Datensatz

February 12, 2021

Script for creating a combined dataset of CIFAR-10 and CIFAR-100
Attaching the category bottles from CIFAR-100 to CIFAR-10
Written by C.Joachim January 2021

[ ]:
```
import numpy as np
import tensorflow as tf
from tensorflow.keras import datasets
```

[ ]:
```
#import the CIFAR-10 and CIFAR-100 Datasets
(X_train10, y_train10), (X_test10, y_test10)     = tf.keras.datasets.cifar10.
 ↪load_data()
(X_train100, y_train100), (X_test100, y_test100) = tf.keras.datasets.cifar100.
 ↪load_data()
```

[ ]:
```
#reduce CIFAR-100 to the category 'bottles' and change label
idx        = (y_train100 == 9).reshape(X_train100.shape[0])
X_train100 = X_train100[idx]
y_train100 = y_train100[idx]
for i in range(len(y_train100)):
    y_train100[i]=10

idx       = (y_test100 == 9).reshape(X_test100.shape[0])
X_test100 = X_test100[idx]
y_test100 = y_test100[idx]
for i in range(len(y_test100)):
    y_test100[i]=10
```

[ ]:
```
#reduce CIFAR-10 to 500 training and 100 test images,
#to have the same number of images for every category
X_train10_red = [None]*5000
y_train10_red = [None]*5000

for i in range(10):
    idx = (y_train10 == i).reshape(X_train10.shape[0])
    x   = X_train10[idx]
    y   = y_train10[idx]
    X_train10_red[i*500:i*500+500] = x[0:500]
    y_train10_red[i*500:i*500+500] = y[0:500]
```

```
X_test10_red = [None]*1000
y_test10_red = [None]*1000

for i in range(10):
    idx = (y_test10 == i).reshape(X_test10.shape[0])
    x   = X_test10[idx]
    y   = y_test10[idx]
    X_test10_red[i*500:i*500+500] = x[0:500]
    y_test10_red[i*500:i*500+500] = y[0:500]
```

[ ]:
```
#attach the filtered CIFAR-100 to the reduced CIFAR-10
X_train  = np.concatenate((X_train10_red, X_train100))
y_train  = np.concatenate((y_train10_red, y_train100))

X_test   = np.concatenate((X_test10_red, X_test100))
y_test   = np.concatenate((y_test10_red, y_test100))
```

[ ]:
```
#shuffle the dataset
shuffler = np.random.permutation(len(X_train))
X_train  = X_train[shuffler]
y_train  = y_train[shuffler]

shuffler = np.random.permutation(len(X_test))
X_test   = X_test[shuffler]
y_test   = y_test[shuffler]
```

[ ]:
```
y_train  = tf.keras.utils.to_categorical(y_train)
y_test   = tf.keras.utils.to_categorical(y_test)
```

[ ]:
```
#store the data in single array to be stored
Data_CIFAR = (X_train, y_train, X_test, y_test)
```

[ ]:
```
#store dataset to be used in Training
%store Data_CIFAR
```

[ ]:
```

```