import tensorflow as tf

from tensorflow import keras

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Conv2D,MaxPooling2D ,Dense,Flatten

import numpy as np

import matplotlib.pyplot as plt

import random

#Load Dataset

X\_train=np.loadtxt('C:/Users/priti/Desktop/Assignment\_3/input.csv', delimiter =',')

Y\_train=np.loadtxt('C:/Users/priti/Desktop/Assignment\_3/labels.csv',delimiter =',')

X\_test=np.loadtxt('C:/Users/priti/Desktop/Assignment\_3/input\_test.csv',delimiter =',')

Y\_test=np.loadtxt('C:/Users/priti/Desktop/Assignment\_3/labels\_test.csv', delimiter =',')

#Reshaping appropriate size..removing comma

X\_train = X\_train.reshape(len(X\_train),100,100,3)

Y\_train= Y\_train.reshape(len(Y\_train),1)

X\_test = X\_test.reshape(len(X\_test),100,100,3)

Y\_test= Y\_test.reshape(len(Y\_test),1)

print("Shape of X\_train:", X\_train.shape)

print("Shape of Y\_train:", Y\_train.shape)

print("Shape of X\_test:", X\_test.shape)

print("Shape of Y\_test:", Y\_test.shape)

X\_train[1]

#Feature Scaling rescale to 0 and 1

X\_train=X\_train/255.0

X\_test = X\_test/255.0

X\_train[1]

#display image

idx=random.randint(0,len(X\_train))

plt.imshow(X\_train[idx,:])

plt.show()

#2 model buliding

model= Sequential()

model.add(Conv2D(32, (3,3), activation ='relu', input\_shape = (100,100,3)))

model.add(MaxPooling2D((2,2)))

model.add(Conv2D(32,(3,3),activation='relu'))

model.add(MaxPooling2D((2,2)))

model.add(Flatten())

model.add(Dense(64,activation = 'relu'))

model.add(Dense(1,activation ='sigmoid'))

model.compile(loss='binary\_crossentropy',optimizer ='adam', metrics =['accuracy'])

model.fit(X\_train,Y\_train,epochs=7, batch\_size =64 )

model.evaluate(X\_test,Y\_test)

#Making Predictions

idx2 = random.randint(0,len(Y\_test))

plt.imshow(X\_test[idx2,:])

plt.show()

y\_pred = model.predict(X\_test[idx2,:].reshape(1,100,100,3))

print(y\_pred)

y\_pred = y\_pred>0.5

if(y\_pred == 0):

pred ='dog'

else:

pred='cat'

print("Our model says it is a:", pred)