Cats and Dogs Image Classification Using Deep Learning with Python

Python Implementation of CNN Model using Keras and TensorFlow is available on **GitHub**

1. CNN Neural Network Model Accuracy Results on Training and Validation Data

```
Epoch 193/200
60/60 [==============================] - 48s 792ms/step - loss: 0.2547 - accuracy: 0.8878 - val_loss: 0.2662 - val_accuracy: 0.8994
Epoch 194/200
60/60 [==============================] - 48s 793ms/step - loss: 0.2669 - accuracy: 0.8873 - val_loss: 0.2147 - val_accuracy: 0.9111
Epoch 195/200
60/60 [==============================] - 48s 791ms/step - loss: 0.2715 - accuracy: 0.8787 - val_loss: 0.2302 - val_accuracy: 0.9083
Epoch 196/200
60/60 [==============================] - 48s 794ms/step - loss: 0.2796 - accuracy: 0.8717 - val_loss: 0.2366 - val_accuracy: 0.9061
Epoch 197/200
60/60 [==============================] - 48s 791ms/step - loss: 0.2619 - accuracy: 0.8907 - val_loss: 0.2140 - val_accuracy: 0.9128
Epoch 198/200
60/60 [==============================] - 48s 792ms/step - loss: 0.2657 - accuracy: 0.8875 - val_loss: 0.2238 - val_accuracy: 0.9078
Epoch 199/200
60/60 [==============================] - 48s 791ms/step - loss: 0.2555 - accuracy: 0.8895 - val_loss: 0.2124 - val_accuracy: 0.9106
Epoch 200/200
60/60 [==============================] - 48s 790ms/step - loss: 0.2524 - accuracy: 0.8928 - val_loss: 0.2109 - val_accuracy: 0.9178
```

Results:

60/60 - 41s 681ms/step - loss: 0.2243 - accuracy: 0.9040

Training Accuracy: 90.40%

18/18 - 5s 273ms/step - loss: 0.2109 - accuracy: 0.9178

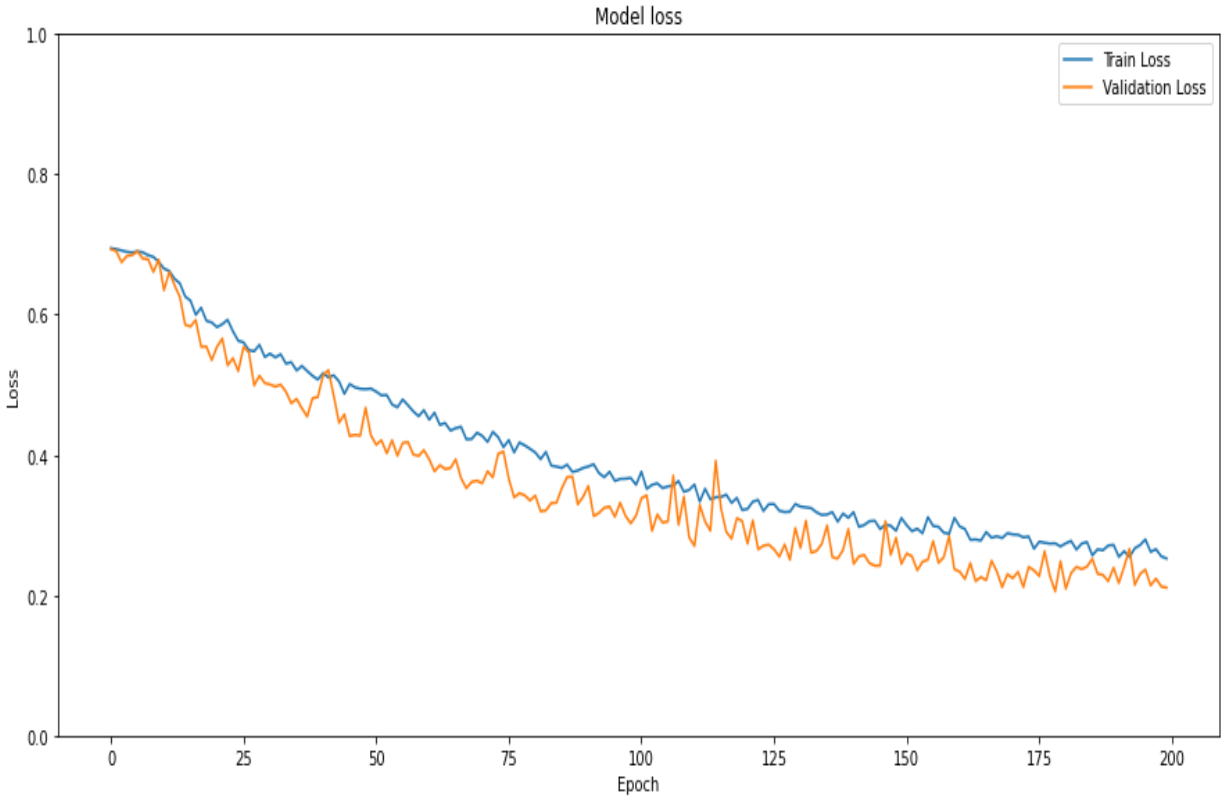
Validation Accuracy: 91.78%

- We can observe that the model accuracy of the **Training data** is 90.40% and the **validation data** is **91.78% (≈92%)** after 200 epochs.
- The validation accuracy is slightly greater than the training accuracy in almost every training. That means that our model doesn't overfit the training set.
- Hence, we can say that our Convolutional Neural Network (CNN) model is more **generalized** and prevented overfitting.

2. CNN Model Architecture Summary

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 148, 148, 128)     3584

 max_pooling2d (MaxPooling2D  (None, 74, 74, 128)      0
 )

 conv2d_1 (Conv2D)           (None, 72, 72, 64)        73792

 max_pooling2d_1 (MaxPooling  (None, 36, 36, 64)       0
 2D)

 conv2d_2 (Conv2D)           (None, 34, 34, 128)       73856

 max_pooling2d_2 (MaxPooling  (None, 17, 17, 128)      0
 2D)

 conv2d_3 (Conv2D)           (None, 15, 15, 128)       147584

 max_pooling2d_3 (MaxPooling  (None, 7, 7, 128)        0
 2D)

 dropout (Dropout)           (None, 7, 7, 128)         0

 flatten (Flatten)           (None, 6272)              0

 dense (Dense)               (None, 512)               3211776

 dense_1 (Dense)             (None, 2)                 1026

=================================================================
Total params: 3,511,618
Trainable params: 3,511,618
Non-trainable params: 0
```

3. CNN Model Loss on Train and Validation Data



4. CNN Model Accuracy on Train and Validation Data