ge-reconstruction-with-autoencoder

April 4, 2024

[2]:
```
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.layers import Input, Dense
from tensorflow.keras.models import Model
from sklearn.datasets import fetch_lfw_people
from skimage.transform import resize

# Load the LFW dataset
lfw_people = fetch_lfw_people(min_faces_per_person=70, resize=0.4)

# Normalize pixel values to be between 0 and 1
X = lfw_people.images.astype('float32') / 255.0

# Resize images to 64x64
X_resized = np.array([resize(img, (64, 64), anti_aliasing=True) for img in X])

# Flatten the images for the autoencoder
X_flattened = X_resized.reshape((len(X_resized), np.prod(X_resized.shape[1:])))

# Define the autoencoder model
encoding_dim = 128  # Size of the encoded representations
input_img = Input(shape=(X_flattened.shape[1],))
encoded = Dense(encoding_dim, activation='relu')(input_img)
decoded = Dense(X_flattened.shape[1], activation='sigmoid')(encoded)

autoencoder = Model(input_img, decoded)

# Compile the autoencoder
autoencoder.compile(optimizer='adam', loss='binary_crossentropy')

# Train the autoencoder
autoencoder.fit(X_flattened, X_flattened, epochs=50, batch_size=256,␣
 ↪shuffle=True, validation_split=0.2)

# Create a separate encoder model
encoder = Model(input_img, encoded)
```

```
# Encode the images
encoded_imgs = encoder.predict(X_flattened)

# Decode the encoded images
decoded_imgs = autoencoder.predict(X_flattened)

# Display original and reconstructed images
n = 10  # Number of images to display
plt.figure(figsize=(20, 4))
for i in range(n):
    # Original images
    ax = plt.subplot(2, n, i + 1)
    plt.imshow(X_flattened[i].reshape(64, 64), cmap='gray')
    ax.get_xaxis().set_visible(False)
    ax.get_yaxis().set_visible(False)

    # Reconstructed images
    ax = plt.subplot(2, n, i + 1 + n)
    plt.imshow(decoded_imgs[i].reshape(64, 64), cmap='gray')
    ax.get_xaxis().set_visible(False)
    ax.get_yaxis().set_visible(False)

plt.show()
```

```
Epoch 1/50
5/5 [==============================] - 2s 150ms/step - loss: 0.6924 - val_loss:
0.6904
Epoch 2/50
5/5 [==============================] - 0s 88ms/step - loss: 0.6892 - val_loss:
0.6854
Epoch 3/50
5/5 [==============================] - 0s 92ms/step - loss: 0.6827 - val_loss:
0.6739
Epoch 4/50
5/5 [==============================] - 1s 95ms/step - loss: 0.6679 - val_loss:
0.6508
Epoch 5/50
5/5 [==============================] - 1s 104ms/step - loss: 0.6402 - val_loss:
0.6119
Epoch 6/50
5/5 [==============================] - 1s 126ms/step - loss: 0.5960 - val_loss:
0.5557
Epoch 7/50
5/5 [==============================] - 1s 143ms/step - loss: 0.5344 - val_loss:
0.4837
Epoch 8/50
5/5 [==============================] - 1s 154ms/step - loss: 0.4583 - val_loss:
```

```
0.4009
Epoch 9/50
5/5 [==============================] - 1s 137ms/step - loss: 0.3739 - val_loss:
0.3158
Epoch 10/50
5/5 [==============================] - 1s 115ms/step - loss: 0.2901 - val_loss:
0.2377
Epoch 11/50
5/5 [==============================] - 0s 90ms/step - loss: 0.2157 - val_loss:
0.1731
Epoch 12/50
5/5 [==============================] - 0s 92ms/step - loss: 0.1561 - val_loss:
0.1246
Epoch 13/50
5/5 [==============================] - 0s 82ms/step - loss: 0.1123 - val_loss:
0.0905
Epoch 14/50
5/5 [==============================] - 0s 87ms/step - loss: 0.0822 - val_loss:
0.0678
Epoch 15/50
5/5 [==============================] - 0s 88ms/step - loss: 0.0622 - val_loss:
0.0528
Epoch 16/50
5/5 [==============================] - 0s 94ms/step - loss: 0.0490 - val_loss:
0.0428
Epoch 17/50
5/5 [==============================] - 0s 87ms/step - loss: 0.0403 - val_loss:
0.0362
Epoch 18/50
5/5 [==============================] - 0s 83ms/step - loss: 0.0343 - val_loss:
0.0315
Epoch 19/50
5/5 [==============================] - 0s 90ms/step - loss: 0.0302 - val_loss:
0.0282
Epoch 20/50
5/5 [==============================] - 0s 90ms/step - loss: 0.0272 - val_loss:
0.0258
Epoch 21/50
5/5 [==============================] - 0s 93ms/step - loss: 0.0250 - val_loss:
0.0240
Epoch 22/50
5/5 [==============================] - 0s 85ms/step - loss: 0.0234 - val_loss:
0.0226
Epoch 23/50
5/5 [==============================] - 0s 91ms/step - loss: 0.0221 - val_loss:
0.0215
Epoch 24/50
5/5 [==============================] - 0s 86ms/step - loss: 0.0211 - val_loss:
```

```
0.0206
Epoch 25/50
5/5 [==============================] - 0s 82ms/step - loss: 0.0203 - val_loss:
0.0199
Epoch 26/50
5/5 [==============================] - 0s 92ms/step - loss: 0.0196 - val_loss:
0.0193
Epoch 27/50
5/5 [==============================] - 0s 85ms/step - loss: 0.0191 - val_loss:
0.0188
Epoch 28/50
5/5 [==============================] - 0s 88ms/step - loss: 0.0186 - val_loss:
0.0183
Epoch 29/50
5/5 [==============================] - 0s 79ms/step - loss: 0.0182 - val_loss:
0.0180
Epoch 30/50
5/5 [==============================] - 0s 88ms/step - loss: 0.0178 - val_loss:
0.0176
Epoch 31/50
5/5 [==============================] - 0s 88ms/step - loss: 0.0175 - val_loss:
0.0173
Epoch 32/50
5/5 [==============================] - 0s 81ms/step - loss: 0.0173 - val_loss:
0.0171
Epoch 33/50
5/5 [==============================] - 1s 125ms/step - loss: 0.0170 - val_loss:
0.0169
Epoch 34/50
5/5 [==============================] - 1s 126ms/step - loss: 0.0168 - val_loss:
0.0167
Epoch 35/50
5/5 [==============================] - 1s 119ms/step - loss: 0.0166 - val_loss:
0.0165
Epoch 36/50
5/5 [==============================] - 1s 126ms/step - loss: 0.0165 - val_loss:
0.0163
Epoch 37/50
5/5 [==============================] - 1s 124ms/step - loss: 0.0163 - val_loss:
0.0162
Epoch 38/50
5/5 [==============================] - 0s 84ms/step - loss: 0.0162 - val_loss:
0.0161
Epoch 39/50
5/5 [==============================] - 0s 88ms/step - loss: 0.0161 - val_loss:
0.0160
Epoch 40/50
5/5 [==============================] - 0s 88ms/step - loss: 0.0160 - val_loss:
```

```
0.0159
Epoch 41/50
5/5 [==============================] - 0s 89ms/step - loss: 0.0159 - val_loss:
0.0158
Epoch 42/50
5/5 [==============================] - 0s 84ms/step - loss: 0.0158 - val_loss:
0.0157
Epoch 43/50
5/5 [==============================] - 0s 82ms/step - loss: 0.0157 - val_loss:
0.0156
Epoch 44/50
5/5 [==============================] - 0s 87ms/step - loss: 0.0157 - val_loss:
0.0156
Epoch 45/50
5/5 [==============================] - 0s 93ms/step - loss: 0.0156 - val_loss:
0.0155
Epoch 46/50
5/5 [==============================] - 0s 86ms/step - loss: 0.0155 - val_loss:
0.0154
Epoch 47/50
5/5 [==============================] - 0s 87ms/step - loss: 0.0155 - val_loss:
0.0154
Epoch 48/50
5/5 [==============================] - 0s 87ms/step - loss: 0.0154 - val_loss:
0.0154
Epoch 49/50
5/5 [==============================] - 0s 87ms/step - loss: 0.0154 - val_loss:
0.0153
Epoch 50/50
5/5 [==============================] - 0s 80ms/step - loss: 0.0154 - val_loss:
0.0153
41/41 [==============================] - 0s 5ms/step
41/41 [==============================] - 0s 9ms/step
```