In [1]:
```
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']

#data augmentation: mirror version
samples = 67251 # 2~67252
training_samples = 28317*2  # 2~56635 (Training)
validation_samples = 3541*2 # 56636~63717 (PublicTest)
test_samples = 3535         # 63718~67252 (PrivateTest)

image_path = "./dataset/images.npy"
emotion_multi_path = "./dataset/emotions_multi.npy"
emotion_single_path = "./dataset/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)
```

```
(67251, 48, 48, 1)
(67251, 8)
(67251, 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: (63716, 48, 48, 3)
training_emotions shape: (63716, 8)
test_images shape: (3535, 48, 48, 3)
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 = [tf.keras.metrics.TopKCategoricalAccuracy(k=2)])

model.fit(x=training_images,
          y=training_emotions,
          batch_size=32,
          epochs=40,
          validation_data=(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
1992/1992 [==============================] - 82s 39ms/step - loss: 1.0753 - top_k_categorical_accuracy: 0.8758 - val_loss: 0.
9159 - val_top_k_categorical_accuracy: 0.9191
Epoch 2/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.8812 - top_k_categorical_accuracy: 0.9406 - val_loss: 0.
8717 - val_top_k_categorical_accuracy: 0.9364
Epoch 3/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.8179 - top_k_categorical_accuracy: 0.9614 - val_loss: 0.
8667 - val_top_k_categorical_accuracy: 0.9355
Epoch 4/40
1992/1992 [==============================] - 79s 40ms/step - loss: 0.7752 - top_k_categorical_accuracy: 0.9713 - val_loss: 0.
8439 - val_top_k_categorical_accuracy: 0.9454
Epoch 5/40
1992/1992 [==============================] - 80s 40ms/step - loss: 0.7406 - top_k_categorical_accuracy: 0.9794 - val_loss: 0.
8314 - val_top_k_categorical_accuracy: 0.9508
Epoch 6/40
1992/1992 [==============================] - 80s 40ms/step - loss: 0.7159 - top_k_categorical_accuracy: 0.9826 - val_loss: 0.
8394 - val_top_k_categorical_accuracy: 0.9508
Epoch 7/40
1992/1992 [==============================] - 79s 40ms/step - loss: 0.6906 - top_k_categorical_accuracy: 0.9867 - val_loss: 0.
8406 - val_top_k_categorical_accuracy: 0.9482
```

```
Epoch 8/40
1992/1992 [==============================] - 79s 39ms/step - loss: 0.6681 - top_k_categorical_accuracy: 0.9886 - val_loss: 0.
8443 - val_top_k_categorical_accuracy: 0.9530
Epoch 9/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.6471 - top_k_categorical_accuracy: 0.9892 - val_loss: 0.
8674 - val_top_k_categorical_accuracy: 0.9485
Epoch 10/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.6275 - top_k_categorical_accuracy: 0.9905 - val_loss: 0.
8920 - val_top_k_categorical_accuracy: 0.9454
Epoch 11/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.6122 - top_k_categorical_accuracy: 0.9914 - val_loss: 0.
9148 - val_top_k_categorical_accuracy: 0.9511
Epoch 12/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.5999 - top_k_categorical_accuracy: 0.9921 - val_loss: 0.
9158 - val_top_k_categorical_accuracy: 0.9468
Epoch 13/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.5902 - top_k_categorical_accuracy: 0.9927 - val_loss: 0.
9558 - val_top_k_categorical_accuracy: 0.9485
Epoch 14/40
1992/1992 [==============================] - 79s 40ms/step - loss: 0.5847 - top_k_categorical_accuracy: 0.9927 - val_loss: 0.
9516 - val_top_k_categorical_accuracy: 0.9499
Epoch 15/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.5798 - top_k_categorical_accuracy: 0.9932 - val_loss: 0.
9720 - val_top_k_categorical_accuracy: 0.9477
Epoch 16/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.5766 - top_k_categorical_accuracy: 0.9934 - val_loss: 0.
9877 - val_top_k_categorical_accuracy: 0.9502
Epoch 17/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.5737 - top_k_categorical_accuracy: 0.9934 - val_loss: 0.
9913 - val_top_k_categorical_accuracy: 0.9502
Epoch 18/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.5713 - top_k_categorical_accuracy: 0.9939 - val_loss: 0.
9986 - val_top_k_categorical_accuracy: 0.9454
Epoch 19/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5688 - top_k_categorical_accuracy: 0.9939 - val_loss: 1.
0079 - val_top_k_categorical_accuracy: 0.9468
Epoch 20/40
1992/1992 [==============================] - 78s 39ms/step - loss: 0.5670 - top_k_categorical_accuracy: 0.9938 - val_loss: 1.
0360 - val_top_k_categorical_accuracy: 0.9522
Epoch 21/40
1992/1992 [==============================] - 80s 40ms/step - loss: 0.5656 - top_k_categorical_accuracy: 0.9940 - val_loss: 1.
0244 - val_top_k_categorical_accuracy: 0.9513
Epoch 22/40
1992/1992 [==============================] - 80s 40ms/step - loss: 0.5647 - top_k_categorical_accuracy: 0.9947 - val_loss: 1.
```

```
0301 - val_top_k_categorical_accuracy: 0.9511
Epoch 23/40
1992/1992 [==============================] - 80s 40ms/step - loss: 0.5632 - top_k_categorical_accuracy: 0.9946 - val_loss: 1.
0506 - val_top_k_categorical_accuracy: 0.9479
Epoch 24/40
1992/1992 [==============================] - 79s 40ms/step - loss: 0.5618 - top_k_categorical_accuracy: 0.9946 - val_loss: 1.
0201 - val_top_k_categorical_accuracy: 0.9528
Epoch 25/40
1992/1992 [==============================] - 79s 40ms/step - loss: 0.5607 - top_k_categorical_accuracy: 0.9948 - val_loss: 1.
0365 - val_top_k_categorical_accuracy: 0.9533
Epoch 26/40
1992/1992 [==============================] - 82s 41ms/step - loss: 0.5607 - top_k_categorical_accuracy: 0.9954 - val_loss: 1.
0446 - val_top_k_categorical_accuracy: 0.9499
Epoch 27/40
1992/1992 [==============================] - 81s 41ms/step - loss: 0.5587 - top_k_categorical_accuracy: 0.9952 - val_loss: 1.
0319 - val_top_k_categorical_accuracy: 0.9511
Epoch 28/40
1992/1992 [==============================] - 82s 41ms/step - loss: 0.5587 - top_k_categorical_accuracy: 0.9953 - val_loss: 1.
0376 - val_top_k_categorical_accuracy: 0.9522
Epoch 29/40
1992/1992 [==============================] - 81s 41ms/step - loss: 0.5579 - top_k_categorical_accuracy: 0.9951 - val_loss: 1.
0628 - val_top_k_categorical_accuracy: 0.9519
Epoch 30/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5566 - top_k_categorical_accuracy: 0.9953 - val_loss: 1.
0493 - val_top_k_categorical_accuracy: 0.9539
Epoch 31/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5564 - top_k_categorical_accuracy: 0.9953 - val_loss: 1.
0768 - val_top_k_categorical_accuracy: 0.9519
Epoch 32/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5555 - top_k_categorical_accuracy: 0.9953 - val_loss: 1.
0414 - val_top_k_categorical_accuracy: 0.9522
Epoch 33/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5547 - top_k_categorical_accuracy: 0.9954 - val_loss: 1.
0758 - val_top_k_categorical_accuracy: 0.9494
Epoch 34/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5544 - top_k_categorical_accuracy: 0.9958 - val_loss: 1.
0628 - val_top_k_categorical_accuracy: 0.9530
Epoch 35/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5537 - top_k_categorical_accuracy: 0.9959 - val_loss: 1.
0594 - val_top_k_categorical_accuracy: 0.9516
Epoch 36/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5530 - top_k_categorical_accuracy: 0.9964 - val_loss: 1.
0806 - val_top_k_categorical_accuracy: 0.9454
Epoch 37/40
```

```
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5529 - top_k_categorical_accuracy: 0.9959 - val_loss: 1.
0939 - val_top_k_categorical_accuracy: 0.9494
Epoch 38/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5522 - top_k_categorical_accuracy: 0.9962 - val_loss: 1.
1154 - val_top_k_categorical_accuracy: 0.9513
Epoch 39/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5515 - top_k_categorical_accuracy: 0.9963 - val_loss: 1.
0534 - val_top_k_categorical_accuracy: 0.9522
Epoch 40/40
1992/1992 [==============================] - 77s 39ms/step - loss: 0.5523 - top_k_categorical_accuracy: 0.9961 - val_loss: 1.
0866 - val_top_k_categorical_accuracy: 0.9471
```

Out[6]:

```
<tensorflow.python.keras.callbacks.History at 0x22bf8ccc790>
```

In [ ]:

```
base_model = resnet_v2.ResNet50V2(include_top=False,
                        weights="imagenet",
                        input_shape=(48,48,3))
base_model.trainable=True
model = Sequential([
    base_model,
    layers.GlobalAveragePooling2D(),
    layers.Dense(2048, activation='relu'),
    layers.Dense(2048, 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=training_images,
          y=training_emotions,
          batch_size=32,
          epochs=40,
          validation_data=(test_images, test_emotions))
```

```
Epoch 1/40
1992/1992 [==============================] - 296s 148ms/step - loss: 1.2006 - model_acc: 0.6661 - val_loss: 1.0267 - val_mode
l_acc: 0.7386
Epoch 2/40
1992/1992 [==============================] - 296s 149ms/step - loss: 0.9611 - model_acc: 0.7832 - val_loss: 0.9534 - val_mode
l_acc: 0.7710
Epoch 3/40
1992/1992 [==============================] - 304s 152ms/step - loss: 0.8764 - model_acc: 0.8283 - val_loss: 0.9463 - val_mode
```

```
l_acc: 0.7831
Epoch 4/40
1992/1992 [==============================] - 298s 149ms/step - loss: 0.8204 - model_acc: 0.8594 - val_loss: 0.9329 - val_mode
l_acc: 0.7881
Epoch 5/40
1992/1992 [==============================] - 298s 149ms/step - loss: 0.7823 - model_acc: 0.8799 - val_loss: 0.9272 - val_mode
l_acc: 0.7998
Epoch 6/40
1992/1992 [==============================] - 297s 149ms/step - loss: 0.7513 - model_acc: 0.8963 - val_loss: 0.9245 - val_mode
l_acc: 0.7972
Epoch 7/40
1992/1992 [==============================] - 296s 148ms/step - loss: 0.7253 - model_acc: 0.9061 - val_loss: 0.9213 - val_mode
l_acc: 0.8084
Epoch 8/40
1992/1992 [==============================] - 304s 153ms/step - loss: 0.7021 - model_acc: 0.9153 - val_loss: 0.9357 - val_mode
l_acc: 0.8067
Epoch 9/40
1992/1992 [==============================] - 299s 150ms/step - loss: 0.6820 - model_acc: 0.9230 - val_loss: 0.9484 - val_mode
l_acc: 0.7955
Epoch 10/40
1992/1992 [==============================] - 299s 150ms/step - loss: 0.6648 - model_acc: 0.9264 - val_loss: 0.9526 - val_mode
l_acc: 0.8112
Epoch 11/40
1992/1992 [==============================] - 308s 155ms/step - loss: 0.6498 - model_acc: 0.9317 - val_loss: 0.9825 - val_mode
l_acc: 0.8051
Epoch 12/40
1992/1992 [==============================] - 311s 156ms/step - loss: 0.6331 - model_acc: 0.9354 - val_loss: 1.0025 - val_mode
l_acc: 0.8030
Epoch 13/40
1992/1992 [==============================] - 317s 159ms/step - loss: 0.6235 - model_acc: 0.9384 - val_loss: 0.9964 - val_mode
l_acc: 0.8138
Epoch 14/40
1992/1992 [==============================] - 314s 158ms/step - loss: 0.6148 - model_acc: 0.9412 - val_loss: 1.0207 - val_mode
l_acc: 0.8126
Epoch 15/40
1992/1992 [==============================] - 319s 160ms/step - loss: 0.6070 - model_acc: 0.9452 - val_loss: 1.0546 - val_mode
l_acc: 0.8188
Epoch 16/40
 633/1992 [========>.....................] - ETA: 3:26 - loss: 0.5951 - model_acc: 0.9481
```

In [ ]: