In [1]:

```
# data augmentation: mirror and rotate +-25 degree (use read_dataset3, dataset3)
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']

samples = 130967 # 2~130968
training_samples = 28317*4  # 2~113269 (Training)
validation_samples = 3541*4 # 113270~127433 (PublicTest)
test_samples = 3535         # 127434~130968 (PrivateTest)

image_path = "./dataset3/images.npy"
emotion_multi_path = "./dataset3/emotions_multi.npy"
emotion_single_path = "./dataset3/emotions_single.npy"
```

In [2]:

```
images = np.load(image_path)
emotions_multi = np.load(emotion_multi_path)
emotions_single = np.load(emotion_single_path)

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

```
(130967, 48, 48, 1)
(130967, 8)
(130967, 8)
```

In [3]:

```
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 [4]:

```
#emotions = emotions_single
emotions = emotions_multi

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: (127432, 48, 48, 1)
training_emotions shape: (127432, 8)
test_images shape: (3535, 48, 48, 1)
test_emotions shape: (3535, 8)
```

In [5]:

```
from tensorflow.python.keras.applications import vgg16, resnet_v2
from tensorflow.python.keras import optimizers
from tensorflow.python.keras.optimizer_v2 import adam
```

In [6]:

```
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=losses.CategoricalCrossentropy(),
              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))
```

```
C:\Users\Darkl\anaconda3\lib\site-packages\tensorflow\python\data\ops\dataset_ops.py:3703: UserWarning: Even though the `tf.c
onfig.experimental_run_functions_eagerly` option is set, this option does not apply to tf.data functions. To force eager exec
ution of tf.data functions, please use `tf.data.experimental.enable.debug_mode()`.
  warnings.warn(
Epoch 1/40
3983/3983 [==============================] - 261s 65ms/step - loss: 1.0984 - model_acc: 0.7154 - val_loss: 0.8909 - val_model
_acc: 0.8058
Epoch 2/40
3983/3983 [==============================] - 255s 64ms/step - loss: 0.8873 - model_acc: 0.8218 - val_loss: 0.8600 - val_model
_acc: 0.8205
Epoch 3/40
3983/3983 [==============================] - 252s 63ms/step - loss: 0.8217 - model_acc: 0.8575 - val_loss: 0.8375 - val_model
_acc: 0.8337
Epoch 4/40
3983/3983 [==============================] - 256s 64ms/step - loss: 0.7768 - model_acc: 0.8823 - val_loss: 0.8255 - val_model
_acc: 0.8425
Epoch 5/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.7434 - model_acc: 0.9020 - val_loss: 0.8253 - val_model
_acc: 0.8462
Epoch 6/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.7156 - model_acc: 0.9165 - val_loss: 0.8244 - val_model
_acc: 0.8485
Epoch 7/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.6904 - model_acc: 0.9278 - val_loss: 0.8356 - val_model
_acc: 0.8475
```

```
Epoch 8/40
3983/3983 [==============================] - 256s 64ms/step - loss: 0.6680 - model_acc: 0.9365 - val_loss: 0.8405 - val_model
_acc: 0.8481
Epoch 9/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.6478 - model_acc: 0.9408 - val_loss: 0.8503 - val_model
_acc: 0.8490
Epoch 10/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.6293 - model_acc: 0.9466 - val_loss: 0.8787 - val_model
_acc: 0.8496
Epoch 11/40
3983/3983 [==============================] - 253s 63ms/step - loss: 0.6141 - model_acc: 0.9492 - val_loss: 0.8987 - val_model
_acc: 0.8547
Epoch 12/40
3983/3983 [==============================] - 255s 64ms/step - loss: 0.6026 - model_acc: 0.9524 - val_loss: 0.9469 - val_model
_acc: 0.8442
Epoch 13/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.5940 - model_acc: 0.9537 - val_loss: 0.9828 - val_model
_acc: 0.8493
Epoch 14/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5878 - model_acc: 0.9558 - val_loss: 0.9684 - val_model
_acc: 0.8504
Epoch 15/40
3983/3983 [==============================] - 255s 64ms/step - loss: 0.5830 - model_acc: 0.9570 - val_loss: 1.0088 - val_model
_acc: 0.8487
Epoch 16/40
3983/3983 [==============================] - 255s 64ms/step - loss: 0.5788 - model_acc: 0.9594 - val_loss: 1.0041 - val_model
_acc: 0.8414
Epoch 17/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5758 - model_acc: 0.9598 - val_loss: 1.0091 - val_model
_acc: 0.8447
Epoch 18/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5728 - model_acc: 0.9610 - val_loss: 1.0269 - val_model
_acc: 0.8473
Epoch 19/40
3983/3983 [==============================] - 255s 64ms/step - loss: 0.5711 - model_acc: 0.9629 - val_loss: 1.0121 - val_model
_acc: 0.8515
Epoch 20/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5692 - model_acc: 0.9629 - val_loss: 1.0636 - val_model
_acc: 0.8476
Epoch 21/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5674 - model_acc: 0.9640 - val_loss: 1.0569 - val_model
_acc: 0.8484
Epoch 22/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5656 - model_acc: 0.9648 - val_loss: 1.0689 - val_model
```

```
_acc: 0.8459
Epoch 23/40
3983/3983 [==============================] - 256s 64ms/step - loss: 0.5645 - model_acc: 0.9651 - val_loss: 1.0637 - val_model
_acc: 0.8481
Epoch 24/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5631 - model_acc: 0.9660 - val_loss: 1.1008 - val_model
_acc: 0.8444
Epoch 25/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5619 - model_acc: 0.9668 - val_loss: 1.0627 - val_model
_acc: 0.8501
Epoch 26/40
3983/3983 [==============================] - 256s 64ms/step - loss: 0.5611 - model_acc: 0.9676 - val_loss: 1.0843 - val_model
_acc: 0.8495
Epoch 27/40
3983/3983 [==============================] - 257s 65ms/step - loss: 0.5600 - model_acc: 0.9684 - val_loss: 1.1049 - val_model
_acc: 0.8450
Epoch 28/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.5590 - model_acc: 0.9689 - val_loss: 1.1050 - val_model
_acc: 0.8448
Epoch 29/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.5579 - model_acc: 0.9693 - val_loss: 1.1025 - val_model
_acc: 0.8479
Epoch 30/40
3983/3983 [==============================] - 253s 63ms/step - loss: 0.5572 - model_acc: 0.9696 - val_loss: 1.0986 - val_model
_acc: 0.8498
Epoch 31/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5570 - model_acc: 0.9701 - val_loss: 1.1366 - val_model
_acc: 0.8459
Epoch 32/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5557 - model_acc: 0.9708 - val_loss: 1.0924 - val_model
_acc: 0.8444
Epoch 33/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5559 - model_acc: 0.9709 - val_loss: 1.0981 - val_model
_acc: 0.8425
Epoch 34/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.5545 - model_acc: 0.9724 - val_loss: 1.1554 - val_model
_acc: 0.8479
Epoch 35/40
3983/3983 [==============================] - 256s 64ms/step - loss: 0.5540 - model_acc: 0.9727 - val_loss: 1.1195 - val_model
_acc: 0.8470
Epoch 36/40
3983/3983 [==============================] - 253s 64ms/step - loss: 0.5539 - model_acc: 0.9728 - val_loss: 1.1160 - val_model
_acc: 0.8461
Epoch 37/40
```

```
3983/3983 [==============================] - 254s 64ms/step - loss: 0.5530 - model_acc: 0.9734 - val_loss: 1.1019 - val_model
_acc: 0.8524
Epoch 38/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.5527 - model_acc: 0.9734 - val_loss: 1.0905 - val_model
_acc: 0.8487
Epoch 39/40
3983/3983 [==============================] - 255s 64ms/step - loss: 0.5523 - model_acc: 0.9738 - val_loss: 1.1045 - val_model
_acc: 0.8501
Epoch 40/40
3983/3983 [==============================] - 254s 64ms/step - loss: 0.5517 - model_acc: 0.9749 - val_loss: 1.1483 - val_model
_acc: 0.8462
```

Out[6]:

```
<tensorflow.python.keras.callbacks.History at 0x17b807f3640>
```

In [7]:

```
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=losses.CategoricalCrossentropy(),
              metrics = "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))
```

```
---------------------------------------------------------------------------
ResourceExhaustedError                    Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_16324/164833257.py in <module>
     15               metrics = "acc")
     16 
---> 17 model.fit(x=tf.image.grayscale_to_rgb(training_images),
     18           y=training_emotions,
     19           batch_size=32,
```