Proposing a method to calculate the invariance of a model

We developed a method to calculate how much a model is invariant to various transformations.

The reason to calculate the invariance of a model is that in a traditional way to get the estimation of how much an augmentation technique is going to improve the performance of the model, we need to generate the augmented data, train the model on that data and testing the model on validation or test dataset. The problem with this approach is that we can only get the estimation of the performance of the model after training it. Here, training time can range from a few hours to a few weeks. So, if the augmentation technique which we are going to perform is not improving the performance of the model or in the worst case where the augmentation is degrading the performance of the model, the time that we spend on training the model is wasted. So, to avoid the wastage of time and choose the right/required augmentation techniques we propose a method to calculate the invariance of the model so that we can know whether we should apply a transformation technique or not.

Experimental setup that we are using:

Dataset: Cat vs Dog

Size: 25000

Training size: 3000 (1500 cats and 1500 dogs)

Testing size: 22000

Method:

Step 1: Develop a basic model. This model is trained on the original dataset. In our case, we trained the model on 3000 images.

Step 2: Select some random samples from the same original dataset. And make sure that these samples are perfectly predicted by the model. In our case, we have selected 100 such random samples.

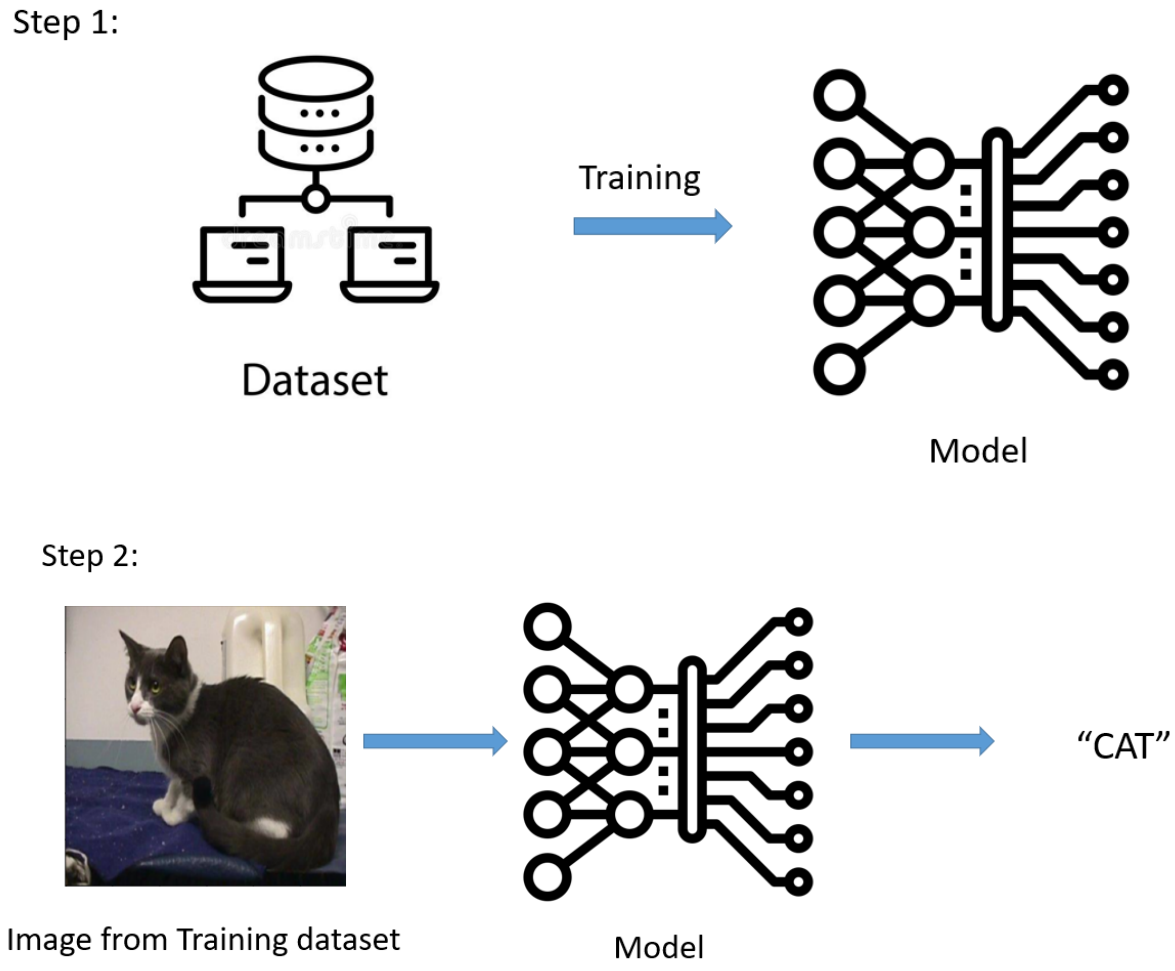
Step 3: Develop the feature map using which the model is predicting these 100 random samples. Let's call them "Original Feature maps".

Step 4: Select whatever augmentation technique you want to test. For example, let's select the 45-rotation technique. Now, we will apply 45 rotation

augmentation on the above 100 samples and pass these images into the model and develop the feature maps of these augmented images. We will call them “Augmented Feature maps”. Next, we will again take the “Original Feature maps” and perform the same augmentation on these maps and we will call them “Augmented Original Feature maps”.

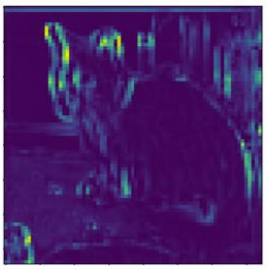
Step 5: Now, we will compare the “Augmented Feature maps” and “Augmented Original Feature maps”. The amount of overlapping these two feature maps will have is the invariant score of the model. We can use metrics like SSIM (Structural Similarity Index Method) to measure the overlapping. Or we can just compare pixel to pixel and on average how much they are variating. And before comparing the pixel-to-pixel variation we have normalized the values of the pixel to the 0-1 range.

Pictorial Representation of the above method:



Step 3:

Image from Training dataset



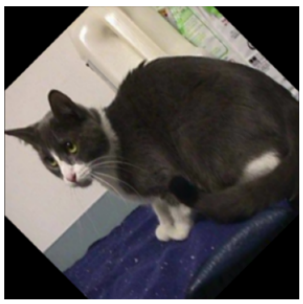
Original features

“CAT”

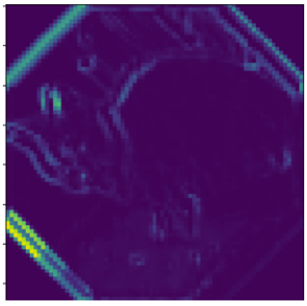
Features learned by the model

Model

Step 4:



45 degree Augmented Image



Augmented Features

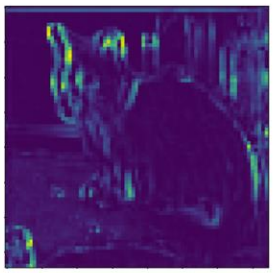
“CAT”

Features detected by the model

Model

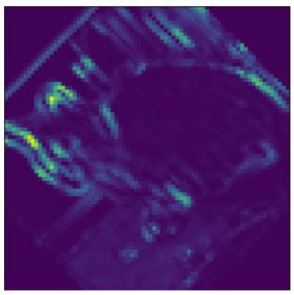
Step 4:

Original Sample



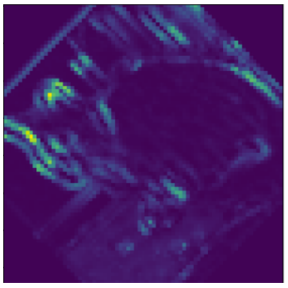
Original Features

Augmentation

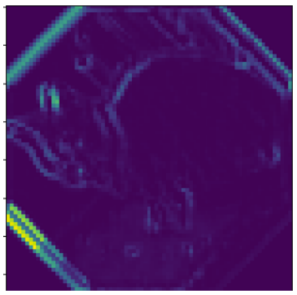


Here, we are applying same augmentation on the original feature map

Step 5:



Original Features



Features detected on Augmented Image

Amount of overlapping

"Invariance Score"

We are still working on the code. The main issue we are facing in generating stable scores is the unnecessary activation by the model for example:

We are working on how to minimize these activations and generate stable scores for invariance. As soon as possible, we will upload the code regarding the same. So, that is our idea, Sir. It would be really helpful if we can get your view and insights about this method.

Thank You