

In [1]: *# data augmentation test: rotate different degree (pay attention to adjustable filename etc.)*

```
import numpy as np
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow.python.keras.layers import Dense, GlobalAveragePooling2D
from tensorflow.python.keras.models import Model
from tensorflow.python.keras import layers, Sequential, losses, metrics

image_height = 48
image_width = 48
emotions_count = 8
emotion_labels = ['neutral', 'happiness', 'surprise', 'sadness',
                  'anger', 'disgust', 'fear', 'contempt']

# !!! change sample size
samples = 99109 # 2~99110
training_samples = 28317*3 # (Training)
validation_samples = 3541*3 # (PublicTest)
test_samples = 3535 # (PrivateTest)

# !!! change npy folder name
image_path = "./dataset_rran/images.npy"
emotion_multi_path = "./dataset_rran/emotions_multi.npy"
emotion_single_path = "./dataset_rran/emotions_single.npy"
images = np.load(image_path)
emotions_multi = np.load(emotion_multi_path)
emotions_single = np.load(emotion_single_path)

# !!! change s/m dataset
#emotions = emotions_single
emotions = emotions_multi

print(images.shape)
print(emotions_multi.shape)
print(emotions_single.shape)
```

```
(99109, 48, 48, 1)
(99109, 8)
(99109, 8)
```

In [2]:

```
cce = losses.CategoricalCrossentropy()
mse = losses.MeanSquaredError()

tf.config.run_functions_eagerly(True)
def model_acc(y_true, y_pred):
```

```

size = y_true.shape[0]
acc = 0
for i in range(size):
    true = y_true[i]
    pred = y_pred[i]
    index_max = tf.argmax(pred).numpy()
    if true[index_max].numpy() == tf.reduce_max(true).numpy():
        acc += 1
return acc/size

```

```

In [3]: images = tf.convert_to_tensor(images)
#images = tf.image.grayscale_to_rgb(images)
emotions = tf.convert_to_tensor(emotions)
#images = tf.image.resize(images, [224,224])
images = layers.Rescaling(1./127.5, offset= -1)(images)

training_size = training_samples + validation_samples
test_size = test_samples

training_images = images[:training_size]
test_images = images[training_size:]
training_emotions = emotions[:training_size]
test_emotions = emotions[training_size:]

print("training_images shape:", training_images.shape)
print("training_emotions shape:", training_emotions.shape)
print("test_images shape:", test_images.shape)
print("test_emotions shape:", test_emotions.shape)

```

```

training_images shape: (95574, 48, 48, 1)
training_emotions shape: (95574, 8)
test_images shape: (3535, 48, 48, 1)
test_emotions shape: (3535, 8)

```

```

In [4]: from tensorflow.python.keras.applications import vgg16, resnet_v2
from tensorflow.python.keras import optimizers
from tensorflow.python.keras.optimizer_v2 import adam

```

```

In [5]: base_model = vgg16.VGG16(include_top=False,
                                weights="imagenet",
                                input_shape=(48,48,3))
base_model.trainable=True
model = Sequential([
    base_model,
    layers.GlobalAveragePooling2D(),

```

```

layers.Dense(4096, activation='relu'),
layers.Dense(4096, activation='relu'),
layers.Dense(emotions_count, activation='softmax'),
])

model.compile(optimizer=adam.Adam(learning_rate=1e-4),
              loss=mse,
              metrics = [model_acc])

model.fit(x=tf.image.grayscale_to_rgb(training_images),
          y=training_emotions,
          batch_size=32,
          epochs=40,
          validation_data=(tf.image.grayscale_to_rgb(test_images), test_emotions))

```

/userhome/2072/fyp21022/anaconda3/lib/python3.8/site-packages/tensorflow/python/data/ops/dataset\_ops.py:3503: UserWarning: Even though the tf.config.experimental\_run\_functions\_eagerly option is set, this option does not apply to tf.data functions. tf.data functions are still traced and executed as graphs.

```

warnings.warn(
Epoch 1/40
2987/2987 [=====] - 384s 112ms/step - loss: 0.0370 - model_acc: 0.6021 - val_loss: 0.0219 - val_
model_acc: 0.7514
Epoch 2/40
2987/2987 [=====] - 256s 86ms/step - loss: 0.0179 - model_acc: 0.7871 - val_loss: 0.0164 - val_m
odel_acc: 0.8013
Epoch 3/40
2987/2987 [=====] - 263s 88ms/step - loss: 0.0132 - model_acc: 0.8367 - val_loss: 0.0148 - val_m
odel_acc: 0.8146
Epoch 4/40
2987/2987 [=====] - 261s 87ms/step - loss: 0.0105 - model_acc: 0.8674 - val_loss: 0.0141 - val_m
odel_acc: 0.8228
Epoch 5/40
2987/2987 [=====] - 260s 87ms/step - loss: 0.0084 - model_acc: 0.8928 - val_loss: 0.0138 - val_m
odel_acc: 0.8196
Epoch 6/40
2987/2987 [=====] - 272s 91ms/step - loss: 0.0069 - model_acc: 0.9114 - val_loss: 0.0136 - val_m
odel_acc: 0.8265
Epoch 7/40
2987/2987 [=====] - 268s 90ms/step - loss: 0.0058 - model_acc: 0.9267 - val_loss: 0.0133 - val_m
odel_acc: 0.8349
Epoch 8/40
2987/2987 [=====] - 272s 91ms/step - loss: 0.0050 - model_acc: 0.9351 - val_loss: 0.0137 - val_m
odel_acc: 0.8282
Epoch 9/40
2987/2987 [=====] - 258s 87ms/step - loss: 0.0043 - model_acc: 0.9441 - val_loss: 0.0133 - val_m
odel_acc: 0.8315
Epoch 10/40
2987/2987 [=====] - 266s 89ms/step - loss: 0.0038 - model_acc: 0.9505 - val_loss: 0.0135 - val_m

```

```
odel_acc: 0.8293
Epoch 11/40
2987/2987 [=====] - 275s 92ms/step - loss: 0.0036 - model_acc: 0.9522 - val_loss: 0.0129 - val_m
odel_acc: 0.8383
Epoch 12/40
2987/2987 [=====] - 273s 91ms/step - loss: 0.0030 - model_acc: 0.9607 - val_loss: 0.0126 - val_m
odel_acc: 0.8397
Epoch 13/40
2987/2987 [=====] - 269s 90ms/step - loss: 0.0028 - model_acc: 0.9614 - val_loss: 0.0132 - val_m
odel_acc: 0.8360
Epoch 14/40
2987/2987 [=====] - 266s 89ms/step - loss: 0.0026 - model_acc: 0.9652 - val_loss: 0.0128 - val_m
odel_acc: 0.8380
Epoch 15/40
2987/2987 [=====] - 274s 92ms/step - loss: 0.0023 - model_acc: 0.9693 - val_loss: 0.0130 - val_m
odel_acc: 0.8400
Epoch 16/40
2987/2987 [=====] - 266s 89ms/step - loss: 0.0022 - model_acc: 0.9678 - val_loss: 0.0126 - val_m
odel_acc: 0.8382
Epoch 17/40
2987/2987 [=====] - 257s 86ms/step - loss: 0.0020 - model_acc: 0.9736 - val_loss: 0.0124 - val_m
odel_acc: 0.8462
Epoch 18/40
2987/2987 [=====] - 274s 92ms/step - loss: 0.0019 - model_acc: 0.9732 - val_loss: 0.0123 - val_m
odel_acc: 0.8487
Epoch 19/40
2987/2987 [=====] - 268s 90ms/step - loss: 0.0017 - model_acc: 0.9752 - val_loss: 0.0126 - val_m
odel_acc: 0.8473
Epoch 20/40
2987/2987 [=====] - 248s 83ms/step - loss: 0.0017 - model_acc: 0.9764 - val_loss: 0.0125 - val_m
odel_acc: 0.8405
Epoch 21/40
2987/2987 [=====] - 275s 92ms/step - loss: 0.0015 - model_acc: 0.9795 - val_loss: 0.0124 - val_m
odel_acc: 0.8467
Epoch 22/40
2987/2987 [=====] - 287s 96ms/step - loss: 0.0014 - model_acc: 0.9803 - val_loss: 0.0122 - val_m
odel_acc: 0.8473
Epoch 23/40
2987/2987 [=====] - 273s 91ms/step - loss: 0.0014 - model_acc: 0.9801 - val_loss: 0.0123 - val_m
odel_acc: 0.8433
Epoch 24/40
2987/2987 [=====] - 257s 86ms/step - loss: 0.0012 - model_acc: 0.9836 - val_loss: 0.0129 - val_m
odel_acc: 0.8430
Epoch 25/40
2987/2987 [=====] - 255s 85ms/step - loss: 0.0012 - model_acc: 0.9841 - val_loss: 0.0121 - val_m
odel_acc: 0.8498
Epoch 26/40
2987/2987 [=====] - 275s 92ms/step - loss: 0.0010 - model_acc: 0.9859 - val_loss: 0.0122 - val_m
odel_acc: 0.8467
```

```

Epoch 27/40
2987/2987 [=====] - 273s 91ms/step - loss: 0.0011 - model_acc: 0.9847 - val_loss: 0.0123 - val_m
odel_acc: 0.8461
Epoch 28/40
2987/2987 [=====] - 269s 90ms/step - loss: 0.0010 - model_acc: 0.9848 - val_loss: 0.0125 - val_m
odel_acc: 0.8422
Epoch 29/40
2987/2987 [=====] - 277s 93ms/step - loss: 9.4432e-04 - model_acc: 0.9867 - val_loss: 0.0126 - v
al_model_acc: 0.8453
Epoch 30/40
2987/2987 [=====] - 241s 81ms/step - loss: 9.0584e-04 - model_acc: 0.9878 - val_loss: 0.0126 - v
al_model_acc: 0.8470
Epoch 31/40
2987/2987 [=====] - 271s 91ms/step - loss: 8.6584e-04 - model_acc: 0.9885 - val_loss: 0.0126 - v
al_model_acc: 0.8427
Epoch 32/40
2987/2987 [=====] - 274s 92ms/step - loss: 8.2724e-04 - model_acc: 0.9888 - val_loss: 0.0122 - v
al_model_acc: 0.8478
Epoch 33/40
2987/2987 [=====] - 285s 96ms/step - loss: 7.0494e-04 - model_acc: 0.9924 - val_loss: 0.0123 - v
al_model_acc: 0.8453
Epoch 34/40
2987/2987 [=====] - 272s 91ms/step - loss: 7.1362e-04 - model_acc: 0.9908 - val_loss: 0.0122 - v
al_model_acc: 0.8459
Epoch 35/40
2987/2987 [=====] - 284s 95ms/step - loss: 6.7124e-04 - model_acc: 0.9921 - val_loss: 0.0130 - v
al_model_acc: 0.8380
Epoch 36/40
2987/2987 [=====] - 272s 91ms/step - loss: 7.7669e-04 - model_acc: 0.9884 - val_loss: 0.0126 - v
al_model_acc: 0.8419
Epoch 37/40
2987/2987 [=====] - 274s 92ms/step - loss: 9.9542e-04 - model_acc: 0.9848 - val_loss: 0.0124 - v
al_model_acc: 0.8453
Epoch 38/40
2987/2987 [=====] - 274s 92ms/step - loss: 5.7162e-04 - model_acc: 0.9934 - val_loss: 0.0125 - v
al_model_acc: 0.8419
Epoch 39/40
2987/2987 [=====] - 270s 90ms/step - loss: 5.2587e-04 - model_acc: 0.9936 - val_loss: 0.0125 - v
al_model_acc: 0.8422
Epoch 40/40
2987/2987 [=====] - 275s 92ms/step - loss: 6.7488e-04 - model_acc: 0.9901 - val_loss: 0.0124 - v
al_model_acc: 0.8376

```

Out[5]: <tensorflow.python.keras.callbacks.History at 0x14c4cda96160>