Deep Learning

Practical: 09

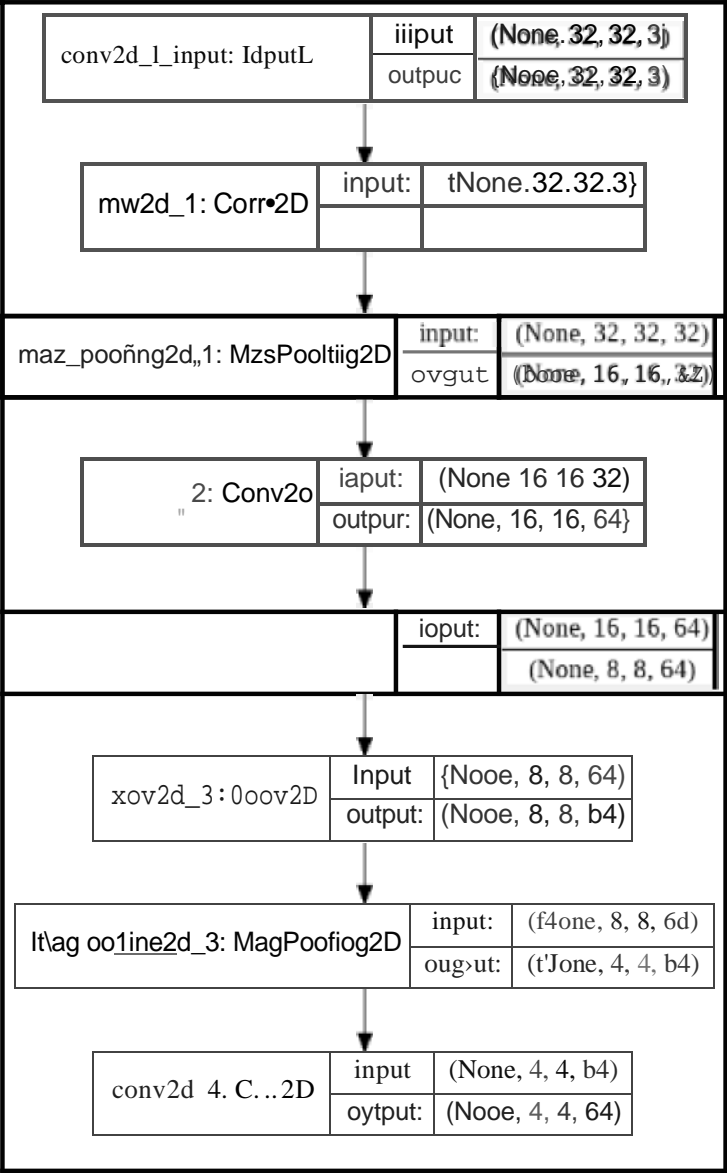| Name of the student	Vishesh Gupta
Roll No.	J018
Class	B.Tech Data Science 3rd Year

Aim:

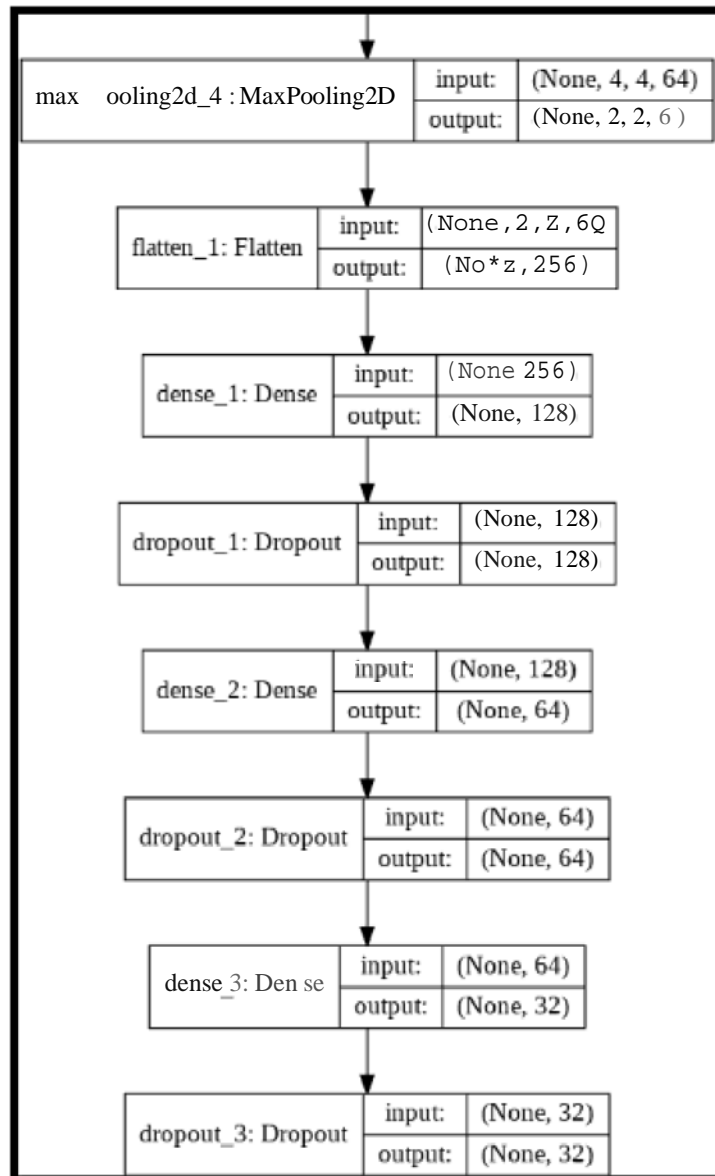Identify and download 10 types of images from imagenet and build a CNN model to classify them.

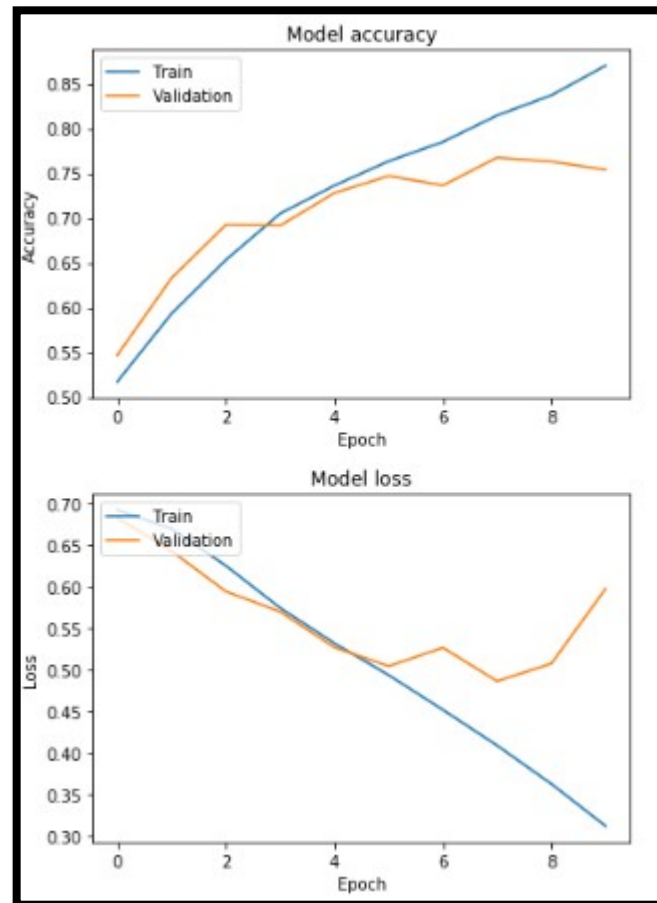The CNN model includes filters as 32, kernel_size as (3,3) and activation function used in reLu.

The optimizer used in this model is adadelta and the loss function is binary_crossentropy.

Accuracy: 76.3%.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 3Z, 32, 32)	896
maxpoolingZd_1 (BaxPooling2	(None, 16, 16, 32)	0
conv2d_Z (Conv2D)	(None, l6, 16, 64)	18496
max oo1ing2d_2 laxPooling2	(None, B 8, 64)	0
conv2d_3 (Conw2D)	(None, B 8, 64-)	36928
maxpoolingZd_3 (HaxPooling2	(None, 4, 4, d4)	0
conv2d_4 (Conw2D)	(None, 4, 4, dv)	36928
maxpooling2b_4 (HaxPooling2	(None, 2, 2, d4)	0
flatten_l (Flatten)	(None, 256)	0
dense_l (Dense)	(done, 128)	32896
dropout_l (Dropout)	(done, 12B)	0
dense_2 (Dense)	(done, b4)	8256
dropout_2 (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 32)	2888
dropout_3 (Dropout)	(None, 32)	0
dense_4 (Dense)	(None, 1)	33

Total params : 136, 513
Tra1nab1e params : 136, 513
Non- tra1nable params : 0

conv2d_l_input: IdputL	iiiput	(None, 32, 32, 3)
	outpuc	(Nooe, 32, 32, 3)

mw2d_1: Corr•2D	input:	tNone.32.32.3}

maz_pooñng2d„1: MzsPooltiig2D	input:	(None, 32, 32, 32)
	ovgut	(Nooe, 16, 16, 32)

2: Conv2o	iaput:	(None 16 16 32)
	outpur:	(None, 16, 16, 64}

	ioput:	(None, 16, 16, 64)
		(None, 8, 8, 64)

xov2d_3:0oov2D	Input	{Nooe, 8, 8, 64)
	output:	(Nooe, 8, 8, b4)

It\ag oo1ine2d_3: MagPoofiog2D	input:	(f4one, 8, 8, 6d)
	oug›ut:	(t'Jone, 4, 4, b4)

conv2d 4. C…2D	input	(None, 4, 4, b4)
	oytput:	(Nooe, 4, 4, 64)

max ooling2d_4 : MaxPooling2D	input:	(None, 4, 4, 64)
	output:	(None, 2, 2, 6)

flatten_1: Flatten	input:	(None,2,Z,6Q
	output:	(No*z,256)

dense_1: Dense	input:	(None 256)
	output:	(None, 128)

dropout_1: Dropout	input:	(None, 128)
	output:	(None, 128)

dense_2: Dense	input:	(None, 128)
	output:	(None, 64)

dropout_2: Dropout	input:	(None, 64)
	output:	(None, 64)

dense_3: Den se	input:	(None, 64)
	output:	(None, 32)

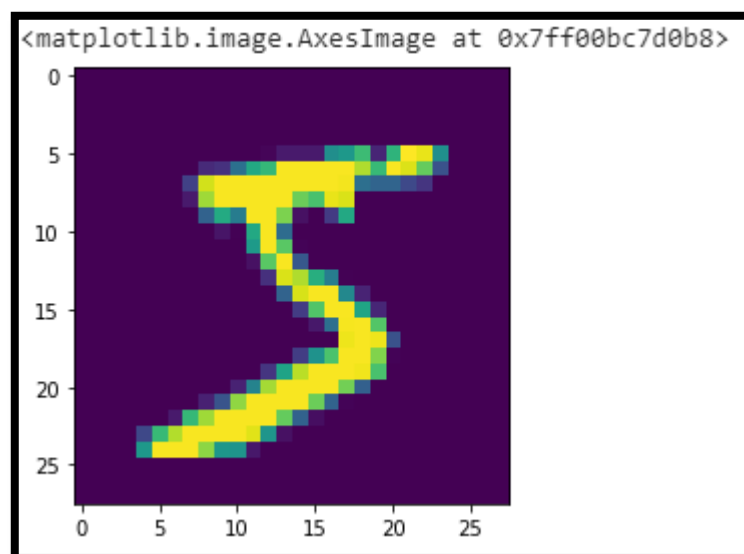| dropout_3: Dropout	input:	(None, 32)
	output:	(None, 32)

Cat/Dog Classifier gets an accuracy of 49.53% after using optimizer as adam and the loss function as binary_crossentropy.

For MNIST, Out of the 70,000 images provided in the dataset, 60,000 are given for training and 10,000 are given for testing.

conv2d_7_inpui: InpulLayer	mput	(Alone, 28, 28, 1)
	ouQuc:	[None 2B, 2B, 1)

COTlV2d 7: Gonv2D	input:	(None, 2B, 2B, 1)
	output	(Nnne, 26, 26, 64)

coov2d_8: Conv2D	mpuu	(None 26, 26, 64)
	oul@ub	(NoOe, 24, 24, 32)

Qatten_3: Flahm	input:	(None, 24, 24, 32)
	oupul:	{None,18432)

dense 7: Dense	input:	(None, 18432)
	output:	(None, 10)

model : "sequent1al_3 "

Layer (type)	Output Sh ape	Param #
c onv zd_7 (Conv2D)	(None, 26, 26, 6-4)	640
c onv zd_B (Conv2D)	(None, 24-, 24, 32)	18464
-F1atten_3 (FI a tten)	(Non e, 18432)	0
dense_y (Dense)	(None, le)	184330

Total pa raw s : 283, 434
*Tra1*nable p a rams: ze3, 434-
Non -l r a1 nab 1e par ams : e
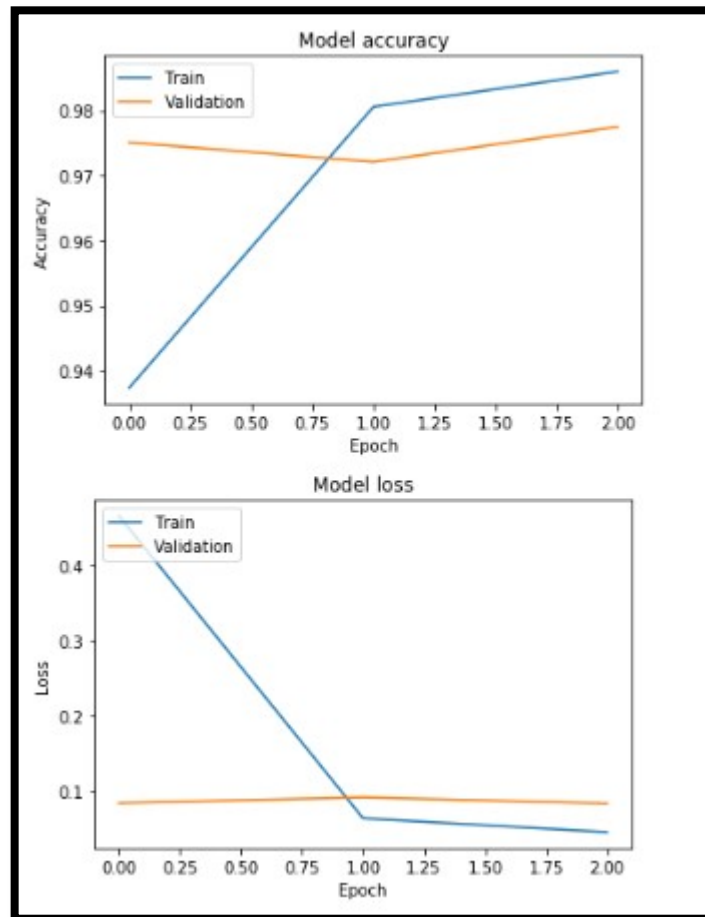
Accuracy: 98%

Code:
https://github.com/visheshtechie/DL/blob/master/Lab9_Keras_2DCNN.ipynb