BirdCLEF2021: An ornithology expert system

September 29, 2021

Edoardo De Matteis

Abstract

In this report I approached the BirdCLEF2021 bird classification task, the dataset contains a huge amount of data, making it difficult to work with. I came up with a model that uses only soundscape recording, thus making most of the data privileged, and getting good results. The code is available on GitHub: https://github.com/edodema/Birdcalls

1. Introduction

Birds are sensitive to changes in the ecosystem due to them being high up in the food chain, so monitoring them can give us informations about pollution and the environment. While small birds are often difficult to sight they tend to sing a lot, and these calls are generally characteristic of each species, that's why a smart way to identify them is through their call. Our goal is to guess a bird's species by its call in a noisy recording, the task is set up as an image classification among $n+1$ classes i.e. n species plus no bird being detected.

Dataset. The BirdCLEF2021 dataset contains recordings of both birds in a controlled environment and noisy soundscapes, the latters being split in 5 seconds windows and labeled with primary and secondary labels. For our purposes I only considered primary labels, that should not be a problem since we are just being conservative, secondary ones are just some of the privileged data that we will ignore, in fact we will not use birds recordings at all and still get good results.

2. Related work

Many previous works have exploited spectrograms to reduce audio recognition to an image task (Hamdy et al.) (Michelashvili & Wolf, 2020) (Xie et al., 2021), among them (Xie et al., 2021) uses a recurrent layer to extrapolate time sensitive knowlgedge. In the same way we canuse attention and one of the basic blocks that will form our model has been inspired by (Zhang et al., 2020), in which a variation of *ResNet* (He et al., 2016) with split-attention is developed. Also from *ResNet* itself I took some numerical techniques to stabilize results.

3. Method

What happens when ornithologists recognize a birdcall, do they first detect a sound and then focus to classify it or does their brain automatically recognize the bird species? Let's call the first approach *split* and the second one *joint*, I tried both of them but only the latter model is covered here due to computational issues with the first one (for those interested, results are still shown on the GitHub repository).

Preprocessing. The dataset recordings, regardless of bird calls or soudscapes, are highly unbalanced (figure 1), that's why the dataset is augmented by random oversampling. Then it is split in train, validation and test sets by a 80-10-10 ratio and each 5 seconds window in audio tracks is encoded as a spectrogram in mel scale. Transforming spectrograms (e.g. random crops) could have led to losing positive data (i.e. a bird singing) and keeping silence, or vice versa, leading to wrong labeling; even if that did not happen it would have not been strange to lose relevant data, due to the task being modeled as seq2seq. Anyway due to computing limitation it was not possible to use all the data so a smaller split of randomly selected data was used during training.

CNNAtt. This custom defined layer expoits multi-head attention layers to build a seq2seq model with the intent to grasp time-series knowledge. First the spectrogram passes through three different convolution pipelines, each one used respectively as a query, key and value for the attention. The RNNs in the attention layers work on tensor rows independently, this means that we would learn looking at frequencies with no connection whatsoever between them, thus a second attention layer works on transposed spectrograms' feature images. The two representations are then concatenated to get a 2 channel image. As a note,

Email: Edoardo De Matteis <dematteis.1746561@studenti.uniroma1.it>.

Deep Learning and Applied AI 2021, Sapienza University of Rome, 2nd semester a.y. 2020/2021.

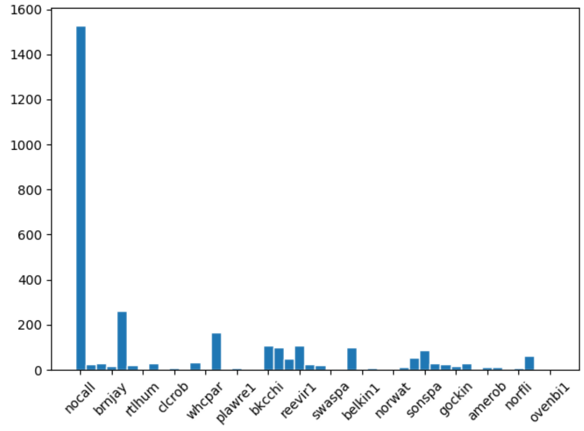

Figure 1. Primary label distribution in soundscape recordings.

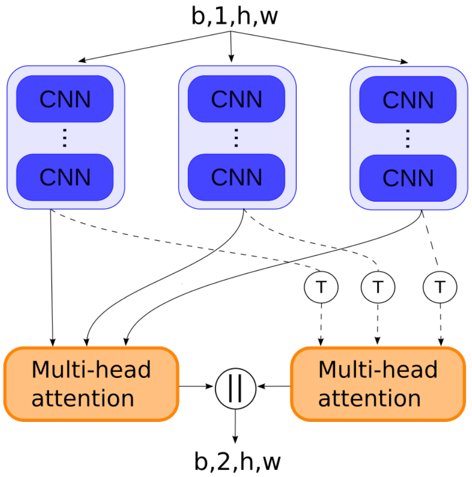

Figure 2. A CNNAtt block.

when we refer to a CNNAtt$k$ block it means that convolutions have all the same kernel size k.

CNNRes. Residual networks are effective at reducing error when dealing with deep networks, and I tried one pretrained *ResNet18* model as a feature extraction backbone. Also this simpler residual network has been defined, it employs six convolutions of the same kernel size among with two residual paths between them (figure 3). Is possible to stack n layers one on top of the other, in such cases we will write CNNRes$n$, as a rule the number of kernels stays the same when the feature map has the same size as the input image, and it is doubled when feature map has half the size of the image (He et al., 2016). There are no residuals between different CNNRes block therefore there is no need in dealing with tensor with different shape.

The afore mentioned layers build up the feature extraction backbone, between CNNAtt and CNNRes we have another

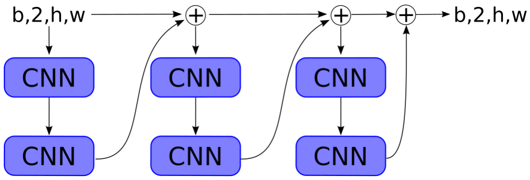

Figure 3. A CNNRes block.

convolution to deal with the number of filters. After that we have m bilinear GRU layers as a seq2seq model to extract other sequential knowledge, and last a linear layer to output logits.

In our experiments we use a cross entropy loss, and use both models build from scratch i.e. CNNRes2 (with kernel sizes 3 and 5) and CNNAtt5CNNRes2 that adds a CNNAtt block to it, according to the afore-mentioned rules. But we also fine-tune pretrained models *ResNet18* and *ResNet50*, with a convolution and a linear layer to match *ResNet*s input and output shapes.

4. Results

Training has been done using an *Adam* optimizer with 8 samples per batch, primarly due to memory limitations but it turned out good sinc we get some nice spikes. Results are reported at table 1, since datasets have been balanced and all classes are equally important it makes sense to use accuracy.

Table 1. Performances are for the validation set with a fixed seed of 42, the test accuracy is reported for CNNRes2 only.

Model	LR	Loss ↓	Acc. ↑
ResNet18	1e-4	.21	92.2%
ResNet50	1e-4	-	%
CNNRes2	1e-4	**.02**	**99.1%**
CNNAtt5 + CNNRes2	1e-4	.04	98.7%
Test	-	-	99.4%

Occam's razor holds and it seems that a simpler approach is the way to go, pretrained models are not very good but it can happen with transfer learning.

Future works. It would be nice to have enough resources to train on the whole dataset, this way we could also effectively try the split method. Also adding a threshold as an hyperparameter we could consider secondary labels too.

References

Hamdy, A., Vedula, P. K., and Konduru, M. V. J. Audio separation and isolation: A deep neural network approach.

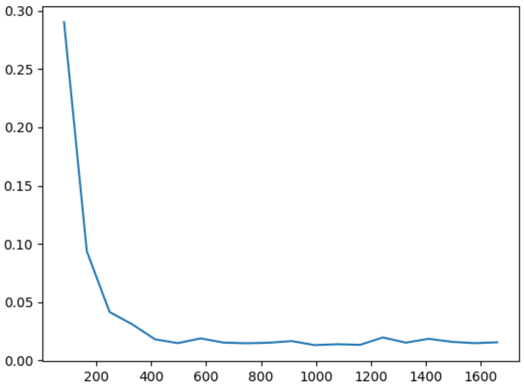

Figure 4. Validation loss for CNNRes2GRU1FC1.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Michelashvili, M. and Wolf, L. Speech denoising by accumulating per-frequency modeling fluctuations. *arXiv preprint arXiv:1904.07612*, 2020.

Xie, J., Aubert, X., Long, X., van Dijk, J., Arsenali, B., Fonseca, P., and Overeem, S. Audio-based snore detection using deep neural networks. *Computer Methods and Programs in Biomedicine*, 200:105917, 2021.

Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., Sun, Y., He, T., Mueller, J., Manmatha, R., Li, M., and Smola, A. Resnest: Split-attention networks, 2020.