

## Model Building

Date	12 October 2022
Team ID	PNT2022TMID35969
Project Name	AI-powered Nutrition Analyzer for Fitness Enthusiasts

### Import Model Building Libraries

```
import numpy as np
import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
from tensorflow.keras.layers import Dense, Flatten
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Dropout
from keras.preprocessing.image import ImageDataGenerator
```

### Initialize Model

```
model = Sequential()
```

### Adding CNN Layers

```
# First CNN layer
model.add(Conv2D(32,(3,3),input_shape=(64,64,3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
# Second CNN Layer
model.add(Conv2D(32,(3,3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
# Flatten layers
model.add(Flatten())
```

## Adding Dense Layers

```
model.add(Dense(units=128,activation='relu'))  
model.add(Dense(units=5, activation='softmax'))
```

## Model Summary

```
model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_1 (Conv2D)	(None, 29, 29, 32)	9248
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 32)	0
flatten (Flatten)	(None, 6272)	0
dense (Dense)	(None, 128)	802944
dense_1 (Dense)	(None, 5)	645

```
=====  
Total params: 813,733  
Trainable params: 813,733  
Non-trainable params: 0  
=====
```

## Configure Learning Process

```
# Compile the model

model.compile(optimizer='adam', loss='sparse_categorical_crossentropy',
metrics=['accuracy'])

train_datagen = ImageDataGenerator(rescale=1./225, shear_range=0.2,
zoom_range=0.2, horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1./225)

x_train = train_datagen.flow_from_directory
(r'C:\Users\Deepshika\TRAIN_SET',target_size=(64,64),batch_size=5,color
_mode='rgb',class_mode='sparse')

x_test = train_datagen.flow_from_directory
(r'C:\Users\Deepshika\TEST_SET',target_size=(64,64),batch_size=5,color_
mode='rgb',class_mode='sparse')
```

```
Found 2626 images belonging to 5 classes.
Found 1055 images belonging to 5 classes.
```

## Train Model

```
model.fit_generator(generator = x_train, steps_per_epoch =
len(x_train), epochs = 20, validation_data = x_test, validation_steps =
len(x_test))
```

```
Epoch 1/20
526/526 [=====] - 23s 41ms/step - loss: 0.1654 - accuracy: 0.9379 - val_loss: 0.0060 - val_accuracy:
1.0000
Epoch 2/20
526/526 [=====] - 21s 41ms/step - loss: 5.6409e-04 - accuracy: 1.0000 - val_loss: 0.0019 - val_accu
racy: 1.0000
Epoch 3/20
526/526 [=====] - 21s 40ms/step - loss: 7.6299e-05 - accuracy: 1.0000 - val_loss: 0.0025 - val_accu
racy: 1.0000
Epoch 4/20
526/526 [=====] - 18s 34ms/step - loss: 5.2999e-05 - accuracy: 1.0000 - val_loss: 7.2984e-04 - val_acc
uracy: 1.0000
Epoch 5/20
526/526 [=====] - 20s 38ms/step - loss: 2.2845e-05 - accuracy: 1.0000 - val_loss: 0.0013 - val_accu
racy: 1.0000
Epoch 6/20
526/526 [=====] - 26s 49ms/step - loss: 1.5693e-05 - accuracy: 1.0000 - val_loss: 9.2336e-04 - val_acc
uracy: 1.0000
Epoch 7/20
526/526 [=====] - 24s 45ms/step - loss: 8.5315e-06 - accuracy: 1.0000 - val_loss: 0.0010 - val_accu
racy: 1.0000
Epoch 8/20
526/526 [=====] - 22s 42ms/step - loss: 9.2632e-06 - accuracy: 1.0000 - val_loss: 0.0014 - val_accu
racy: 1.0000
Epoch 9/20
526/526 [=====] - 22s 42ms/step - loss: 0.0772 - accuracy: 0.9787 - val_loss: 0.2698 - val_accuracy:
0.9545
Epoch 10/20
526/526 [=====] - 23s 44ms/step - loss: 7.7843e-04 - accuracy: 1.0000 - val_loss: 0.2088 - val_accu
racy: 0.9725
```

```
Epoch 11/20
526/526 [=====] - 24s 45ms/step - loss: 3.7376e-05 - accuracy: 1.0000 - val_loss: 0.1913 - val_accuracy: 0.9763
Epoch 12/20
526/526 [=====] - 24s 46ms/step - loss: 2.2467e-05 - accuracy: 1.0000 - val_loss: 0.1838 - val_accuracy: 0.9735
Epoch 13/20
526/526 [=====] - 23s 43ms/step - loss: 1.5592e-05 - accuracy: 1.0000 - val_loss: 0.1904 - val_accuracy: 0.9725
Epoch 14/20
526/526 [=====] - 24s 47ms/step - loss: 2.7025e-05 - accuracy: 1.0000 - val_loss: 0.1724 - val_accuracy: 0.9773
Epoch 15/20
526/526 [=====] - 24s 46ms/step - loss: 9.2424e-06 - accuracy: 1.0000 - val_loss: 0.1554 - val_accuracy: 0.9763
Epoch 16/20
526/526 [=====] - 24s 46ms/step - loss: 5.2953e-06 - accuracy: 1.0000 - val_loss: 0.1740 - val_accuracy: 0.9763
Epoch 17/20
526/526 [=====] - 22s 42ms/step - loss: 3.6720e-06 - accuracy: 1.0000 - val_loss: 0.1921 - val_accuracy: 0.9763
Epoch 18/20
526/526 [=====] - 24s 46ms/step - loss: 4.7641e-06 - accuracy: 1.0000 - val_loss: 0.1761 - val_accuracy: 0.9754
Epoch 19/20
526/526 [=====] - 21s 41ms/step - loss: 3.4135e-06 - accuracy: 1.0000 - val_loss: 0.1599 - val_accuracy: 0.9773
Epoch 20/20
526/526 [=====] - 24s 45ms/step - loss: 1.6362e-06 - accuracy: 1.0000 - val_loss: 0.1852 - val_accuracy: 0.9763
<keras.callbacks.History at 0x25608bccd00>
```

## Save Model

```
model.save('nutrition.h5')
```

## Test Model

```
from tensorflow.keras.models import load_model
from keras.preprocessing import image

Model = load_model("nutrition.h5")

import keras

img = keras.utils.load_img(r"C:\Users\Deepshika\TEST_SET\image4.jpg",
                             grayscale=False, target_size=(64,64))

x = keras.utils.img_to_array(img)

x = np.expand_dims(x,axis = 0)

pred = Model.predict(x)

pred
```

```
1/1 [=====] - 0s 82ms/step  
array([[1., 0., 0., 0., 0.]], dtype=float32)
```

```
index = ['APPLES', 'BANANA', 'ORANGE', 'PINEAPPLE', 'WATERMELON']  
res = ""  
for i in range(len(pred[0])):  
    if pred[0][i] == 1:  
        res = index[i]  
res
```

res

'APPLES'

