

07_FoodVision

August 24, 2021

1 Food Vision

1.1 Check GPU & Get Helper Functions

Ideally need a Tesla T4 (i.e. compute score higher than 7.0) to see best speed up from mixed precision training.

```
In [3]: !nvidia-smi -L
```

```
GPU 0: Tesla T4 (UUID: GPU-266d6a40-0295-47c3-a446-910f59edca1a)
```

```
In [4]: !wget "https://raw.githubusercontent.com/mrdbourke/tensorflow-deep-learning/main/extra_helper_functions.py"
from helper_functions import *
```

```
--2021-08-24 18:53:38-- https://raw.githubusercontent.com/mrdbourke/tensorflow-deep-learning/main/extra_helper_functions.py
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.108.133, 185.199.110.133, 185.199.111.133, 185.199.109.133
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.108.133|:443... connected
HTTP request sent, awaiting response... 200 OK
Length: 10246 (10K) [text/plain]
Saving to: helper_functions.py.1
```

```
helper_functions.py 100%[=====] 10.01K --.-KB/s in 0s
```

```
2021-08-24 18:53:38 (101 MB/s) - helper_functions.py.1 saved [10246/10246]
```

1.2 Import data with tensorflow datasets

```
In [5]: import tensorflow as tf
import tensorflow_datasets as tfds
import matplotlib.pyplot as plt
```

```
In [6]: # list all available datasets
datasets_list = tfds.list_builders()
print("food101" in datasets_list) # is food101 in this list?
```

```
True
```

```
In [7]: (train_data, test_data), ds_info = tfds.load(name="food101",
                                                    split=["train", "validation"],
                                                    shuffle_files=True,
                                                    as_supervised=True, # data as tuple i.e.
                                                    with_info=True)
```

Downloading and preparing dataset food101/2.0.0 (download: 4.65 GiB, generated: Unknown size, 1

Dl Completed....: 0 url [00:00, ? url/s]

Dl Size....: 0 MiB [00:00, ? MiB/s]

Extraction completed....: 0 file [00:00, ? file/s]

0 examples [00:00, ? examples/s]

Shuffling and writing examples to /root/tensorflow_datasets/food101/2.0.0.incompleteN4R0EH/food

0%| | 0/75750 [00:00<?, ? examples/s]

0 examples [00:00, ? examples/s]

Shuffling and writing examples to /root/tensorflow_datasets/food101/2.0.0.incompleteN4R0EH/food

0%| | 0/25250 [00:00<?, ? examples/s]

Dataset food101 downloaded and prepared to /root/tensorflow_datasets/food101/2.0.0. Subsequent

1.3 Exploring Food101 data from TensorFlow Datasets

```
In [8]: ds_info.features
```

```
Out[8]: FeaturesDict({
    'image': Image(shape=(None, None, 3), dtype=tf.uint8),
    'label': ClassLabel(shape=(), dtype=tf.int64, num_classes=101),
})
```

```
In [9]: class_names = ds_info.features["label"].names
        class_names[:10]
```

```
Out[9]: ['apple_pie',
        'baby_back_ribs',
        'baklava',
        'beef_carpaccio',
        'beef_tartare',
        'beet_salad',
        'beignets',
        'bibimbap',
        'bread_pudding',
        'breakfast_burrito']
```

```
In [10]: # Take one sample of the train data
         train_one_sample = train_data.take(1)
         train_one_sample
```

```
Out[10]: <TakeDataset shapes: ((None, None, 3), ()), types: (tf.uint8, tf.int64)>
```

Plot an image from tensorflow datasets

```
In [11]: # output info about our training sample
         for image, label in train_one_sample:
             img_class = class_names[label.numpy()]
             print(f"""
             Image shape: {image.shape}
             Image datatype: {image.dtype}
             Target class from Food101 (tensor form): {label}
             Class name (str form): {img_class}
             """)
             # i.e. labels are not one hot encoded - use loss="sparse_crossentropy"
             plt.figure()
             plt.imshow(image)
             plt.title(f"Class name: {img_class}")
             plt.axis(False)
             plt.show()
```

```
Image shape: (512, 512, 3)
Image datatype: <dtype: 'uint8'>
Target class from Food101 (tensor form): 43
Class name (str form): fried_calamari
```

Class name: fried_calamari



1.4 Preprocessing Function

```
In [12]: def preprocess_img(image, label, img_shape=224):  
        """  
        Convert image datatype from 'uint8' > 'float32' and reshapes  
        image to [img_shape, img_shape, 3]  
        """  
        image = tf.image.resize(image, [img_shape, img_shape])  
        return tf.cast(image, tf.float32), label
```

1.5 Data Input Pipeline

```
In [13]: train_data = train_data.map(map_func=preprocess_img, num_parallel_calls=tf.data.AUTOTUNE)  
        train_data = train_data.shuffle(buffer_size=1000).batch(batch_size=32).prefetch(buffer_size=1)  
  
        test_data = test_data.map(map_func=preprocess_img, num_parallel_calls=tf.data.AUTOTUNE)  
        test_data = test_data.shuffle(buffer_size=1000).batch(batch_size=32).prefetch(buffer_size=1)  
  
        print(f"Train Data: {train_data} \nTest Data: {test_data}\n")
```

```
Train Data: <PrefetchDataset shapes: ((None, 224, 224, 3), (None,)), types: (tf.float32, tf.int32)>  
Test Data: <PrefetchDataset shapes: ((None, 224, 224, 3), (None,)), types: (tf.float32, tf.int32)>
```

2 Models 0 & 1 (No Data Aug)

2.1 Feature Extraction Model

```
In [14]: from tensorflow.keras import mixed_precision
         from tensorflow.keras import layers
         from tensorflow.keras.layers.experimental import preprocessing

In [15]: checkpoint_path = "model_checkpoints/cp.ckpt"
         model_checkpoint = tf.keras.callbacks.ModelCheckpoint(checkpoint_path,
                                                                monitor="val_accuracy",
                                                                save_best_only=True,
                                                                save_weights_only=True,
                                                                verbose=0)

         mixed_precision.set_global_policy("mixed_float16")

         # #####
         # FEATURE EXTRACTION MODEL
         # #####
         input_shape=(224, 224, 3)
         base_model = tf.keras.applications.EfficientNetB0(include_top=False)
         base_model.trainable = False

         inputs = layers.Input(shape=input_shape, name="input_layer")
         x = base_model(inputs, training=False)
         x = layers.GlobalAveragePooling2D()(x)
         x = layers.Dense(len(class_names))(x)
         outputs = layers.Activation("softmax", dtype=tf.float32, name="softmax_float32")(x)

         model0 = tf.keras.Model(inputs, outputs, name="FeatureExtraction_ThreeEpochs")

         model0.compile(loss="sparse_categorical_crossentropy",
                        optimizer="adam",
                        metrics=["accuracy"])

         model0.summary()

INFO:tensorflow:Mixed precision compatibility check (mixed_float16): OK
Your GPU will likely run quickly with dtype policy mixed_float16 as it has compute capability 6.1

INFO:tensorflow:Mixed precision compatibility check (mixed_float16): OK
Your GPU will likely run quickly with dtype policy mixed_float16 as it has compute capability 6.1

Downloading data from https://storage.googleapis.com/keras-applications/efficientnetb0_notop.h5
16711680/16705208 [=====] - 0s 0us/step
16719872/16705208 [=====] - 0s 0us/step
Model: "FeatureExtraction_ThreeEpochs"
```

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	[(None, 224, 224, 3)]	0
efficientnetb0 (Functional)	(None, None, None, 1280)	4049571
global_average_pooling2d (Gl	(None, 1280)	0
dense (Dense)	(None, 101)	129381
softmax_float32 (Activation)	(None, 101)	0
Total params: 4,178,952		
Trainable params: 129,381		
Non-trainable params: 4,049,571		

```
In [16]: history_feat_extr = model0.fit(train_data,
                                         epochs=3,
                                         steps_per_epoch=len(train_data),
                                         validation_data=test_data,
                                         validation_steps=int(0.15*len(test_data)),
                                         callbacks=[model_checkpoint,
                                                  create_tensorboard_callback(dir_name="tensorboard",
                                                                              experiment_name="EfficientNetB0_feature_extraction")],
                                         verbose=1)
```

Saving TensorBoard log files to: tensorboard_callbacks/EfficientNetB0_feature_extractor/20210825_102030
Epoch 1/3

```
/usr/local/lib/python3.7/dist-packages/keras/utils/generic_utils.py:497: CustomMaskWarning: Cu
category=CustomMaskWarning)
```

```
2368/2368 [=====] - 222s 78ms/step - loss: 1.8232 - accuracy: 0.5555 -
Epoch 2/3
2368/2368 [=====] - 163s 68ms/step - loss: 1.2946 - accuracy: 0.6648 -
Epoch 3/3
2368/2368 [=====] - 160s 67ms/step - loss: 1.1436 - accuracy: 0.7011 -
```

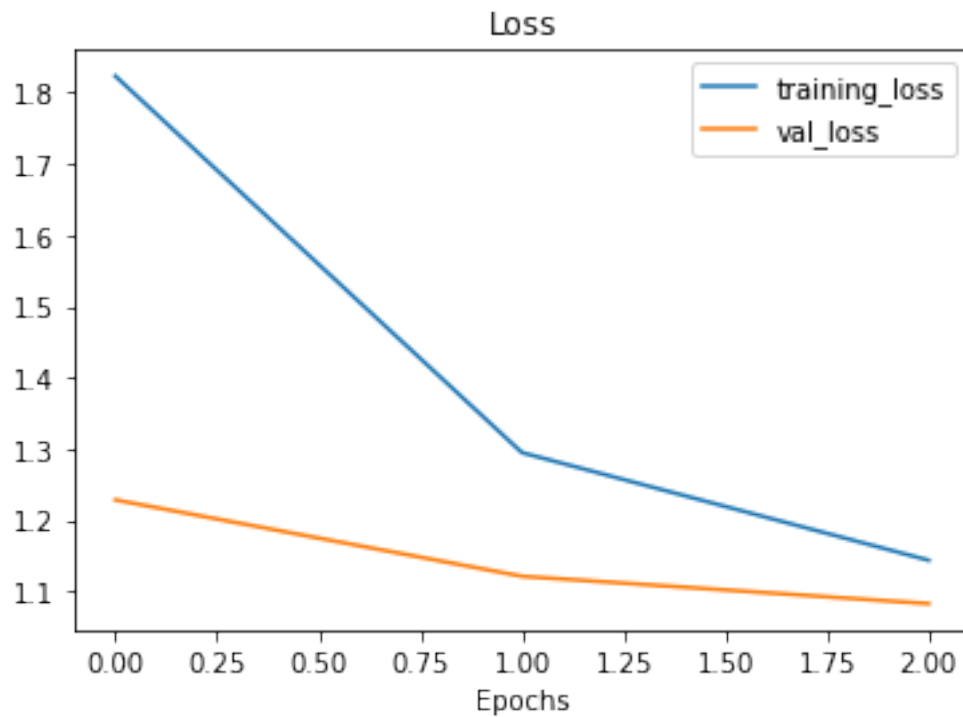
```
In [17]: model0.save("drive/MyDrive/tf/models/FoodVision/featureExtractor")
```

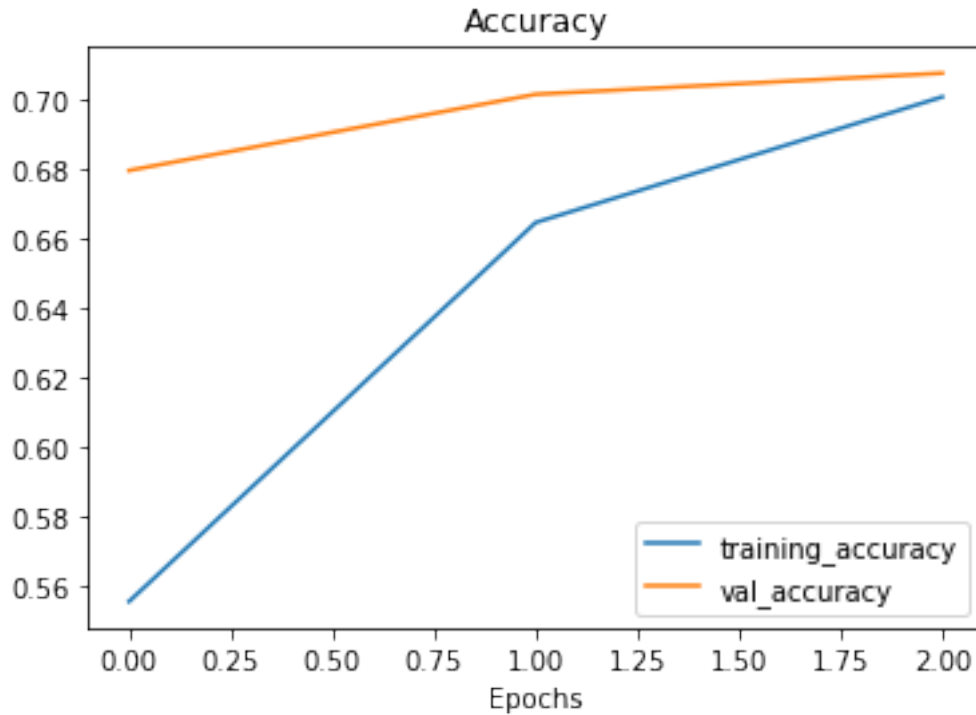
```
INFO:tensorflow:Assets written to: drive/MyDrive/tf/models/FoodVision/featureExtractor/assets
```

```
INFO:tensorflow:Assets written to: drive/MyDrive/tf/models/FoodVision/featureExtractor/assets
/usr/local/lib/python3.7/dist-packages/keras/utils/generic_utils.py:497: CustomMaskWarning: Custom
category=CustomMaskWarning)
```

```
In [18]: plot_loss_curves(history_feat_extr)
        feat_extr_res = model0.evaluate(test_data)
```

```
790/790 [=====] - 55s 68ms/step - loss: 1.0884 - accuracy: 0.7063
```





2.2 Fine Tune Feature Extractor (10 Layers, 3 + 3 Epochs)

In [19]: `model1 = tf.keras.models.load_model("drive/MyDrive/tf/models/FoodVision/featureExtracto`

```
WARNING:absl:Importing a function (__inference_block4c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block5a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block6d_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_FeatureExtraction_ThreeEpochs_layer_call_and_re
WARNING:absl:Importing a function (__inference_block5a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona
WARNING:absl:Importing a function (__inference_block1a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona
WARNING:absl:Importing a function (__inference_block6c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block4c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block3a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block5a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block3a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block3a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona
WARNING:absl:Importing a function (__inference_block6b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block3a_se_reduce_layer_call_and_return_conditi
```


[illegible]

```

WARNING:absl:Importing a function (__inference_block6d_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block7a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block7a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block2b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block3a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6c_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditional
WARNING:absl:Importing a function (__inference_block3a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference__wrapped_model_37605) with ops with unsaved cust
WARNING:absl:Importing a function (__inference_block5c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block2b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block2a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block4c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block4b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6d_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block4a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block2a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block3b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block5a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block2a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block4b_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block1a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block1a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block4b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block4c_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block4c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block4b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block3b_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block4a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block2b_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block6a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block2a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_stem_activation_layer_call_and_return_conditional
WARNING:absl:Importing a function (__inference_block5b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_stem_activation_layer_call_and_return_conditional

```

```

In [20]: # confirm model0 loaded properly into model1
         loaded_model_res = model1.evaluate(test_data)
         print(feat_extr_res[1], loaded_model_res[1], np.isclose(feat_extr_res[1], loaded_model_res[1]))

```

```

790/790 [=====] - 54s 64ms/step - loss: 1.0884 - accuracy: 0.7063
0.7063366174697876 0.7063366174697876 True

```

```
In [21]: model1._name = "Model0_FineTuned_10Layers_SixEpochsTotal"
        model1.summary()
```

Model: "Model0_FineTuned_10Layers_SixEpochsTotal"

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	[(None, 224, 224, 3)]	0
efficientnetb0 (Functional)	(None, None, None, 1280)	4049571
global_average_pooling2d (G1	(None, 1280)	0
dense (Dense)	(None, 101)	129381
softmax_float32 (Activation)	(None, 101)	0

Total params: 4,178,952
 Trainable params: 129,381
 Non-trainable params: 4,049,571

```
In [22]: for layer in model1.layers:
        print(layer.name, layer.trainable)
```

```
input_layer True
efficientnetb0 False
global_average_pooling2d True
dense True
softmax_float32 True
```

```
In [23]: model1.layers[1].trainable=True
        for layer in model1.layers[1].layers[:-10]:
            layer.trainable = False
        for i, layer in enumerate(model1.layers[1].layers):
            print(i, layer.name, layer.trainable)
```

```
0 input_1 False
1 rescaling False
2 normalization False
3 stem_conv_pad False
4 stem_conv False
5 stem_bn False
6 stem_activation False
7 block1a_dwconv False
8 block1a_bn False
9 block1a_activation False
```

10 block1a_se_squeeze False
11 block1a_se_reshape False
12 block1a_se_reduce False
13 block1a_se_expand False
14 block1a_se_excite False
15 block1a_project_conv False
16 block1a_project_bn False
17 block2a_expand_conv False
18 block2a_expand_bn False
19 block2a_expand_activation False
20 block2a_dwconv_pad False
21 block2a_dwconv False
22 block2a_bn False
23 block2a_activation False
24 block2a_se_squeeze False
25 block2a_se_reshape False
26 block2a_se_reduce False
27 block2a_se_expand False
28 block2a_se_excite False
29 block2a_project_conv False
30 block2a_project_bn False
31 block2b_expand_conv False
32 block2b_expand_bn False
33 block2b_expand_activation False
34 block2b_dwconv False
35 block2b_bn False
36 block2b_activation False
37 block2b_se_squeeze False
38 block2b_se_reshape False
39 block2b_se_reduce False
40 block2b_se_expand False
41 block2b_se_excite False
42 block2b_project_conv False
43 block2b_project_bn False
44 block2b_drop False
45 block2b_add False
46 block3a_expand_conv False
47 block3a_expand_bn False
48 block3a_expand_activation False
49 block3a_dwconv_pad False
50 block3a_dwconv False
51 block3a_bn False
52 block3a_activation False
53 block3a_se_squeeze False
54 block3a_se_reshape False
55 block3a_se_reduce False
56 block3a_se_expand False
57 block3a_se_excite False

58 block3a_project_conv False
59 block3a_project_bn False
60 block3b_expand_conv False
61 block3b_expand_bn False
62 block3b_expand_activation False
63 block3b_dwconv False
64 block3b_bn False
65 block3b_activation False
66 block3b_se_squeeze False
67 block3b_se_reshape False
68 block3b_se_reduce False
69 block3b_se_expand False
70 block3b_se_excite False
71 block3b_project_conv False
72 block3b_project_bn False
73 block3b_drop False
74 block3b_add False
75 block4a_expand_conv False
76 block4a_expand_bn False
77 block4a_expand_activation False
78 block4a_dwconv_pad False
79 block4a_dwconv False
80 block4a_bn False
81 block4a_activation False
82 block4a_se_squeeze False
83 block4a_se_reshape False
84 block4a_se_reduce False
85 block4a_se_expand False
86 block4a_se_excite False
87 block4a_project_conv False
88 block4a_project_bn False
89 block4b_expand_conv False
90 block4b_expand_bn False
91 block4b_expand_activation False
92 block4b_dwconv False
93 block4b_bn False
94 block4b_activation False
95 block4b_se_squeeze False
96 block4b_se_reshape False
97 block4b_se_reduce False
98 block4b_se_expand False
99 block4b_se_excite False
100 block4b_project_conv False
101 block4b_project_bn False
102 block4b_drop False
103 block4b_add False
104 block4c_expand_conv False
105 block4c_expand_bn False

106 block4c_expand_activation False
107 block4c_dwconv False
108 block4c_bn False
109 block4c_activation False
110 block4c_se_squeeze False
111 block4c_se_reshape False
112 block4c_se_reduce False
113 block4c_se_expand False
114 block4c_se_excite False
115 block4c_project_conv False
116 block4c_project_bn False
117 block4c_drop False
118 block4c_add False
119 block5a_expand_conv False
120 block5a_expand_bn False
121 block5a_expand_activation False
122 block5a_dwconv False
123 block5a_bn False
124 block5a_activation False
125 block5a_se_squeeze False
126 block5a_se_reshape False
127 block5a_se_reduce False
128 block5a_se_expand False
129 block5a_se_excite False
130 block5a_project_conv False
131 block5a_project_bn False
132 block5b_expand_conv False
133 block5b_expand_bn False
134 block5b_expand_activation False
135 block5b_dwconv False
136 block5b_bn False
137 block5b_activation False
138 block5b_se_squeeze False
139 block5b_se_reshape False
140 block5b_se_reduce False
141 block5b_se_expand False
142 block5b_se_excite False
143 block5b_project_conv False
144 block5b_project_bn False
145 block5b_drop False
146 block5b_add False
147 block5c_expand_conv False
148 block5c_expand_bn False
149 block5c_expand_activation False
150 block5c_dwconv False
151 block5c_bn False
152 block5c_activation False
153 block5c_se_squeeze False

154 block5c_se_reshape False
155 block5c_se_reduce False
156 block5c_se_expand False
157 block5c_se_excite False
158 block5c_project_conv False
159 block5c_project_bn False
160 block5c_drop False
161 block5c_add False
162 block6a_expand_conv False
163 block6a_expand_bn False
164 block6a_expand_activation False
165 block6a_dwconv_pad False
166 block6a_dwconv False
167 block6a_bn False
168 block6a_activation False
169 block6a_se_squeeze False
170 block6a_se_reshape False
171 block6a_se_reduce False
172 block6a_se_expand False
173 block6a_se_excite False
174 block6a_project_conv False
175 block6a_project_bn False
176 block6b_expand_conv False
177 block6b_expand_bn False
178 block6b_expand_activation False
179 block6b_dwconv False
180 block6b_bn False
181 block6b_activation False
182 block6b_se_squeeze False
183 block6b_se_reshape False
184 block6b_se_reduce False
185 block6b_se_expand False
186 block6b_se_excite False
187 block6b_project_conv False
188 block6b_project_bn False
189 block6b_drop False
190 block6b_add False
191 block6c_expand_conv False
192 block6c_expand_bn False
193 block6c_expand_activation False
194 block6c_dwconv False
195 block6c_bn False
196 block6c_activation False
197 block6c_se_squeeze False
198 block6c_se_reshape False
199 block6c_se_reduce False
200 block6c_se_expand False
201 block6c_se_excite False

```

202 block6c_project_conv False
203 block6c_project_bn False
204 block6c_drop False
205 block6c_add False
206 block6d_expand_conv False
207 block6d_expand_bn False
208 block6d_expand_activation False
209 block6d_dwconv False
210 block6d_bn False
211 block6d_activation False
212 block6d_se_squeeze False
213 block6d_se_reshape False
214 block6d_se_reduce False
215 block6d_se_expand False
216 block6d_se_excite False
217 block6d_project_conv False
218 block6d_project_bn False
219 block6d_drop False
220 block6d_add False
221 block7a_expand_conv False
222 block7a_expand_bn False
223 block7a_expand_activation False
224 block7a_dwconv False
225 block7a_bn False
226 block7a_activation False
227 block7a_se_squeeze True
228 block7a_se_reshape True
229 block7a_se_reduce True
230 block7a_se_expand True
231 block7a_se_excite True
232 block7a_project_conv True
233 block7a_project_bn True
234 top_conv True
235 top_bn True
236 top_activation True

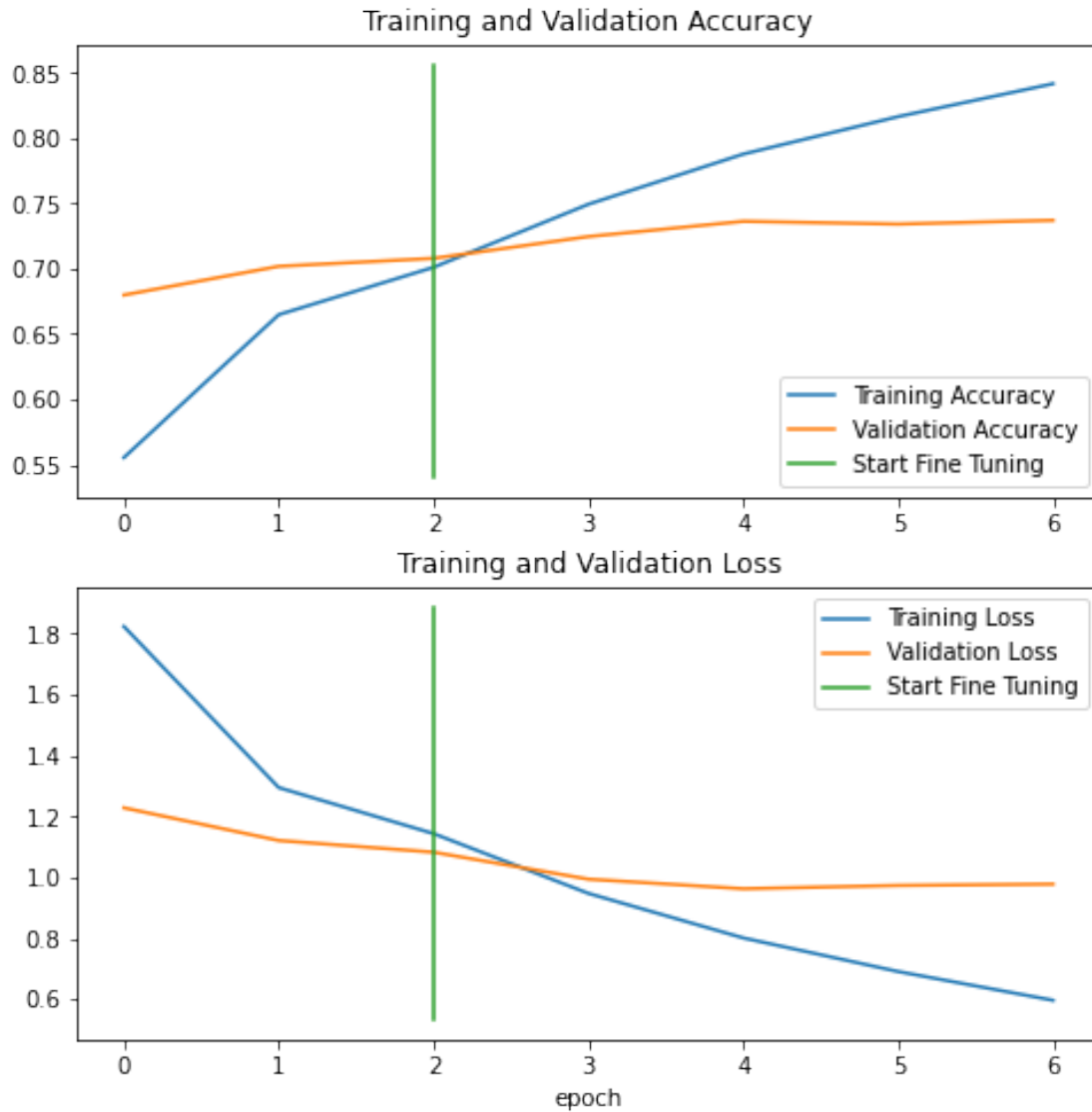
```

```

In [24]: checkpoint_path = "model_checkpoints/fine_tuned.ckpt"
        model_checkpoint = tf.keras.callbacks.ModelCheckpoint(checkpoint_path,
                                                                monitor="val_accuracy",
                                                                save_best_only=True,
                                                                save_weights_only=True,
                                                                verbose=0)

        model1.compile(loss="sparse_categorical_crossentropy",
                        optimizer=tf.keras.optimizers.Adam(learning_rate=1e-4),
                        metrics=["accuracy"])

```

790/790 [=====] - 55s 68ms/step - loss: 0.9724 - accuracy: 0.7397

Feature Extractor Accuracy: 70.63%

Fine Tuned Accuracy: 73.97%

3 Models 2 and 3 (Data Aug)

3.1 Augment dataset

```
In [27]: from tensorflow.keras import layers
         from tensorflow.keras.layers.experimental import preprocessing
         from tensorflow.keras.models import Sequential
```

```

In [28]: def augment_img(image, label, img_shape=224):
        """
        Convert image datatype from 'uint8' > 'float32' and reshapes
        image to [img_shape, img_shape, 3]
        """
        image = tf.image.resize(image, [img_shape, img_shape])
        return tf.cast(image, tf.float32), label

In [29]: (train_data_a, test_data_a), ds_info = tfds.load(name="food101",
        split=["train", "validation"],
        shuffle_files=True,
        as_supervised=True,
        with_info=True)

data_augmentation = Sequential([
    preprocessing.Resizing(224, 224),
    preprocessing.RandomFlip("horizontal"),
    preprocessing.RandomRotation(0.2),
    preprocessing.RandomZoom(0.2)
], name="augmentation_layer")

train_data_a = train_data_a.map(map_func=preprocess_img, num_parallel_calls=tf.data.AUTOTUNE)
train_data_a = train_data_a.shuffle(buffer_size=500).batch(batch_size=32).prefetch(buffer_size=1)

test_data_a = test_data_a.map(map_func=preprocess_img, num_parallel_calls=tf.data.AUTOTUNE)
test_data_a = test_data_a.shuffle(buffer_size=500).batch(batch_size=32).prefetch(buffer_size=1)

## Feature Extraction Model (Data Aug)

In [30]: checkpoint_path = "model_checkpoints/model2.ckpt"
        model2_checkpoint = tf.keras.callbacks.ModelCheckpoint(checkpoint_path,
        monitor="val_accuracy",
        save_best_only=True,
        save_weights_only=True,
        verbose=0)

        mixed_precision.set_global_policy("mixed_float16")

        # #####
        # FEATURE EXTRACTION MODEL
        # #####
        input_shape=(224, 224, 3)
        base_model = tf.keras.applications.EfficientNetB0(include_top=False)
        base_model.trainable = False

        inputs = layers.Input(shape=input_shape, name="input_layer")
        x = data_augmentation(inputs)
        x = base_model(x, training=False)
        #x = base_model(inputs, training=False)
        x = layers.GlobalAveragePooling2D()(x)

```



```

2368/2368 [=====] - 174s 71ms/step - loss: 2.2034 - accuracy: 0.4682 -
Epoch 2/3
2368/2368 [=====] - 163s 68ms/step - loss: 1.7021 - accuracy: 0.5687 -
Epoch 3/3
2368/2368 [=====] - 163s 69ms/step - loss: 1.5807 - accuracy: 0.5953

```

```
In [32]: model2.save("drive/MyDrive/tf/models/FoodVision/featureExtractor_dataAug")
```

```
INFO:tensorflow:Assets written to: drive/MyDrive/tf/models/FoodVision/featureExtractor_dataAug
```

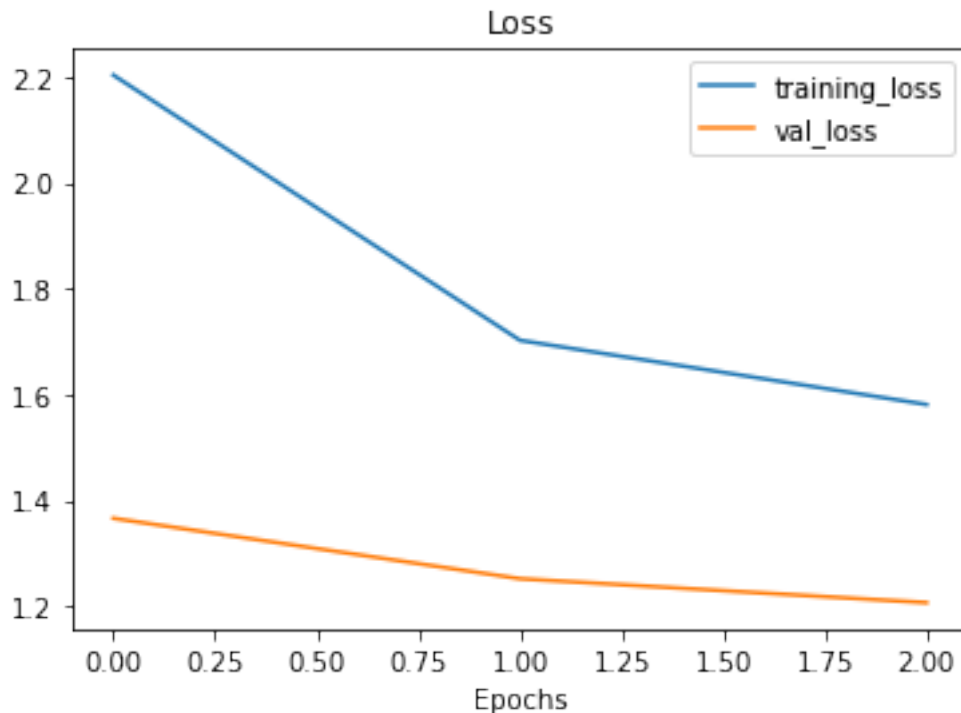
```

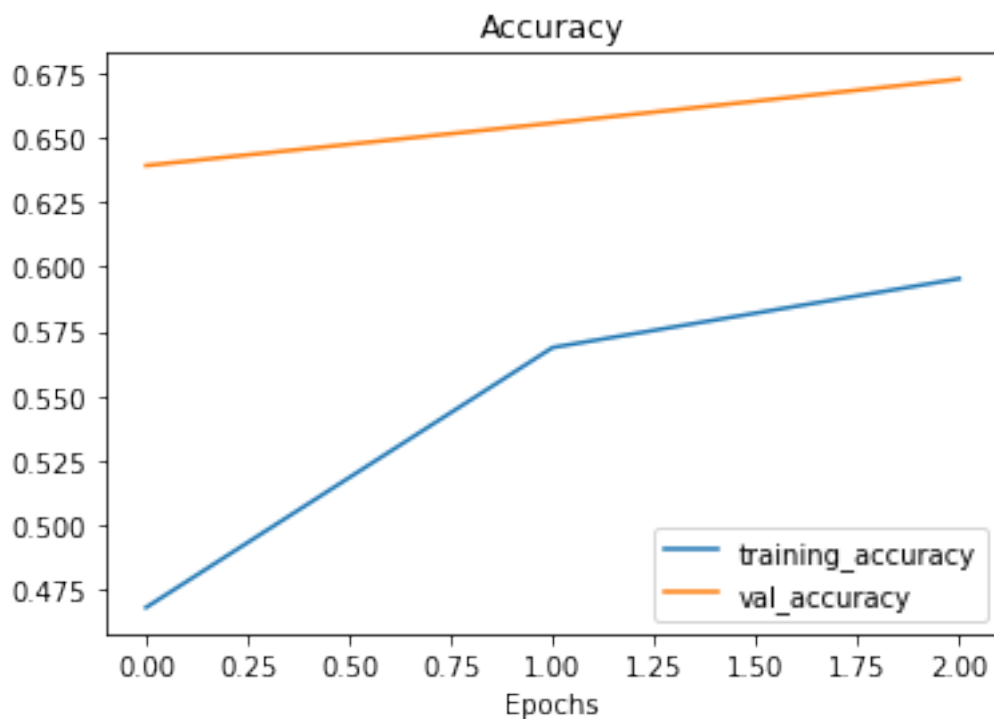
INFO:tensorflow:Assets written to: drive/MyDrive/tf/models/FoodVision/featureExtractor_dataAug
/usr/local/lib/python3.7/dist-packages/keras/utils/generic_utils.py:497: CustomMaskWarning: Custom
category=CustomMaskWarning)

```

```
In [33]: plot_loss_curves(history_feat_extr_data_aug)
        feat_extr_res_data_aug = model2.evaluate(test_data_a)
```

```
790/790 [=====] - 51s 63ms/step - loss: 1.2251 - accuracy: 0.6688
```





3.2 Fine Tune Model 1 (10 Layers for 3 more epochs, Data Aug)

```
In [34]: model3 = tf.keras.models.load_model("drive/MyDrive/tf/models/FoodVision/featureExtract")
        model3._name = "Model3_FineTuned_10Layers_SixEpochsTotal_DataAug"
```

```
WARNING:absl:Importing a function (__inference_block2a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block6a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_top_activation_layer_call_and_return_conditional
WARNING:absl:Importing a function (__inference_block1a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block4c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block4c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block7a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block7a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block1a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6c_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block4b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block7a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditional
WARNING:absl:Importing a function (__inference_block5c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_stem_activation_layer_call_and_return_conditional
```


WARNING:absl:Importing a function (__inference_block6d_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block4b_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block2a_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_block5b_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block4a_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_block7a_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_block2b_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_block6c_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditional
 WARNING:absl:Importing a function (__inference_block5c_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block5a_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_block4a_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block5a_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block2b_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_top_activation_layer_call_and_return_conditional
 WARNING:absl:Importing a function (__inference_block4c_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block5a_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_block3b_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block5a_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block3b_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_block2a_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_block6b_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditional
 WARNING:absl:Importing a function (__inference_block6a_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_block2b_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block6a_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block2a_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block3b_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_block3a_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_FeatureExtraction_ThreeEpochs_DataAug_layer_cal
 WARNING:absl:Importing a function (__inference_block4c_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_block2a_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_block2b_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_block6d_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_block6b_activation_layer_call_and_return_condit
 WARNING:absl:Importing a function (__inference_block6b_se_reduce_layer_call_and_return_conditi
 WARNING:absl:Importing a function (__inference_block5c_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_block6c_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_block6d_expand_activation_layer_call_and_return
 WARNING:absl:Importing a function (__inference_block6b_expand_activation_layer_call_and_return

Model: "Model3_FineTuned_10Layers_SixEpochsTotal_DataAug"

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	[(None, 224, 224, 3)]	0


```

augmentation_layer (Sequenti (None, 224, 224, 3)      0
-----
efficientnetb0 (Functional) (None, None, None, 1280) 4049571
-----
global_average_pooling2d_1 ( (None, 1280)            0
-----
dense_1 (Dense) (None, 101) 129381
-----
softmax_float32 (Activation) (None, 101) 0
=====
Total params: 4,178,952
Trainable params: 129,381
Non-trainable params: 4,049,571
-----

```

```

In [35]: model3.layers[2].trainable=True
         for layer in model3.layers[2].layers[:-10]:
             layer.trainable = False

```

```

In [54]: model3.summary()

```

```

Model: "Model3_FineTuned_10Layers_SixEpochsTotal_DataAug"

```

```

-----
Layer (type)                Output Shape              Param #
-----
input_layer (InputLayer)    [(None, 224, 224, 3)]    0
-----
augmentation_layer (Sequenti (None, 224, 224, 3)      0
-----
efficientnetb0 (Functional) (None, None, None, 1280) 4049571
-----
global_average_pooling2d_1 ( (None, 1280)            0
-----
dense_1 (Dense) (None, 101) 129381
-----
softmax_float32 (Activation) (None, 101) 0
=====
Total params: 4,178,952
Trainable params: 1,022,613
Non-trainable params: 3,156,339
-----

```

```

In [36]: checkpoint_path = "model_checkpoints/fine_tuned_data_aug.ckpt"
         model3_checkpoint = tf.keras.callbacks.ModelCheckpoint(checkpoint_path,
                                                                monitor="val_accuracy",
                                                                save_best_only=True,
                                                                save_weights_only=True,

```


3.3 Fine Tune Model 2 (All Layers for 3 More epochs, Augmented Data)

```
In [39]: model4 = tf.keras.models.load_model("drive/MyDrive/tf/models/FoodVision/featureExtract  
model4._name = "Model4_FineTuned_AllLayers_SixEpochsTotal_DataAug"
```

```
WARNING:absl:Importing a function (__inference_block2a_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block3a_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block6a_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_top_activation_layer_call_and_return_conditiona  
WARNING:absl:Importing a function (__inference_block1a_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block6b_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block4c_expand_activation_layer_call_and_return  
WARNING:absl:Importing a function (__inference_block4c_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block7a_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block7a_expand_activation_layer_call_and_return  
WARNING:absl:Importing a function (__inference_block1a_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block6a_expand_activation_layer_call_and_return  
WARNING:absl:Importing a function (__inference_block6c_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block3a_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block4b_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block7a_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona  
WARNING:absl:Importing a function (__inference_block5c_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_stem_activation_layer_call_and_return_conditiona  
WARNING:absl:Importing a function (__inference_block6c_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block4a_expand_activation_layer_call_and_return  
WARNING:absl:Importing a function (__inference_block5c_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block3b_expand_activation_layer_call_and_return  
WARNING:absl:Importing a function (__inference_block6d_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block3b_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block5c_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block5b_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block1a_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block4b_expand_activation_layer_call_and_return  
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona  
WARNING:absl:Importing a function (__inference_block2a_expand_activation_layer_call_and_return  
WARNING:absl:Importing a function (__inference_block4a_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona  
WARNING:absl:Importing a function (__inference_block6c_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block4b_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block3b_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block1a_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block3a_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block7a_expand_activation_layer_call_and_return  
WARNING:absl:Importing a function (__inference_block4a_se_reduce_layer_call_and_return_conditi  
WARNING:absl:Importing a function (__inference_block4c_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_block4b_activation_layer_call_and_return_condit  
WARNING:absl:Importing a function (__inference_FeatureExtraction_ThreeEpochs_DataAug_layer_cal  
WARNING:absl:Importing a function (__inference_block6a_activation_layer_call_and_return_condit
```

[illegible]

```

WARNING:absl:Importing a function (__inference_block6a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block2a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block3a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_FeatureExtraction_ThreeEpochs_DataAug_layer_cal
WARNING:absl:Importing a function (__inference_block4c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block2a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block2b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6d_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6b_activation_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block5c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6d_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6b_expand_activation_layer_call_and_return

```

```
In [42]: model4.layers[2].trainable=True
```

```
In [53]: model4.summary()
```

```
Model: "Model4_FineTuned_AllLayers_SixEpochsTotal_DataAug"
```

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	[(None, 224, 224, 3)]	0
augmentation_layer (Sequenti	(None, 224, 224, 3)	0
efficientnetb0 (Functional)	(None, None, None, 1280)	4049571
global_average_pooling2d_1 ((None, 1280)	0
dense_1 (Dense)	(None, 101)	129381
softmax_float32 (Activation)	(None, 101)	0
Total params: 4,178,952		
Trainable params: 4,136,929		
Non-trainable params: 42,023		

```

In [45]: checkpoint_path = "model_checkpoints/fine_tuned_AllLayers_data_aug.ckpt"
        model4_checkpoint = tf.keras.callbacks.ModelCheckpoint(checkpoint_path,
                                                                monitor="val_accuracy",
                                                                save_best_only=True,
                                                                save_weights_only=True,
                                                                verbose=0)

```


This has beaten the 77.4% accuracy benchmark set by the [DeepFood Paper](#). Now let's see if we can improve our model further by using an adaptive learning rate callback (while also adding an early stopping callback).

4 Fine Tune Model 3 (All Layers for 3 more Epochs, Augmented Data, Adaptive LR)

In [50]: *# model as prior this time with an adaptive learning rate callback*

```
model15 = tf.keras.models.load_model("drive/MyDrive/tf/models/FoodVision/featureExtract")
model15._name = "Model15_FineTuned_AllLayers_SixEpochsTotal_DataAug_DynLR"
```

```
WARNING:absl:Importing a function (__inference_block2a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block6a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_top_activation_layer_call_and_return_conditiona
WARNING:absl:Importing a function (__inference_block1a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block4c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block4c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block7a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block7a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block1a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6c_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block4b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block7a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona
WARNING:absl:Importing a function (__inference_block5c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_stem_activation_layer_call_and_return_conditiona
WARNING:absl:Importing a function (__inference_block6c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block4a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block5c_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6d_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block3b_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block5c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block5b_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block1a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block4b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona
WARNING:absl:Importing a function (__inference_block2a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block4a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona
WARNING:absl:Importing a function (__inference_block6c_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block4b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block3b_se_reduce_layer_call_and_return_conditi
```



```

WARNING:absl:Importing a function (__inference_block3b_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block5a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block2a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6b_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_efficientnetb0_layer_call_and_return_conditiona
WARNING:absl:Importing a function (__inference_block6a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block2b_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block6a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block2a_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block3b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block3a_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_FeatureExtraction_ThreeEpochs_DataAug_layer_cal
WARNING:absl:Importing a function (__inference_block4c_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block2a_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block2b_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6d_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block6b_activation_layer_call_and_return_condit
WARNING:absl:Importing a function (__inference_block6b_se_reduce_layer_call_and_return_conditi
WARNING:absl:Importing a function (__inference_block5c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6c_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6d_expand_activation_layer_call_and_return
WARNING:absl:Importing a function (__inference_block6b_expand_activation_layer_call_and_return

```

```

In [52]: model5.layers[2].trainable=True
         model5.summary()

         checkpoint_path = "model_checkpoints/fine_tuned_DynLR.ckpt"
         model5_checkpoint = tf.keras.callbacks.ModelCheckpoint(checkpoint_path,
                                                                monitor="val_accuracy",
                                                                save_best_only=True,
                                                                save_weights_only=True,
                                                                verbose=0)

         dynLR = tf.keras.callbacks.ReduceLROnPlateau(monitor="val_loss",
                                                       factor=0.2,
                                                       patience=2,
                                                       verbose=1,
                                                       min_lr=1e-7)

         early_stopping = tf.keras.callbacks.EarlyStopping(monitor="val_loss", patience=3)

         model5.compile(loss="sparse_categorical_crossentropy",
                       optimizer=tf.keras.optimizers.Adam(learning_rate=1e-4),
                       metrics=["accuracy"])

         fine_tune_epochs = 6 # (3+3)

```

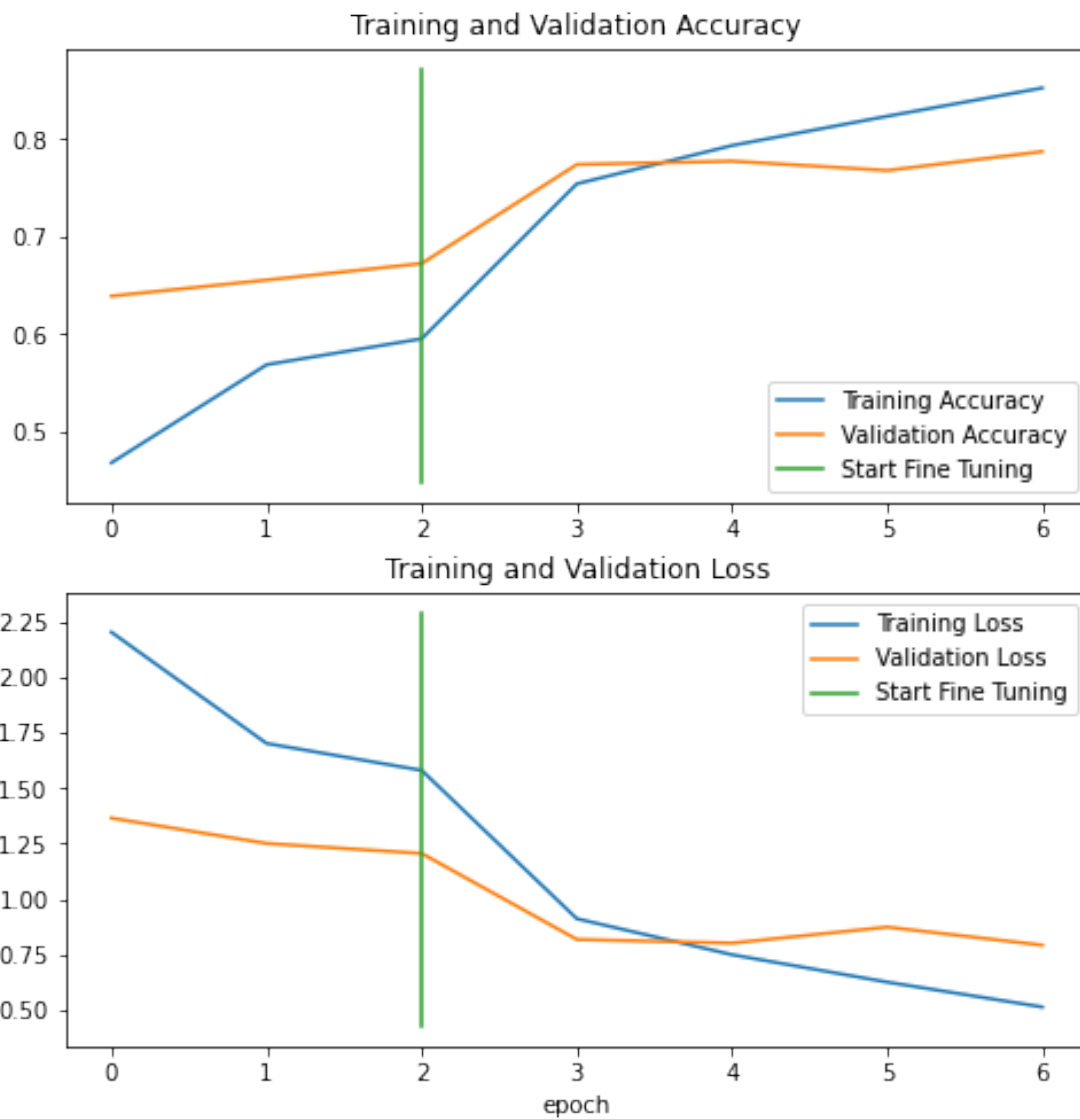
```
history_fine_tune_DynLR = model5.fit(train_data,
                                     epochs=fine_tune_epochs,
                                     steps_per_epoch=len(train_data),
                                     validation_data=test_data,
                                     validation_steps=int(0.15 * len(test_data)),
                                     initial_epoch=history_feat_extr_data_aug.epoch[-1],
                                     callbacks = [model4_checkpoint,
                                                create_tensorboard_callback(dir_name="tensorboard",
                                                                           experiment_name="EfficientNetB0_fine_tune_DynLR",
                                                                           dynLR=True,
                                                                           early_stopping=True),
                                                early_stopping],
                                     verbose=2)
```

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	[(None, 224, 224, 3)]	0
augmentation_layer (Sequenti	(None, 224, 224, 3)	0
efficientnetb0 (Functional)	(None, None, None, 1280)	4049571
global_average_pooling2d_1 ((None, 1280)	0
dense_1 (Dense)	(None, 101)	129381
softmax_float32 (Activation)	(None, 101)	0
Total params: 4,178,952		
Trainable params: 4,136,929		
Non-trainable params: 42,023		

```
/usr/local/lib/python3.7/dist-packages/keras/utils/generic_utils.py:497: CustomMaskWarning: Cu
category=CustomMaskWarning)
```

2368/2368 - 327s - loss: 0.5142 - accuracy: 0.8523 - val_loss: 0.7931 - val_accuracy: 0.7871

```
In [55]: compare_historys(history_feat_extr_data_aug, history_fine_tune_DynLR, 3)
         fine_tune_data_DynLR = model5.evaluate(test_data)
         print()
         print(f"Fine Tuned (w/ Dynamic LR) Accuracy: {100*fine_tune_data_DynLR[1]:.2f}%")
```



790/790 [=====] - 58s 71ms/step - loss: 0.8345 - accuracy: 0.7768

Fine Tuned (w/ Dynamic LR) Accuracy: 77.68%

5 Upload experiments to Tensorboard

```
In [ ]: !tensorboard dev upload --logdir ./tensorboard_callbacks \  
        --name "Fine-tuning EfficientNetB0 on Food101 Dataset" \  
        --description "Series of experiments building a fine-tuned feature extractor to be \  
        --one_shot
```

Link to [Tensorboard experiments](#)