Cat classifier

Albert Gregus

December 2021

1 Introduction

The aim of this task is to make a classifier, that can tell which of the two cats (Sajt and Pali), or both or neither are in the image. As in this problem we actually have 2 classes (the 2 cats), of which either is possible on the images, we actually have a multilabel problem of the two cats. Furthermore, the problem is more difficult, as we have no negative samples (images with neither cats).

2 Exploratory data analysis (EDA)

To have a deeper understanding of the data EDA was made. The full data analysis can be found in *notebooks/EDA.ipynb*.

2.1 Data imbalance

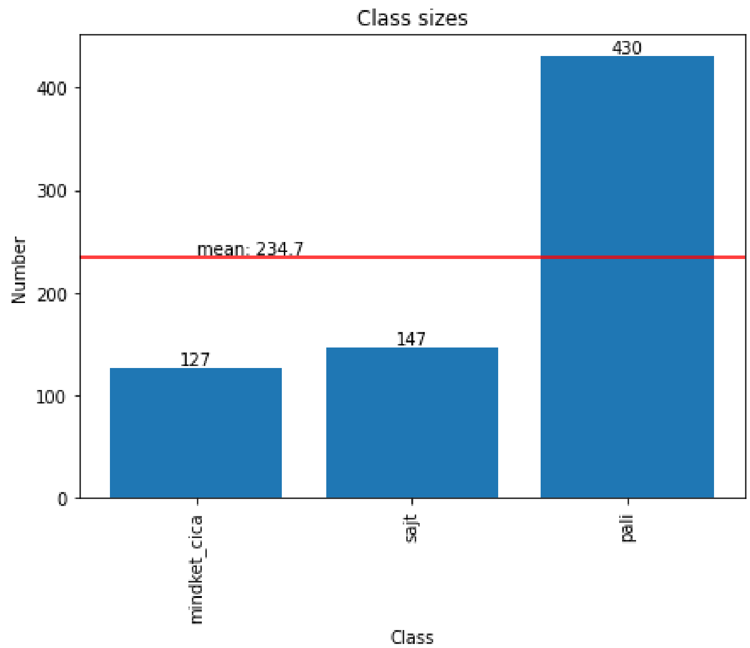

Figure 1: class imbalance

There is a small data imbalance in the dataset

Over- and undersampling
A way of dealing with class imbalance is over- and undersampling. I sampled the classes to have the same number of examples: the mean example number (but I oversampled at most 10 times to avoid overfitting). The sampling was remade randomly after every epoch, this way every element from the large classes are used during training (and not just a subsample of the classes).

There are no images without cats (negative examples)

Ask customers for negative examples
The easiest way of solving this problem is to ask customers to provide images without cats, but from the same domain (house and garden where the cat images are from). This way all the images would be from the same camera and environmental domain, and could be used as negative examples as they

are. Without this there is no way of precisely telling the accuracy of the model.

Train without negative examples

We can train the model without negative samples using separate sigmoid activations. Here we have to define a threshold, from which we consider an object detection positive. Then if the model recognizes one of the cats than it will have a higher than threshold, else it will have a lower than threshold prediction. Hence it can detect whether there is no cat in the image. That said, without negative examples we can not be sure about what will happen in new negative images, and can not be sure whether the model learns to recognize negative examples or not.

Use negative examples from other datasets

An other possible solution is to use negative examples from other datasets. As these examples are from a different domain it could be beneficial to change these to have similar domain or have a selection method to use hard negative examples (similarly to contrastive learning).

2.2 Cat number

The images only contain one, both or none of two cats: *pali* and *sajt*. This can be treated in two ways:

- regular classification: there are 3 possible objects in the single object images: pali, sajt, mindket_cica
- multilabel classification: there are 2 possible objects in the multilabel images: pali, sajt

During the experiments both were implemented and compared.

2.3 Rotation and resolution

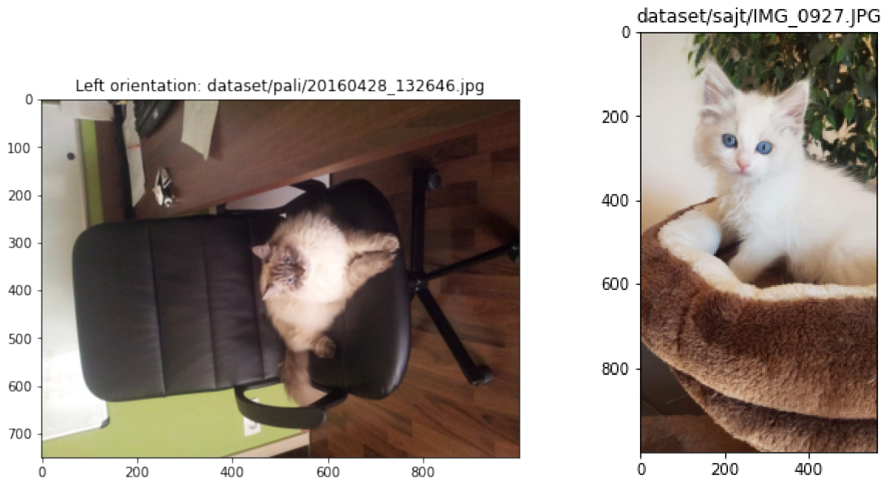

Figure 2: Images of different orientation and resolution

There are images with different resolution, and also images have different orientation (standing or lying). Using the images as they are they could not be used in batches (because of the different resolutions) and would have some images in standing and some in standing position.
To solve this after cropping I resized the images to square and randomly rotated them, this way the images could be put into batches and the model will learn cats from all directions and will be rotation indifferent.

2.4 Cropping

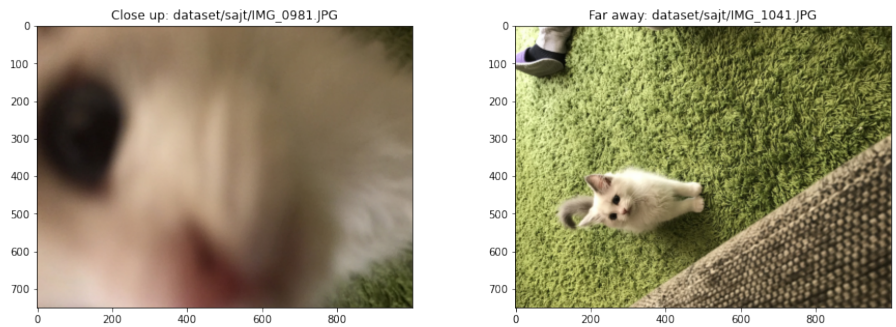

Figure 3: Images with different cat sizes

There are images where the cats are far away, while there are also images where the cats are close. To make sure that the cropping contains the cats I made only smaller crops. The drawback of this method is that this way we can not make a diverse cropping augmentation.
A more sophisticated method could be used, where an already trained cat detector could be used to detect the location of cats. In images where the model is sure about the cat detections we could use crops that include enough of the detected cats, while in images where it is not sure about the cats we could use the regular cropping.

2.5 Mislabelled images

There were some mislabelled images, which were in either *pali* or *sajt* folders, but contained both cats. These were manually selected and put into the right folders.

3 Experiments

The trainings can be treated as singlelabel, hence treating differenc scenarios (sajt, pali, sajt and pali, none), or multilabel, hence treating it as an image that can have two objects: sajt and pali. This way there were 4 different types of trainings:

- softmax with single label: the base scenario, where we use softmax across the four classes to enforce them to sum to 1
- sigmoid with single label: using binary loss for all classes
- sigmoid with multilabel: using binary loss for the two objects, and a minimum threshold to decide whether the particular object is present.

Furthermore, as there were no images without the cats, the model has to learn to recognize the the cats well enough to decide whether there is or there is not at least one of them in the picture. As this is a hard problem, it would be easier to include images without any of the cats. To simulate this I included handpicked scenes from SUN dataset.
A further complication comes from the small number of images. While we only have around 550 images, the objects are similar to each other, which makes it hard for a neural net to learn their difference.

3.1 Training with and without negative examples

Additional negative examples could be used to train the model to distinguish between images with the cats, and images without them. Training the model without negative examples and evaluating them on a validation dataset with negatives, we can see that their performance drops to random guess, hence without negatives we can not learn to predict images without cats.

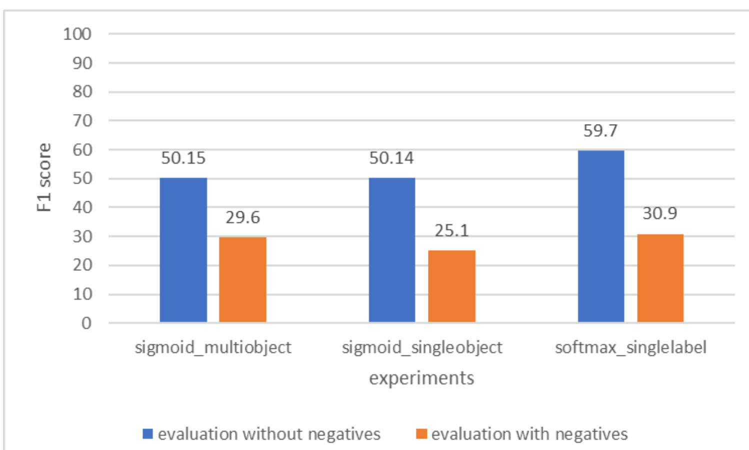

Figure 4: Evaluation results on trainings without negatives

3.2 Single and multilabel training

Comparing the single and multilabel trainings on trainings with and without negatives, we can see that the singlelabel performs better on both.

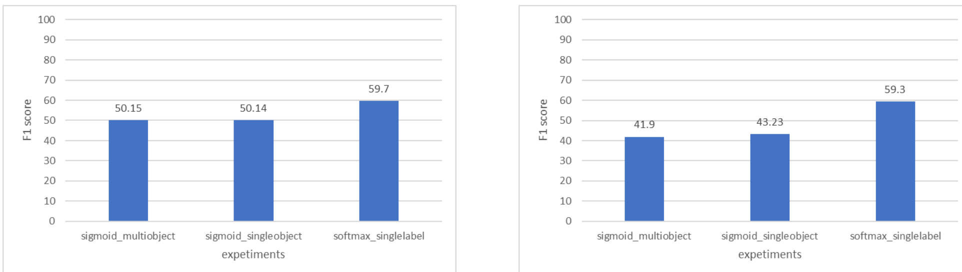

Figure 5: Evaluation results on different trainings without (left) and with (right) negative examples

3.3 The effect of small data

As we have a small amount of data, the network is prone to overfit. Also, as the classes are imbalanced, the model can overfit to the biggest class instead of predicting for all classes.

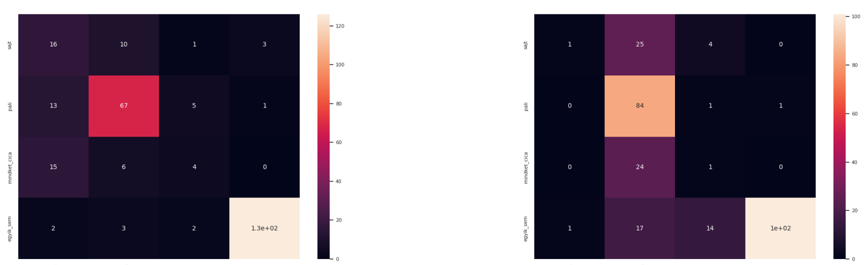

Figure 6: A good training (left) and a training where the network overfitted to always predict pali

4 Model visualization

According to the model visualization the most important parts for the model during predictions are:

- cat ears
- cat eyes
- cat nose
- cat face
- cat paws

During during good predictions (first 2 image) these are the important areas, while bad predictions (last image), there are no clue of using these areas.

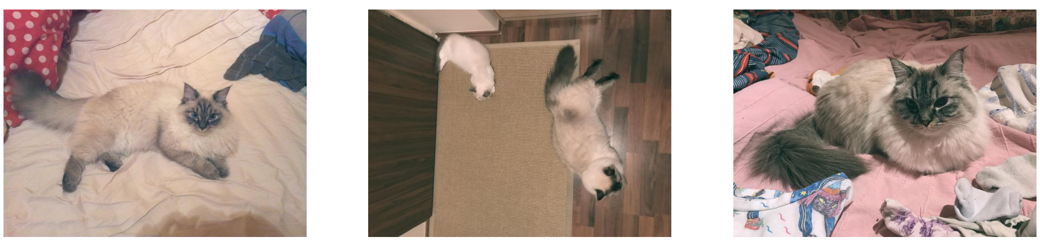

Figure 7: Images of cats

Figure 8: Guided gradients

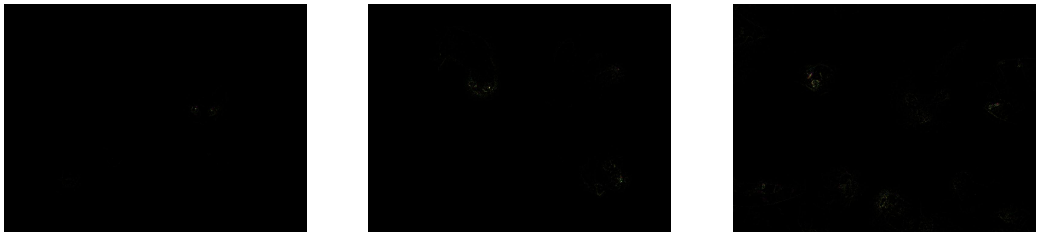

Figure 9: Guided saliency map

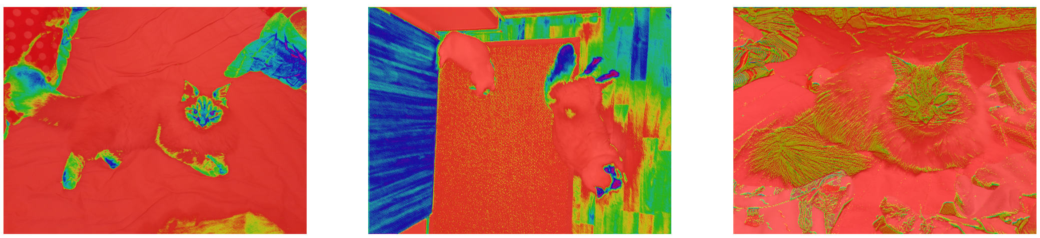

Figure 10: GradCam

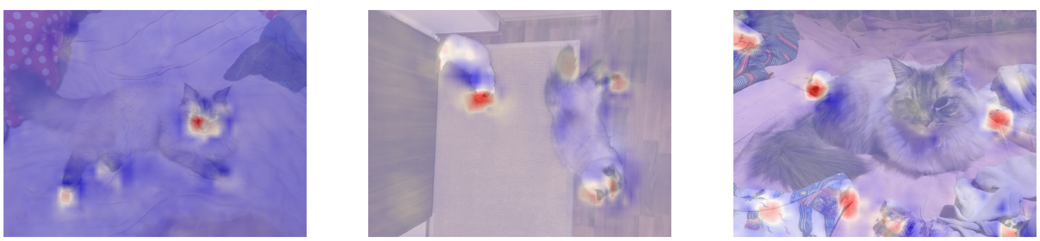

Figure 11: Occlusion sensitivity