MSBD 5012 Machine Learning Homework 4 Report

Student: XXXXX ID: 12345678

In this homework, I use different pretrained model to classify our test set. I summarize the result in the coming chapters. The first part talks about VGG16, the second part is about ResNet50, the third part shows InceptionV3, the fourth part demonstrates DenseNet121, the fifth part talks about EfficientNetB2. The last part is about MobileNetV2.

Part One: VGG16

In this part I use a pretrained VGG16 model to classify our test set. We show 5 classes with the maximum probability. The result is as follows:



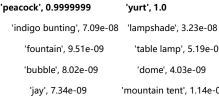








'rapeseed', 0.99995124 'barn', 2.05e-05 'school bus', 1.01e-05 'worm fence', 5.47e-06 'hay', 2.34e-06



'yurt', 1.0 'table lamp', 5.19e-09 'pedestal', 1.0e-13 'dome', 4.03e-09

'hourglass', 1.0 'guillotine', 1.9e-12

'mountain tent', 1.14e-09 'China cabinet', 3.6e-14 'dome', 2.54e-07

'fireboat', 1.0

'beacon', 3.39e-06 'hourglass', 4.44e-07 'table lamp', 9.5e-14 'radio telescope', 2.81e-07

'water tower', 0.9999951



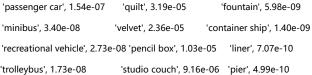




'school bus', 0.99999964 'passenger car', 1.54e-07 'minibus', 3.40e-08

'trolleybus', 1.73e-08







'carousel', 1.0 'toyshop', 1.21e-10 'throne', 4.31e-11 'vestment', 3.53e-11 'slot', 1.69e-11

All the input pictures are resized (224, 224). From the result above, we can see that VGG16 has 100% accuracy on our 10 pictures. This model can fully cover our task. Besides, from the other result with top 5 probabilities, I think maybe VGG classifies the picture by the texture and the color.

Part Two: ResNet50

In this part I use a pretrained ResNet50 model to classify our test set. We show 5 classes with the maximum probability. The result is as follows:



All the input pictures are resized (224, 224). From the result above, we can see that **ResNet50** even focus more on the main item in the picture. This can be seen easily in the 10th picture compared with the result of VGG16. Because the second and the third prediction rates in VGG16 are toyshop and throne (I think mostly decided by the background), but those in ResNet50 refers to Arabian camel and Indian elephant. I think this is because ResNet50 has more layers, even three times more than VGG16. These convolutional layers and linear layers help ResNet50 focus more on the foreground.

Part Three: InceptionV3

In this part I use a pretrained InceptionoV3 model to classify our test set. We show 5 classes with the maximum probability. The result is as follows:













'zebra', 0.9939 'gondola', 2.04e-04 'ostrich', 1.51e-04 'hartebeest', 1,17e-04

'pickup', 5.24e-05

'school bus', 0.9966 'toucan', 3.88e-04 'hornbill', 6.98e-05 'box turtle', 3.05e-05 'remote control', 2.44e-05

'radio', 1.07e-04 'jeep', 8.44e-05 'pencil box', 7.05e-05 'mailbag', 4.95e-05

'pillow', 0.9953

'fireboat', 0.9524 'bittern', 0.001 'badger', 7.63e-04 'fountain', 4.70e-04 'quill', 4.67e-04

'carousel', 0.9749 'toyshop', 3.51e-04 'confectionary', 2.63e-04 'slot', 2.56e-04 'swing', 2.48e-04

All the input pictures are resized (299, 299) (different from the above 2 models). InceptionV3 is different from the above two models because it needs to use its own preprocess input function. This model uses small convolutional kernels, which makes the parameters less than the above two models. From the result above, we can see that Inception also gives the correct answers, but the probabilities are all lower than the above two models. Besides, from the 4th results we can find that InceptionV3 uses more of the color of a picture, because a red and grey hourglass can be predicted as grey fox or red wolf (2nd and 4th largest probability).

Part Four: DenseNet121

In this part I use a pretrained DenseNet121 model to classify our test set. We show 5 classes with the maximum probability. The result is as follows:











'rapeseed', 0.999845 'stone wall', 3,40e-07 'velvet', 9.77e-08 'snowplow', 8.71e-08 'bison', 6.69e-08

'peacock', 0.9996 'fountain', 2.23e-04 'mosquito net', 3.07e-05 'pillow', 1.11e-05 'safety pin', 7.24e-06



'barn', 5.47e-07

'mosquito net', 4.26e-07

'scale', 2.92e-10 'guillotine', 2.13e-10 'pedestal', 1.47e-10

water tower', 0.99977 'beacon', 1.57e-04 'radio telescope', 8.38e-06 'pier', 6.83e-06

'crutch', 3.67e-06









'China cabinet', 1.06e-10



'tiger', 3.76e-05
'tiger cat', 1.51e-05
'worm fence', 8.99e-06
'jaguar', 2.54e-06

'school bus', 0.99998 'passenger car', 8.37e-07 'amphibian', 4.56e-07 'trolleybus', 2.32e-07 'cab', 1.99e-07

'pillow', 0.99999 'pencil box', 7.96e-08 'studio couch', 3.21e-08 'quilt', 2.35e-08 'purse', 1.63e-08

'fireboat', 0.99999 'fountain, 2.09e-06 'speedboat', 1.96e-07 'mosquito net', 1.29e-07 'lifeboat', 8.70e-08

'carousel', 0.99997 'altar', 7.70e-06 'throne', 4.68e-06 'vestment', 3,23e-06 'maypole', 1.53e-06

All the input pictures are resized (224, 224). Besides, we use the preprocess input function from the InceptionV3 package (Otherwise can' t get a satisfying result). From the result above, we can see that DenseNet121 mainly focuses on the texture of the picture. It can be seen in the 2nd 3rd, 6th picture, for the second and third output of these pictures are predicted mainly by the lines of the pictures.

Part Five: EfficientNetB2

In this part I use a pretrained EfficientNetB2 model to classify our test set. We show 5 classes with the maximum probability. The result is as follows:



All the input pictures are resized (260, 260). EffecientNetB2 has a smaller amount of parameters and can get a rather good result. Note that we don't need to add a preprocessing for the input image, because EffectiveNet has a Rescaling layer for that. EffectiveNet uses a more clear image (with more pixels input) to maintain its accuracy even in a smaller amount of parameters. From the result we can find that the accuracy decreases to some extent, but for a classification problem, the main task is to classify correctly. While other classes still hold relatively small probabilities, I think this model works well.

Part Six: MobileNetV2

In this part I use a pretrained ResNet50 model to classify our test set. We show 5 classes with the maximum probability. The result is as follows:



All the input pictures are resized (224, 224). Besides, we use the preprocess input function from the MobileNet package (Otherwise can't get a satisfying result). MobileNet is a relatively light network because it is maily used on mobile devices. From the result above, we can see that MobileNetV2 mainly learns texture (lines) and color to classify the picture. The most salient example is the 2nd and the 6th picture (lines), and the 7th picture (color).

Conclusion

In this experiment I use some pretrained models to classify existing pictures. These models have their own cons and pros. And I also try to analyze the result that model gives.