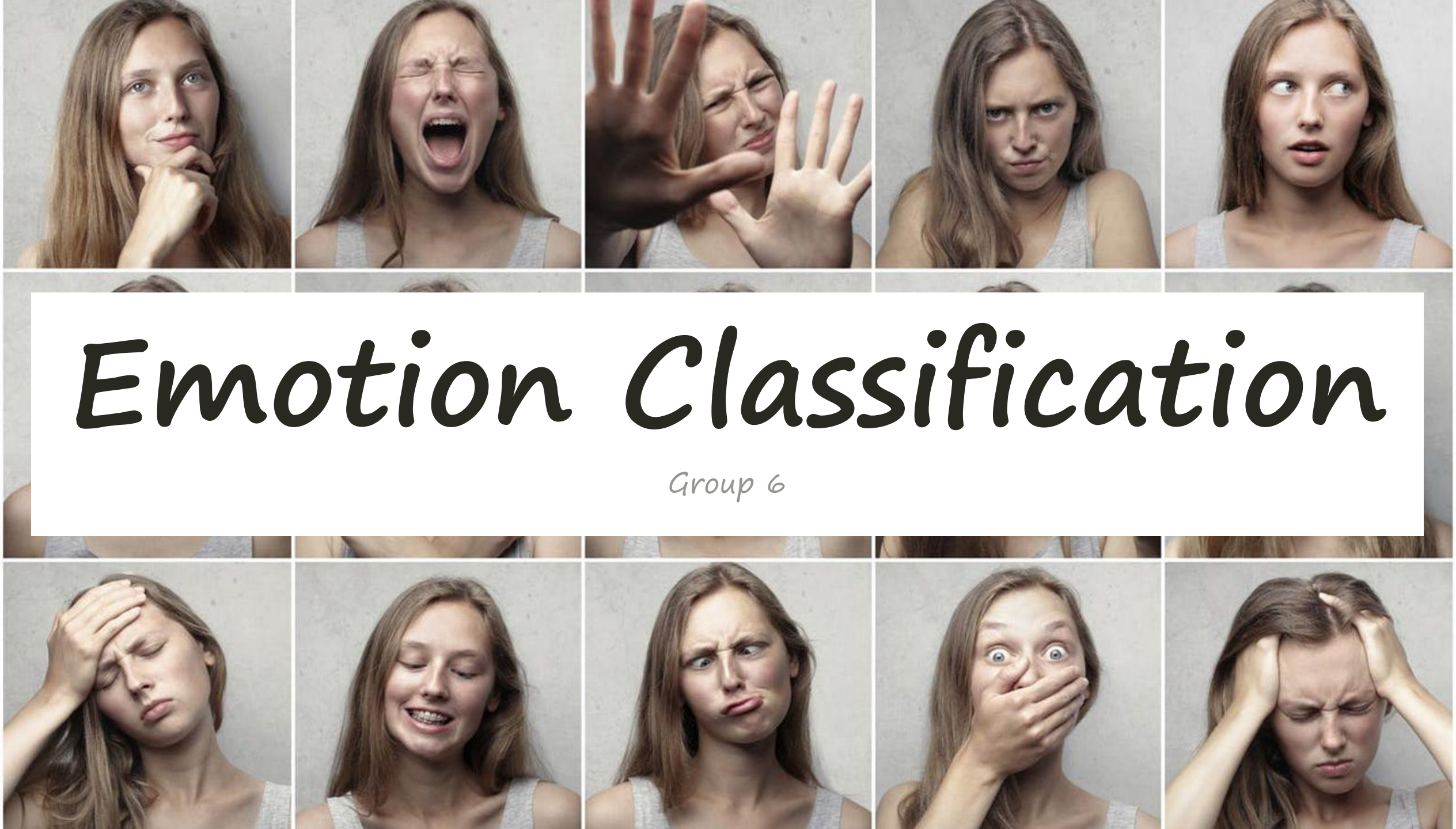
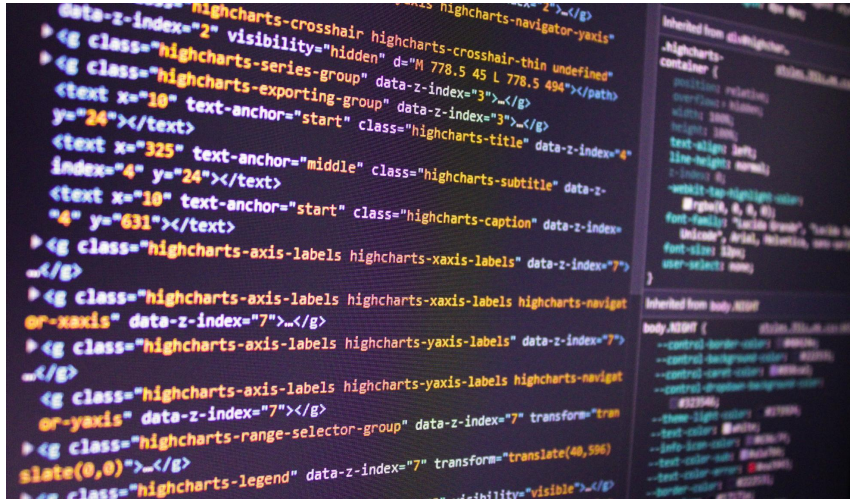


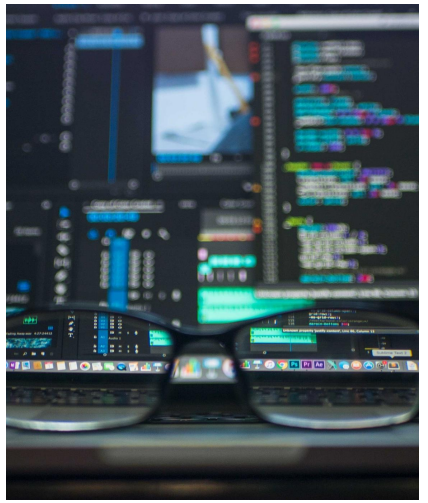
Emotion Classification

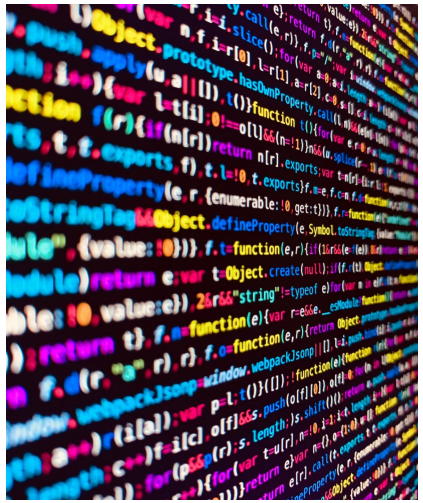
Group 6

Contents

- Data and Data Process
  - Overview of the data and how we preprocess it to be analyzed
- Model Creation
  - Model structure explain
- Result and Analysis
  - Summarize and analyze the result







Data

The data consists of 48x48 pixel grayscale images of faces. The faces have been automatically registered so that the face is more or less centred and occupies about the same amount of space in each image. The task is to categorize each face based on the emotion shown in the facial expression into one of seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). The training set consists of 28,709 examples and the public test set consists of 3,589 examples.

Data Augmentation

```
train_datagen = ImageDataGenerator(#rotation_range = 180,
                                   width_shift_range = 0.1,
                                   height_shift_range = 0.1,
                                   horizontal_flip = True,
                                   rescale = 1./255,
                                   #zoom_range = 0.2,
                                   validation_split = 0.2
                                  )
validation_datagen = ImageDataGenerator(rescale = 1./255,
                                   validation_split = 0.2)
```

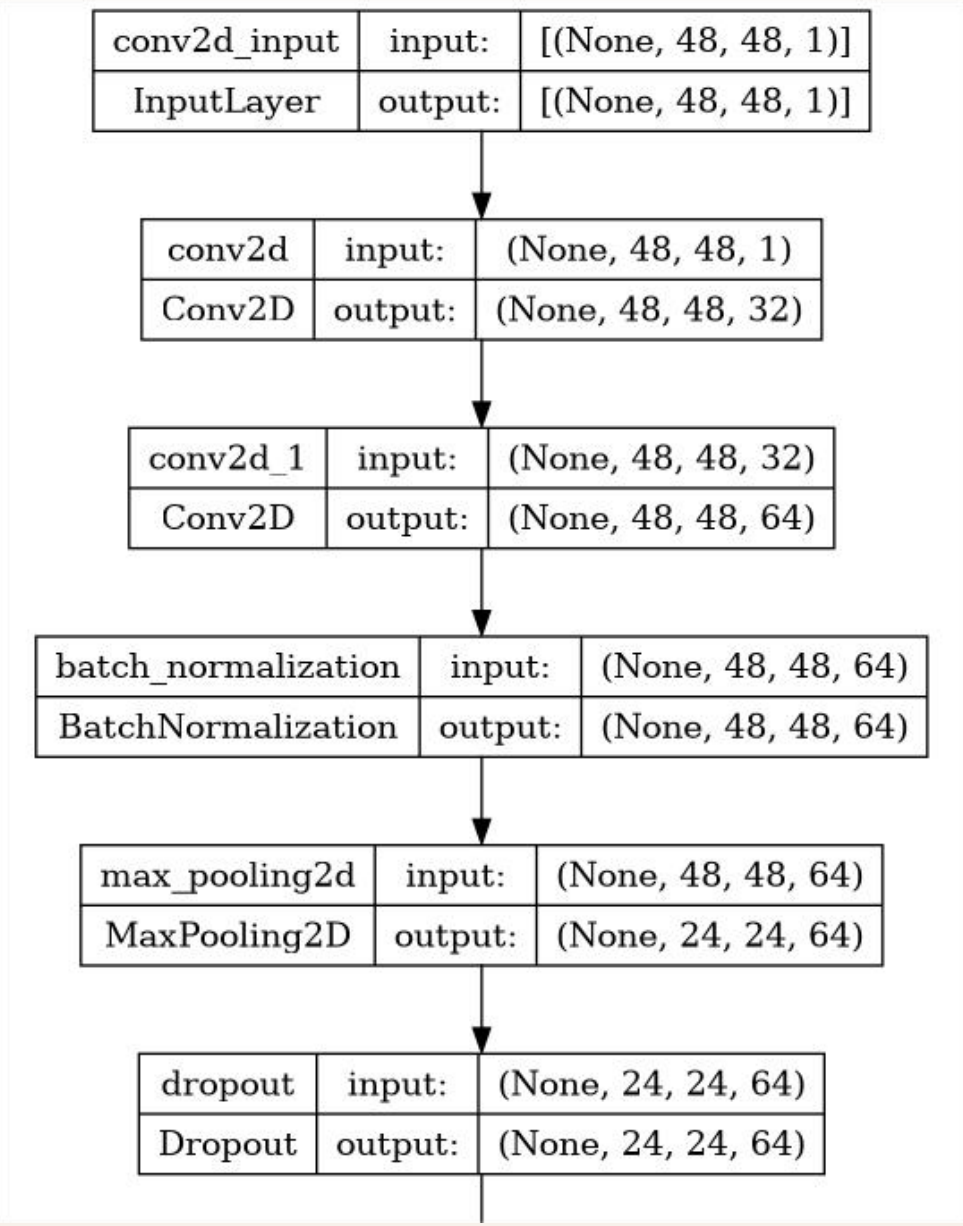
We employ horizontal and vertical shifts, random flipping, and scaling for data augmentation, and we split the dataset into training and validation sets using a specific ratio.

Data Generator

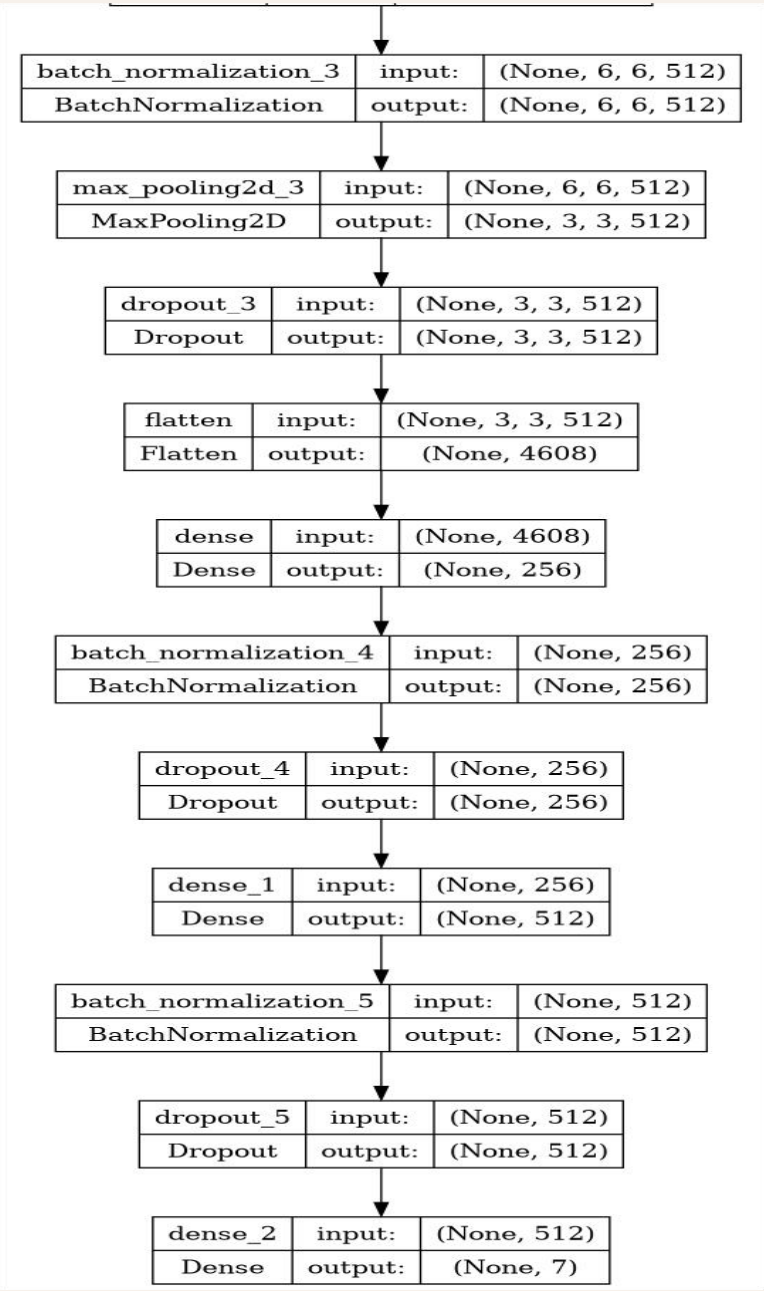
```
train_generator = train_datagen.flow_from_directory(directory = train_dir,
                                                    target_size = (img_size,img_size),
                                                    batch_size = 64,
                                                    color_mode = "grayscale",
                                                    class_mode = "categorical",
                                                    subset = "training"
                                                   )
validation_generator = validation_datagen.flow_from_directory( directory = test_dir,
                                                               target_size = (img_size,img_size),
                                                               batch_size = 64,
                                                               color_mode = "grayscale",
                                                               class_mode = "categorical",
                                                               subset = "validation"
                                                              )
```

This is a multi-class task for grayscale images, and after splitting the data, we need to set the number of images in each batch

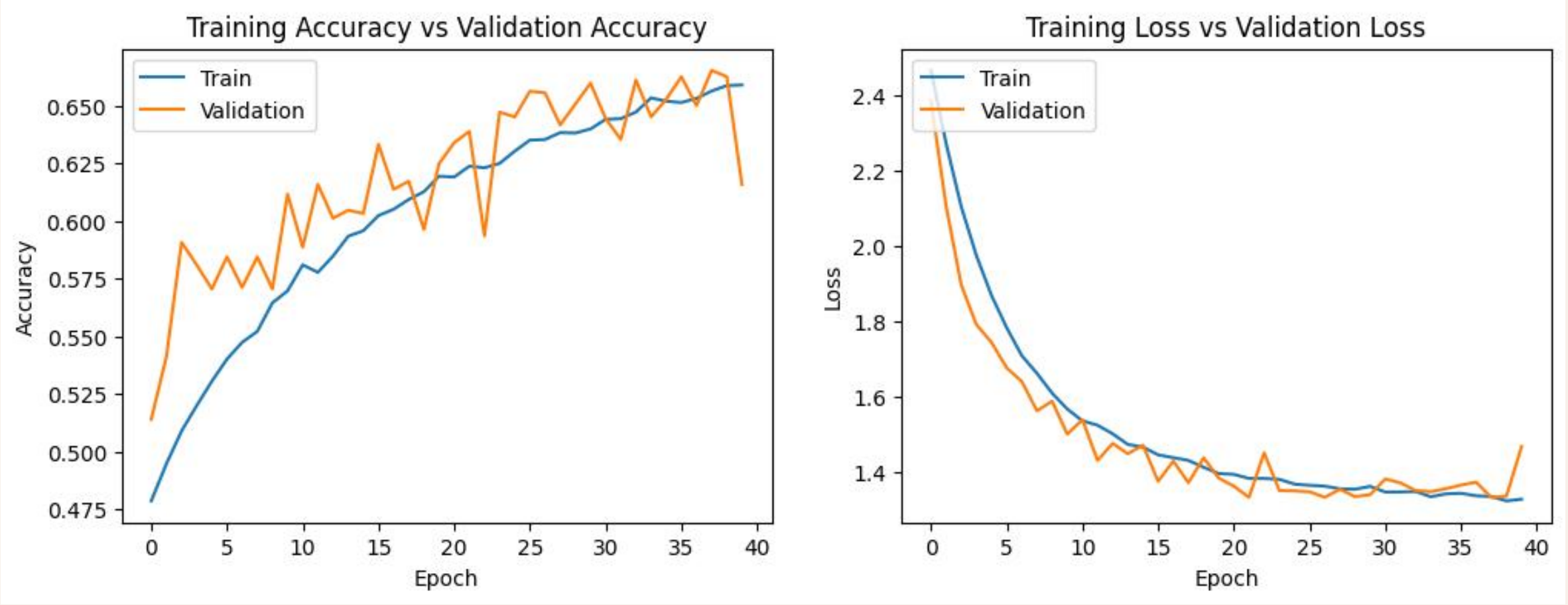
Model Creation



Model Creation



Model Results



Model Results

The results obtained post-training encapsulate a narrative of efficiency and reliability.

With 40 epochs of rigorous training, our model reached a commendable training accuracy of 66.79%. The validation accuracy stood at 61.59%, a testament to the model's ability to generalize and perform adeptly on unseen data.

Thanks!