# Unzip Dataset

```
!unzip '/content/drive/MyDrive/Colab Notebooks/dataset/Flowers-Dataset.zip
```

```
  inflating: flowers/tulip/8712270243_8512cf4fbd.jpg
  inflating: flowers/tulip/8712270665_57b5bda0a2_n.jpg
  inflating: flowers/tulip/8712282563_3819afb7bc.jpg
  inflating: flowers/tulip/8713357842_9964a93473_n.jpg
  inflating: flowers/tulip/8713387500_6a9138b41b_n.jpg
  inflating: flowers/tulip/8713388322_e5ae26263b_n.jpg
  inflating: flowers/tulip/8713389178_66bceb71a8_n.jpg
  inflating: flowers/tulip/8713390684_041148dd3e_n.jpg
  inflating: flowers/tulip/8713391394_4b679ea1e3_n.jpg
  inflating: flowers/tulip/8713392604_90631fb809_n.jpg
  inflating: flowers/tulip/8713394070_b24561b0a9.jpg
  inflating: flowers/tulip/8713396140_5af8136136.jpg
  inflating: flowers/tulip/8713397358_0505cc0176_n.jpg
  inflating: flowers/tulip/8713397694_bcbcbba2c2_n.jpg
  inflating: flowers/tulip/8713398114_bc96f1b624_n.jpg
  inflating: flowers/tulip/8713398614_88202e452e_n.jpg
  inflating: flowers/tulip/8713398906_28e59a225a_n.jpg
  inflating: flowers/tulip/8713407768_f880df361f.jpg
  inflating: flowers/tulip/8717900362_2aa508e9e5.jpg
  inflating: flowers/tulip/8722514702_7ecc68691c.jpg
  inflating: flowers/tulip/8723767533_9145dec4bd_n.jpg
  inflating: flowers/tulip/8729501081_b993185542_m.jpg
  inflating: flowers/tulip/8733586143_3139db6e9e_n.jpg
  inflating: flowers/tulip/8748266132_5298a91dcf_n.jpg
  inflating: flowers/tulip/8750288831_5e49a9f29b.jpg
  inflating: flowers/tulip/8757486380_90952c5377.jpg
  inflating: flowers/tulip/8758464923_75a5ffe320_n.jpg
  inflating: flowers/tulip/8758519201_16e8d2d781_n.jpg
  inflating: flowers/tulip/8759594528_2534c0ec65_n.jpg
  inflating: flowers/tulip/8759597778_7fca5d434b_n.jpg
  inflating: flowers/tulip/8759601388_36e2a50d98_n.jpg
  inflating: flowers/tulip/8759606166_8e475013fa_n.jpg
  inflating: flowers/tulip/8759618746_f5e39fdbf8_n.jpg
  inflating: flowers/tulip/8762189906_8223cef62f.jpg
  inflating: flowers/tulip/8762193202_0fbf2f6a81.jpg
  inflating: flowers/tulip/8768645961_8f1e097170_n.jpg
  inflating: flowers/tulip/8817622133_a42bb90e38_n.jpg
  inflating: flowers/tulip/8838347159_746d14e6c1_m.jpg
  inflating: flowers/tulip/8838354855_c474fc66a3_m.jpg
  inflating: flowers/tulip/8838914676_8ef4db7f50_n.jpg
  inflating: flowers/tulip/8838975946_f54194894e_m.jpg
  inflating: flowers/tulip/8838983024_5c1a767878_n.jpg
  inflating: flowers/tulip/8892851067_79242a7362_n.jpg
  inflating: flowers/tulip/8904780994_8867d64155_n.jpg
  inflating: flowers/tulip/8908062479_449200a1b4.jpg
  inflating: flowers/tulip/8908097235_c3e746d36e_n.jpg
  inflating: flowers/tulip/9019694597_2d3bbedb17.jpg
  inflating: flowers/tulip/9030467406_05e93ff171_n.jpg
  inflating: flowers/tulip/9048307967_40a164a459_m.jpg
  inflating: flowers/tulip/924782410_94ed7913ca_m.jpg
  inflating: flowers/tulip/9378657435_89fabf13c9_n.jpg
  inflating: flowers/tulip/9444202147_405290415b_n.jpg
```

```
  inflating: flowers/tulip/9446982168_06c4d71da3_n.jpg
  inflating: flowers/tulip/9831362123_5aac525a99_n.jpg
  inflating: flowers/tulip/9870557734_88eb3b9e3b_n.jpg
  inflating: flowers/tulip/9947374414_fdf1d0861c_n.jpg
  inflating: flowers/tulip/9947385346_3a8cacea02_n.jpg
  inflating: flowers/tulip/9976515506_d496c5e72c.jpg
```

# Image Augmentation

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(rescale=1./255, zoom_range=0.6, horizontal_flip

x_train = train_datagen.flow_from_directory('/content/flowers', target_size=(64,64
```

```
Found 4317 images belonging to 5 classes.
```

# CNN Model Training

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D, MaxPooling2D, Flatten, Dense

model = Sequential()
model.add(Convolution2D(32,(3,3), activation='relu', input_shape=(64,64,3)))
model.add(MaxPooling2D(pool_size=(3,3)))
model.add(Flatten())
model.add(Dense(128,activation='relu'))
model.add(Dense(256,activation='relu'))
model.add(Dense(512,activation='relu'))
model.add(Dense(256,activation='relu'))
model.add(Dense(5,activation='softmax'))

model.compile(optimizer='adam',loss='categorical_crossentropy', metrics=['accuracy

model.fit(x_train,
          steps_per_epoch=len(x_train),
          epochs=10)
```

```
Epoch 1/10
44/44 [==============================] - 14s 315ms/step - loss: 1.1122 - accu
Epoch 2/10
44/44 [==============================] - 14s 315ms/step - loss: 1.0103 - accu
Epoch 3/10
44/44 [==============================] - 15s 350ms/step - loss: 0.9776 - accu
Epoch 4/10
```

```
44/44 [==============================] - 14s 315ms/step - loss: 0.9232 - accu
Epoch 5/10
44/44 [==============================] - 14s 312ms/step - loss: 0.8961 - accu
Epoch 6/10
44/44 [==============================] - 14s 315ms/step - loss: 0.8982 - accu
Epoch 7/10
44/44 [==============================] - 15s 345ms/step - loss: 0.8423 - accu
Epoch 8/10
44/44 [==============================] - 14s 317ms/step - loss: 0.8268 - accu
Epoch 9/10
44/44 [==============================] - 15s 349ms/step - loss: 0.8152 - accu
Epoch 10/10
44/44 [==============================] - 14s 315ms/step - loss: 0.7984 - accu
<keras.callbacks.History at 0x7fac00093c10>
```

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

# Testing Model

```
import numpy as np
from tensorflow.keras.preprocessing import image



image_path·=·f'/content/4393191.jpg'
img·=·image.load_img(image_path,target_size=(64,64))
x·=·image.img_to_array(img)
x·=·np.expand_dims(x,axis=0)
pred·=·np.argmax(model.predict(x))
#·print(pred,·model.predict(x))
op·=·['daisy','dandelion','rose','sunflower','tulip']
print(op[pred])
```

```
1/1 [==============================] - 0s 122ms/step
rose
```

# Tuning Model

```
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau


early_stop = EarlyStopping(monitor='accuracy', patience=8)

lr = ReduceLROnPlateau(monitor='accuracy', factor=0.2, patience=6, min_lr=0.1)

callbacks = [early_stop,lr]


model.fit(x_train,
          steps_per_epoch=len(x_train),
```

```
          epochs=20,
          callbacks=callbacks,)
```

```
Epoch 1/20
44/44 [==============================] - 16s 354ms/step - loss: 0.8074 - accu
Epoch 2/20
44/44 [==============================] - 14s 323ms/step - loss: 0.7641 - accu
Epoch 3/20
44/44 [==============================] - 14s 320ms/step - loss: 0.7510 - accu
Epoch 4/20
44/44 [==============================] - 14s 323ms/step - loss: 0.7467 - accu
Epoch 5/20
44/44 [==============================] - 15s 349ms/step - loss: 0.7592 - accu
Epoch 6/20
44/44 [==============================] - 14s 324ms/step - loss: 0.7107 - accu
Epoch 7/20
44/44 [==============================] - 14s 320ms/step - loss: 0.7031 - accu
Epoch 8/20
44/44 [==============================] - 14s 324ms/step - loss: 0.6917 - accu
Epoch 9/20
44/44 [==============================] - 16s 354ms/step - loss: 0.6940 - accu
Epoch 10/20
44/44 [==============================] - 16s 357ms/step - loss: 0.6928 - accu
Epoch 11/20
44/44 [==============================] - 14s 328ms/step - loss: 0.6670 - accu
Epoch 12/20
44/44 [==============================] - 14s 325ms/step - loss: 0.6579 - accu
Epoch 13/20
44/44 [==============================] - 16s 354ms/step - loss: 0.6483 - accu
Epoch 14/20
44/44 [==============================] - 14s 327ms/step - loss: 0.6281 - accu
Epoch 15/20
44/44 [==============================] - 14s 324ms/step - loss: 0.6400 - accu
Epoch 16/20
44/44 [==============================] - 14s 326ms/step - loss: 0.6317 - accu
Epoch 17/20
44/44 [==============================] - 16s 363ms/step - loss: 0.6001 - accu
Epoch 18/20
44/44 [==============================] - 14s 324ms/step - loss: 0.6074 - accu
Epoch 19/20
44/44 [==============================] - 14s 325ms/step - loss: 0.6204 - accu
Epoch 20/20
44/44 [==============================] - 15s 333ms/step - loss: 0.5836 - accu
<keras.callbacks.History at 0x7faba6647f90>
```

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

```
image_path = f'/content/4393191.jpg'
img = image.load_img(image_path,target_size=(64,64))
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x))
# print(pred, model.predict(x))
op = ['daisy','dandelion','rose','sunflower','tulip']
print(op[pred])
```

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