

Project Development Phase
Model Performance Test

Date	17 November 2022
Team ID	PNT2022TMID18406
Project Name	Project - Digital Naturalist - AI Enabled tool for Biodiversity Researchers
Maximum Marks	10 Marks

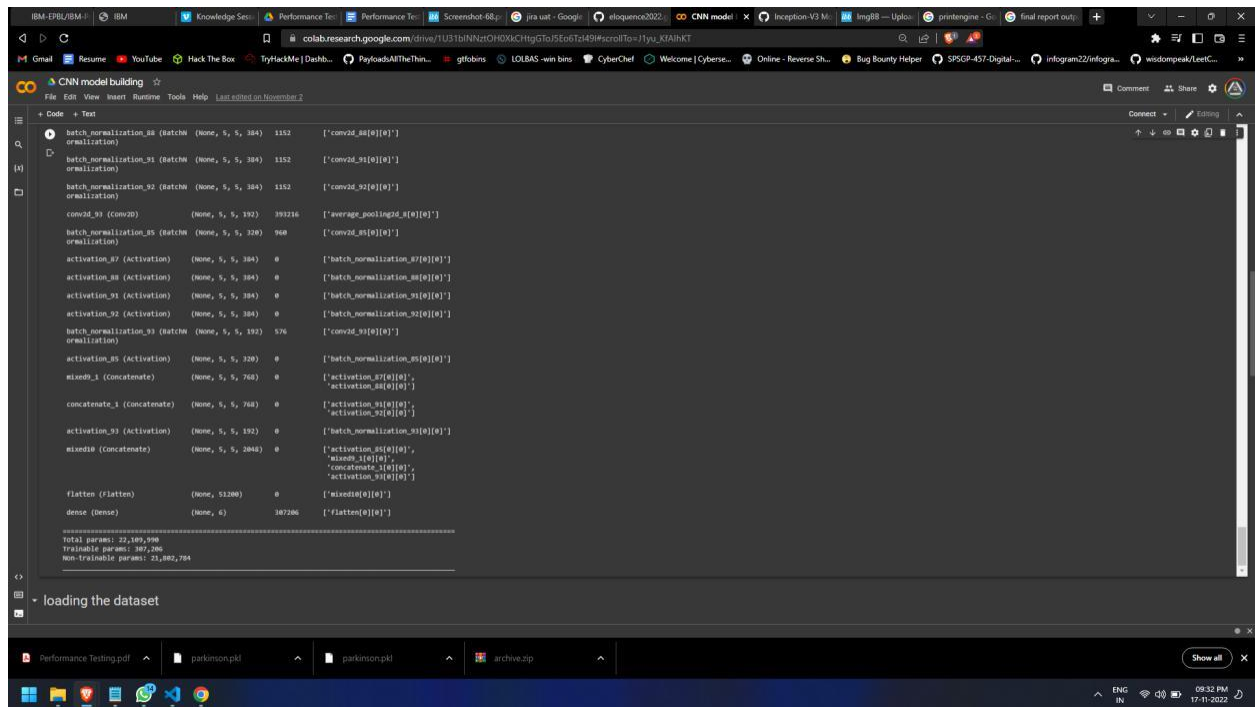
Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Total params: 22,109,990 Trainable params: 307,206 Non-trainable params: 21,802,784	Screenshot 1
2.	Accuracy	Training Accuracy - 92.8% Validation Accuracy - 85.6%	Screenshot 2

Screenshots - Please refer to the next page:

Screenshot 1 :

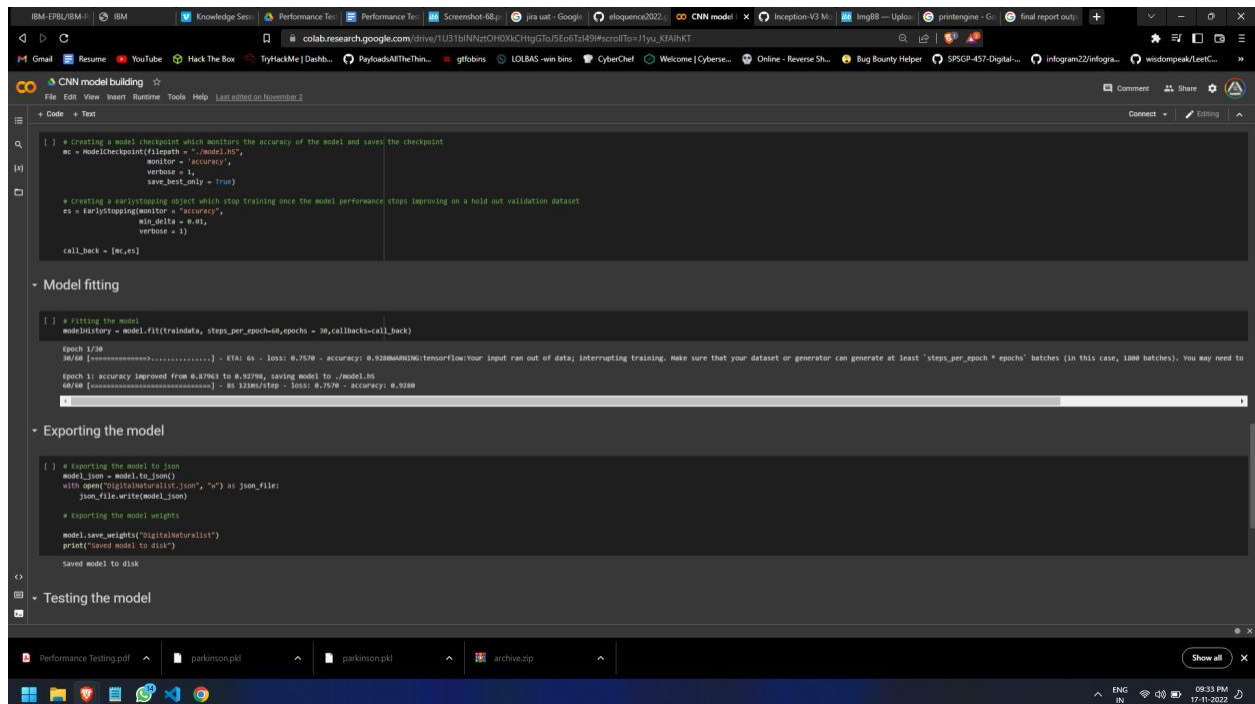


```
batch_normalization_B8 (Batch Normalization) (None, 5, 5, 384) 1352 ['convd_8[0][0]']
batch_normalization_B9 (Batch Normalization) (None, 5, 5, 384) 1352 ['convd_9[0][0]']
batch_normalization_B2 (Batch Normalization) (None, 5, 5, 384) 1352 ['convd_2[0][0]']
convd_3 (Conv2D) (None, 5, 5, 192) 193216 ['average_poolingd_1[0][0]']
batch_normalization_B5 (Batch Normalization) (None, 5, 5, 320) 968 ['convd_5[0][0]']
activation_87 (Activation) (None, 5, 5, 384) 0 ['batch_normalization_B7[0][0]']
activation_88 (Activation) (None, 5, 5, 384) 0 ['batch_normalization_B8[0][0]']
activation_91 (Activation) (None, 5, 5, 384) 0 ['batch_normalization_B9[0][0]']
activation_92 (Activation) (None, 5, 5, 384) 0 ['batch_normalization_B2[0][0]']
batch_normalization_B3 (Batch Normalization) (None, 5, 5, 192) 576 ['convd_3[0][0]']
activation_89 (Activation) (None, 5, 5, 192) 0 ['batch_normalization_B5[0][0]']
mixed_1 (Concatenate) (None, 5, 5, 768) 0 ['activation_87[0][0]', 'activation_88[0][0]']
concatenate_1 (Concatenate) (None, 5, 5, 768) 0 ['activation_91[0][0]', 'activation_92[0][0]']
activation_93 (Activation) (None, 5, 5, 192) 0 ['batch_normalization_B3[0][0]']
mixed18 (Concatenate) (None, 5, 5, 2048) 0 ['activation_89[0][0]', 'mixed_1[0][0]', 'concatenate_1[0][0]', 'activation_93[0][0]']
flatten (Flatten) (None, 51200) 0 ['mixed18[0][0]']
dense (Dense) (None, 4) 167204 ['flatten[0][0]']

Total params: 22,189,598
Trainable params: 187,198
Non-trainable params: 21,982,784
```

loading the dataset

Screenshot 2:



```
# Creating a model checkpoint which monitors the accuracy of the model and saves the checkpoint
mc = ModelCheckpoint(filepath = './model.h5',
                    monitor = 'accuracy',
                    verbose = 1,
                    save_best_only = True)

# creating a earlystopping object which stop training once the model performance stops improving on a hold out validation dataset
es = EarlyStopping(monitor = "accuracy",
                  min_delta = 0.01,
                  verbose = 1)

call_back = [mc, es]

# Fitting the model
modelHistory = model.fit(traindata, steps_per_epoch=40, epochs = 30, callbacks=call_back)

Epoch 1/30
36/68 [=====] - ETA: 6s - loss: 0.7578 - accuracy: 0.9286WARNING:tensorflow:Your input ran out of data; interrupting training. Make sure that your dataset or generator can generate at least 'steps_per_epoch * epochs' batches (in this case, 1800 batches). You may need to
Epoch 1: accuracy improved from 0.87963 to 0.92798, saving model to ./model.h5
68/68 [=====] - 85 121s/step - loss: 0.7578 - accuracy: 0.9286

# Exporting the model

# Exporting the model to json
model_json = model.to_json()
with open("digitalnaturalist.json", "w") as json_file:
    json_file.write(model_json)

# Exporting the model weights
model.save_weights("digitalnaturalist")
print("saved model to disk")

Saved model to disk
```

Model fitting

Exporting the model

Testing the model