# MODEL BUILDIND

|  |  |
| --- | --- |
| DATE | 03/11/22 |
| TEAM ID | PNT2022TMID36873 |
| PROJEET NAME | AI-powered Nutrition Analyzer for Fitness Enthusiasts |
| MARK | 6 |

# Model Building

## Importing The Model Building Libraries

import numpy as np import tensorflow as tf

from tensorflow.keras.models import Sequentialfrom tensorflow.keras import layers

from tensorflow.keras.layers import Dense,Flatten

from tensorflow.keras.layers import Conv2D,MaxPooling2D,Dropout

## Initializing The Model

model = Sequential()

## Adding CNN Layers

# Initializing the CNN classifier = Sequential()

# First convolution layer and pooling

classifier.add(Conv2D(32, (3, 3), input\_shape=(64, 64, 3), activation='relu'))

classifier.add(MaxPooling2D(pool\_size=(2, 2)))

# Second convolution layer and pooling classifier.add(Conv2D(32, (3, 3), activation='relu'))

# input\_shape is going to be the pooled feature maps from the previous convolution layerclassifier.add(MaxPooling2D(pool\_size=(2, 2)))

# Flattening the layers classifier.add(Flatten())

## Adding Dense Layers

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

#summary of our model classifi er.summa ry()

Model: "sequential\_1"

Layer (type) Output Shape Param #

=================================================================

conv2d (Conv2D)

max\_pooling2d (MaxPooling2D

)

conv2d\_1 (Conv2D)

max\_pooling2d\_1 (MaxPooling 2D)

flatten (Flatten) dense (Dense)

dense\_1 (Dense)

(None, 62, 62, 32)

(None, 31, 31, 32)

896

0

(None, 29, 29, 32)

(None, 14, 14, 32)

9248

0

(None, 6272)

(None, 128)

(None, 5)

0

802944

645

=================================================================

Total params: 813,733

Trainable params: 813,733

Non-trainable params: 0

## Configure The Learning Process

# Compiling the CNN

# categorical\_crossentropy for more than 2 classifier.compile(optimizer='adam', loss='sparse\_categorical\_crossentropy', metrics=['acc

## Train The Model

#Fitting the model classifier.fit\_generator(generator=x\_train,steps\_per\_epoch = len(x\_train),epochs=20, valid

Epoch 1/20

/usr/local/lib/python3.7/dist-packages/ipykernel\_launcher.py:2: UserWarning: `Model.

824/824

[==============================]

Epoch 2/20 824/824

[==============================]

Epoch 3/20 824/824

[==============================]

Epoch 4/20 824/824

[==============================]

Epoch 5/20

- 16ms/step - 0.6172 -

21s loss: accuracy:

-

13s

15ms/step

-

loss:

0.4115

-

accuracy:

-

13s

16ms/step

-

loss:

0.3766

-

accuracy:

-

13s

16ms/step

-

loss:

0.3484

-

accuracy:

824/824

[==============================]

Epoch 6/20 824/824

[==============================]

Epoch 7/20 824/824

[==============================]

Epoch 8/20 824/824

[==============================]

Epoch 9/20 824/824

[==============================]

Epoch 10/20 824/824

[==============================]

Epoch 11/20 824/824

[==============================]

Epoch 12/20 824/824

[==============================]

Epoch 13/20 824/824

[==============================]

Epoch 14/20

- 16ms/step - 0.3243 -

13s loss: accuracy:

-

13s

16ms/step

-

loss:

0.3240

-

accuracy:

-

13s

16ms/step

-

loss:

0.2887

-

accuracy:

-

13s

16ms/step

-

loss:

0.2728

-

accuracy:

-

13s

16ms/step

-

loss:

0.2717

-

accuracy:

-

14s

17ms/step

-

loss:

0.2365

-

accuracy:

-

13s

15ms/step

-

loss:

0.2301

-

accuracy:

-

13s

15ms/step

-

loss:

0.2083

-

accuracy:

-

13s

15ms/step

-

loss:

0.2049

-

accuracy:

824/824 [==============================]

Epoch 15/20 824/824

[==============================]

Epoch 16/20 824/824

[==============================]

Epoch 17/20 824/824

[==============================]

Epoch 18/20 824/824

[==============================]

Epoch 19/20 824/824

[==============================]

Epoch 20/20 824/824

[==============================]

-

12s

-

13s

-

13s

-

13s

-

13s

-

13s

-

12s

15ms/step

15ms/step

15ms/step

15ms/step 15ms/step

15ms/step

15ms/step

-

loss:

-

loss:

-

loss:

-

loss:

-

loss:

-

loss:

-

loss:

0.1930

0.1807

0.1712

0.1599

0.1619

0.1505

0.1211

-

accuracy:

-

accuracy:

-

accuracy:

-

accuracy:

-

accuracy:

-

accuracy:

-

accuracy:

<keras.callbacks.History at 0x7fd655833d90>

## Saving The Model

classifier.save('nutrition.h5')

## Testing The Model

#Predict the results

from tensorflow.keras.models import load\_modelfrom keras.preprocessing import image model = load\_model("nutrition.h5")

from tensorflow.keras.utils import img\_to\_array#loading of the image img =

load\_img(r'/content/Sample\_Images/Test\_Image1.jpg',grayscale=False,target\_siz e= (64,#image to array

x = img\_to\_a rray(img

)

#changin g the shape

x = np.expand\_dims(x,axis = 0)

predict\_x=model.predict(x) classes\_x=np.argmax(predi ct\_x,axis=-1)classes\_x

1/1 [==============================] - 0s 18ms/step

array([0])

index=['APPLES', 'BANANA', 'ORANGE','PINEAPPLE','WATERMELON']

result=str(index[c lasses\_x[0]]) result

'APPLES'

[Colab HYPERLINK](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links)

["https://colab.research.google.com/si](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [gnup?utm\_source=footer&utm\_med](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [ium=link&utm\_campaign=footer\_li](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links)

[nks" HYPERLINK](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links)

["https://colab.research.google.co](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [m/signup?utm\_source=footer&u](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [tm\_medium=link&utm\_campaig](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [n=footer\_links"paid](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [HYPERLINK](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links)

["https://colab.research.google.com/s](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [ignup?utm\_source=footer&utm\_me](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [dium=link&utm\_campaign=footer\_](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [links" HYPERLINK](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) ["https://colab.research.google.co](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [m/signup?utm\_source=footer&u](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [tm\_medium=link&utm\_campaig](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) [n=footer\_links"products](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) - [Cancel HYPERLINK](https://colab.research.google.com/cancel-subscription) ["https://colab.research.google.com/c](https://colab.research.google.com/cancel-subscription) [ancel-subscription" HYPERLINK](https://colab.research.google.com/cancel-subscription) ["https://colab.research.google.co](https://colab.research.google.com/cancel-subscription) [m/cancel-subscription"contracts](https://colab.research.google.com/cancel-subscription) [HYPERLINK](https://colab.research.google.com/cancel-subscription)

["https://colab.research.google.com/c](https://colab.research.google.com/cancel-subscription) [ancel-subscription" HYPERLINK](https://colab.research.google.com/cancel-subscription) ["https://colab.research.google.co](https://colab.research.google.com/cancel-subscription) [m/cancel-subscription"here](https://colab.research.google.com/cancel-subscription)