%%)' % (label[1], label[2]*100))

```

 Downloading data from <https://storage.googleapis.com/tensorflow/keras-applications/vgg16/553467096/553467096> [=====] - 4s 0us/step
 1/1 [=====] - 1s 1s/step
 Downloading data from https://storage.googleapis.com/download.tensorflow.org/data/image_net_vgg16.zip
 35363/35363 [=====] - 0s 0us/step
 coffee_mug (78.16%)



Pre-Trained Model as Feature Extractor in PreProcessor

```

from keras.models import Model
from pickle import dump

image2 = load_img('dog.jpg', target_size=(224, 224))
image2 = img_to_array(image2)
image2 = image2.reshape((1, image2.shape[0], image2.shape[1], image2.shape[2]))
image2 = preprocess_input(image2)
model2 = VGG16()
model2 = Model(inputs=model2.inputs, outputs=model2.layers[-2].output)
features = model2.predict(image2)
print(features.shape)
dump(features, open('dog.pkl', 'wb'))

1/1 [=====] - 1s 786ms/step
(1, 4096)

```

Pre-trained Model as Feature Extractor in Model

```

print("INFO loading network....")
model_vgg = VGG16(weights="imagenet", include_top=False, input_shape=train_x.shape[1:])
model_vgg.summary()

```

```
--
NameError                                Traceback (most recent call
last)
<ipython-input-8-222408eb6d91> in <module>
      1 print("INFO loading network....")
----> 2 model_vgg = VGG16(weights="imagenet", include_top=False,
input_shape=train_x.shape[1:])
      3 model_vgg.summary()
```

● ×