dl-ex-5

May 1, 2023

[ ]:

```
from tensorflow.keras.applications.vgg16 import VGG16
from tensorflow.keras.layers import Dense, Flatten
from tensorflow.keras.models import Model
from tensorflow.keras.datasets import cifar10
from tensorflow.keras.utils import to_categorical
```

[2]:

```
# Load the CIFAR-10 dataset
(X_train, y_train), (X_test, y_test) = cifar10.load_data()
```

```
Downloading data from https://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz
170498071/170498071 [==============================] - 3s 0us/step
```

[3]:

```
# Normalize the pixel values
X_train = X_train / 255.0
X_test = X_test / 255.0
```

[4]:

```
# Convert the labels to one-hot encoded vectors
y_train = to_categorical(y_train, num_classes=10)
y_test = to_categorical(y_test, num_classes=10)
```

[5]:

```
# Load the pre-trained VGG-16 model
vgg16 = VGG16(weights='imagenet', include_top=False, input_shape=(32, 32, 3))
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-
applications/vgg16/vgg16_weights_tf_dim_ordering_tf_kernels_notop.h5
58889256/58889256 [==============================] - 0s 0us/step
```

[6]:

```
# Freeze the weights of the pre-trained layers
for layer in vgg16.layers:
    layer.trainable = False
```

[7]:

```
# Add new layers for classification
x = Flatten()(vgg16.output)
x = Dense(units=256, activation='relu')(x)
x = Dense(units=10, activation='softmax')(x)
```

[8]:

```
# Create a new model that combines the pre-trained VGG-16 model with the new classification layers
model = Model(inputs=vgg16.input, outputs=x)
```

[9]:

```
# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

[10]:

```
# Train the model
model.fit(X_train, y_train, epochs=10, batch_size=32, validation_data=(X_test, y_test))
```

```
Epoch 1/10
1563/1563 [==============================] - 772s 493ms/step - loss: 1.3405 -
accuracy: 0.5327 - val_loss: 1.2420 - val_accuracy: 0.5626
Epoch 2/10
1563/1563 [==============================] - 766s 490ms/step - loss: 1.1675 -
accuracy: 0.5912 - val_loss: 1.1956 - val_accuracy: 0.5815
Epoch 3/10
1563/1563 [==============================] - 765s 489ms/step - loss: 1.1027 -
accuracy: 0.6124 - val_loss: 1.1382 - val_accuracy: 0.5961
Epoch 4/10
1563/1563 [==============================] - 766s 490ms/step - loss: 1.0519 -
accuracy: 0.6333 - val_loss: 1.1416 - val_accuracy: 0.6017
Epoch 5/10
1563/1563 [==============================] - 764s 489ms/step - loss: 1.0087 -
accuracy: 0.6465 - val_loss: 1.1113 - val_accuracy: 0.6127
Epoch 6/10
1563/1563 [==============================] - 763s 488ms/step - loss: 0.9701 -
accuracy: 0.6597 - val_loss: 1.1203 - val_accuracy: 0.6080
Epoch 7/10
1563/1563 [==============================] - 766s 490ms/step - loss: 0.9373 -
accuracy: 0.6703 - val_loss: 1.1163 - val_accuracy: 0.6121
Epoch 8/10
1563/1563 [==============================] - 766s 490ms/step - loss: 0.9037 -
accuracy: 0.6813 - val_loss: 1.1054 - val_accuracy: 0.6140
Epoch 9/10
1563/1563 [==============================] - 764s 489ms/step - loss: 0.8730 -
accuracy: 0.6916 - val_loss: 1.1292 - val_accuracy: 0.6104
Epoch 10/10
1563/1563 [==============================] - 765s 489ms/step - loss: 0.8410 -
accuracy: 0.7051 - val_loss: 1.1214 - val_accuracy: 0.6162
```

[10]:

```
<keras.callbacks.History at 0x7f4ab179fd60>
```

[11]:

```
from tensorflow.keras.applications.vgg19 import VGG19
from tensorflow.keras.layers import Dense, Flatten
from tensorflow.keras.models import Model
from tensorflow.keras.datasets import cifar10
from tensorflow.keras.utils import to_categorical
```

[12]:

```
# Load the CIFAR-10 dataset
(X_train, y_train), (X_test, y_test) = cifar10.load_data()
```

[13]:

```
# Normalize the pixel values
X_train = X_train / 255.0
X_test = X_test / 255.0
```

[14]:

```
# Convert the labels to one-hot encoded vectors
y_train = to_categorical(y_train, num_classes=10)
y_test = to_categorical(y_test, num_classes=10)
```

[15]:

```
# Load the pre-trained VGG-19 model
vgg19 = VGG19(weights='imagenet', include_top=False, input_shape=(32, 32, 3))
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-
applications/vgg19/vgg19_weights_tf_dim_ordering_tf_kernels_notop.h5
80134624/80134624 [==============================] - 1s 0us/step
```

[16]:

```
# Freeze the weights of the pre-trained layers
for layer in vgg19.layers:
    layer.trainable = False
```

[17]:

```
# Add new layers for classification
x = Flatten()(vgg19.output)
x = Dense(units=256, activation='relu')(x)
x = Dense(units=10, activation='softmax')(x)
```

[18]:

```
# Create a new model that combines the pre-trained VGG-19 model with the new␣
 ↪classification layers
model = Model(inputs=vgg19.input, outputs=x)
```

[19]:

```
# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy',␣
 ↪metrics=['accuracy'])
```

[20]:

```
# Train the model
model.fit(X_train, y_train, epochs=10, batch_size=32, validation_data=(X_test,␣
 ↪y_test))
```

```
Epoch 1/10
1563/1563 [==============================] - 940s 601ms/step - loss: 1.3967 -
accuracy: 0.5070 - val_loss: 1.2662 - val_accuracy: 0.5582
Epoch 2/10
```

```
1563/1563 [==============================] - 935s 598ms/step - loss: 1.2170 -
accuracy: 0.5725 - val_loss: 1.2204 - val_accuracy: 0.5724
Epoch 3/10
1563/1563 [==============================] - 983s 629ms/step - loss: 1.1562 -
accuracy: 0.5945 - val_loss: 1.1994 - val_accuracy: 0.5796
Epoch 4/10
1563/1563 [==============================] - 984s 630ms/step - loss: 1.1091 -
accuracy: 0.6081 - val_loss: 1.1686 - val_accuracy: 0.5929
Epoch 5/10
1563/1563 [==============================] - 984s 630ms/step - loss: 1.0703 -
accuracy: 0.6229 - val_loss: 1.1450 - val_accuracy: 0.6033
Epoch 6/10
1563/1563 [==============================] - 984s 630ms/step - loss: 1.0366 -
accuracy: 0.6344 - val_loss: 1.1701 - val_accuracy: 0.5942
Epoch 7/10
1563/1563 [==============================] - 984s 629ms/step - loss: 1.0047 -
accuracy: 0.6461 - val_loss: 1.1477 - val_accuracy: 0.6057
Epoch 8/10
1563/1563 [==============================] - 932s 597ms/step - loss: 0.9770 -
accuracy: 0.6550 - val_loss: 1.1215 - val_accuracy: 0.6171
Epoch 9/10
1563/1563 [==============================] - 981s 628ms/step - loss: 0.9480 -
accuracy: 0.6659 - val_loss: 1.1326 - val_accuracy: 0.6099
Epoch 10/10
1563/1563 [==============================] - 937s 599ms/step - loss: 0.9221 -
accuracy: 0.6738 - val_loss: 1.1427 - val_accuracy: 0.6098
```

```
[20]: <keras.callbacks.History at 0x7f4aad716140>
```