AnimalSoundsHLC: Hierarchical Classification of animal sounds using Local Classifiers

Gustav Lang Moesmand (s174169), Alba Reinders Sánchez (s212729) and Alejandro Valverde Mahou (s212728)
Supervisor: Ole Winther

Introduction

We propose a neural network approach for animal sounds classification. This task is specially difficult when the data samples have noise, or the differences between samples are very subtle. It also depends on which level we want to classify, i.e. how many classes there are to classify. Deep learning has been proven to be useful solving complex problems in the audio field [3]. That is why we use Convolutional Neural Networks (CNNs) and Long Short-Term Memory Networks (LSTMs) to try to find the best solution for the problem stated.

To solve this problem we explore a hierarchical method. Real life classification problems can be structured as a hierarchical tree. Usually, neural networks use a naive approach where all elements are at the same level and do not have structured information. For this problem, we try to use the structured information that the taxonomy of animals has, using as level the *phylum*, *class* and *family* of each animal. This way we want to compare if the results obtained with this approach are better than the ones from the naive approaches.

Key points

- Create a new dataset based on the Animal Sound Archive[1] transforming the sounds to both MFCC and Mel Spectrograms to feed the model.
- Classify in the deeper level possible. The data is structured in *phylum*, *class*, *order*, *family*, *genus* and *species*. We are going to classify over *phylum*, *class* and *family*, with *232* different *families*.
- Use data augmentation to solve overfitting, among other techniques.
- Compare results between the CNN approach against the LSTM approach.
- Show the differences between the Flatten one-hot method against the Hierarchical method, and try to prove if the Hierarchical is more suitable to this problem.

Flatten models

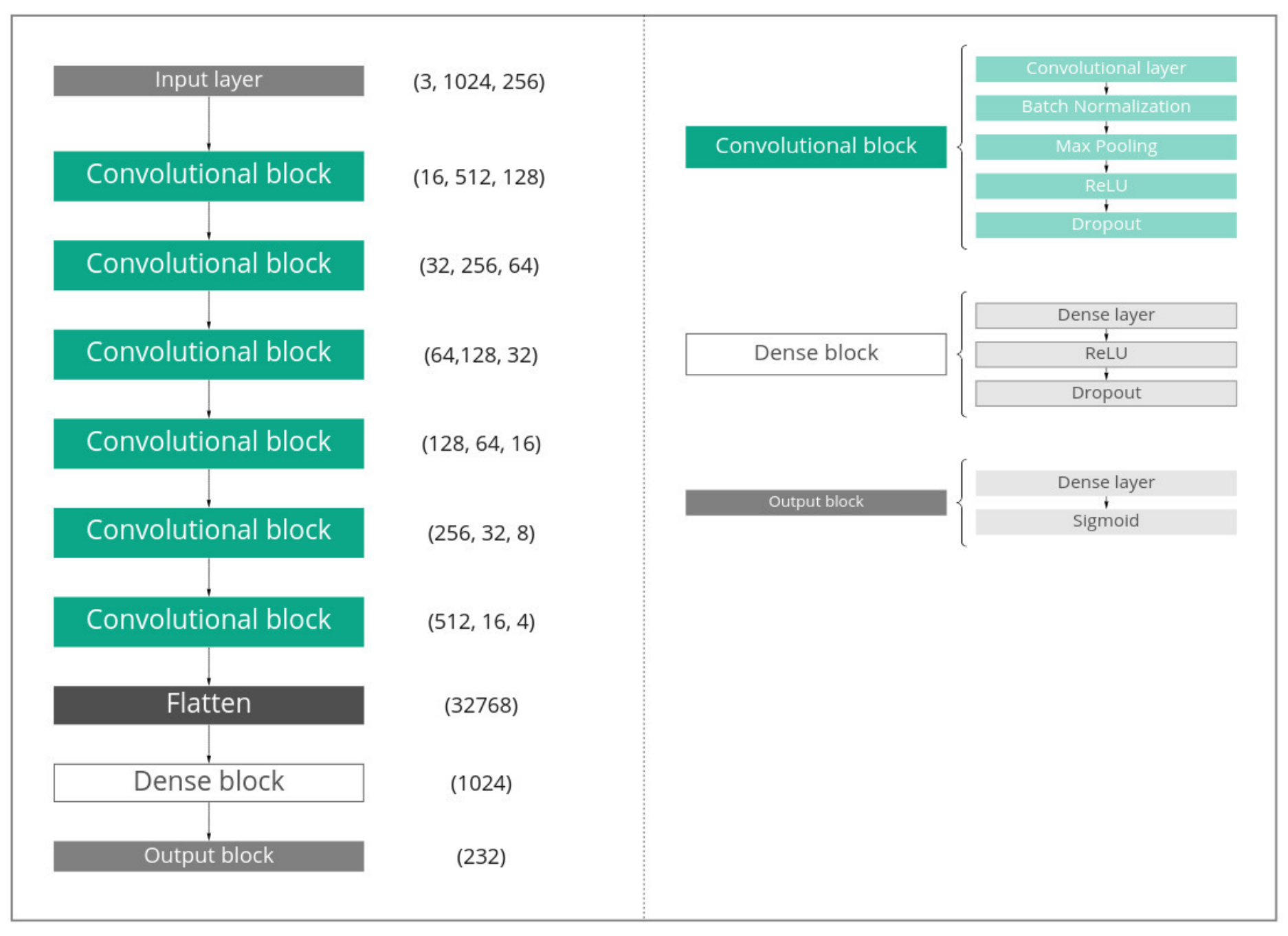

Figure 1: *FlattenCNN*

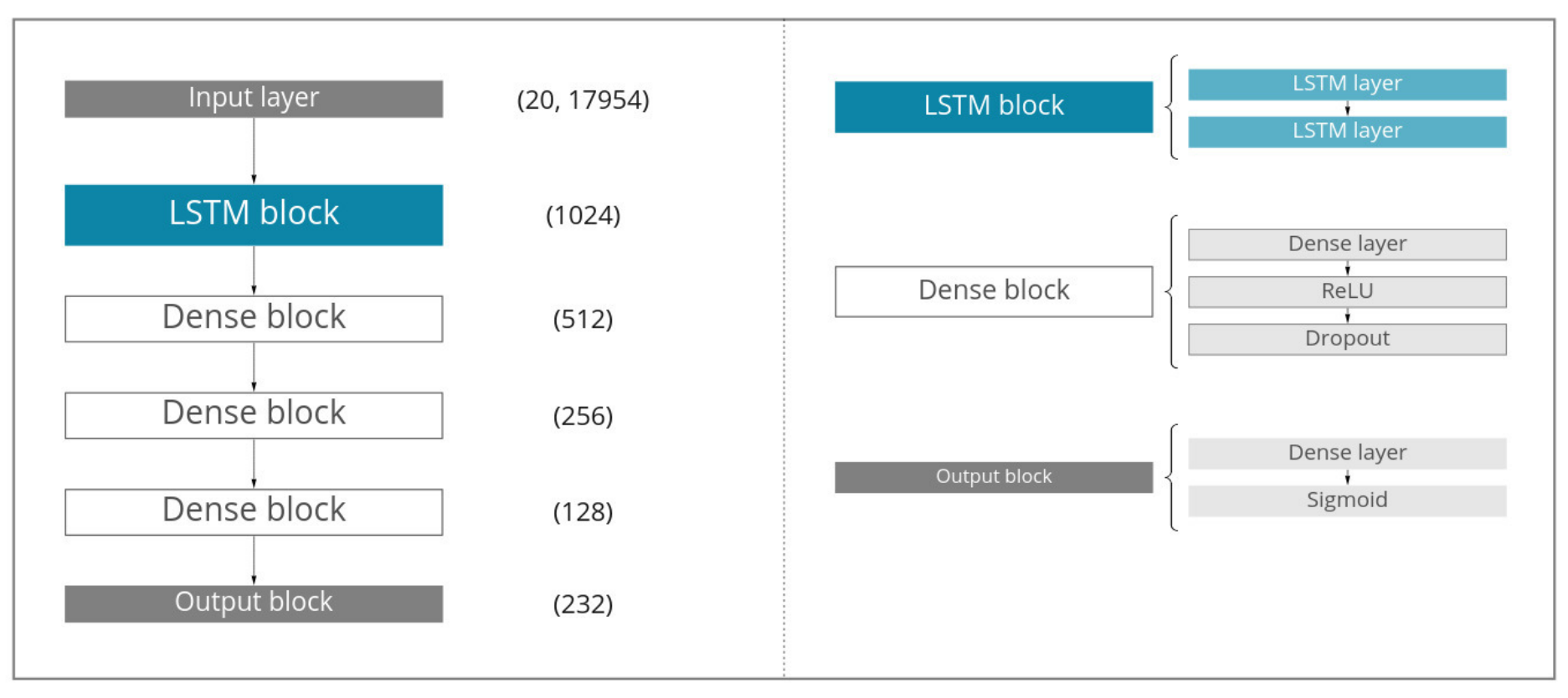

Figure 2: *FlattenLSTM*

The first approach for this problem is the naive approach, that we denominated *Flatten*, as it only classifies over 1 level. Two of the most used arquitectures in audio classification are CNNs and LSTMs, so we proposed two different models, one with each architecture: *FlattenCNN* and *FlattenLSTM*.

The architecture of the best CNN model is the one showed in Figure 1, with six convolutional blocks, one flatten layer and one dense block. While the architecture of the best LSTM model is made of one LSTM block and three dense blocks, as is can be seen in Figure 2. These final models are chosen based on experimentation and their accuracy results.

Hierarchical model

The second proposed approach is a hierarchical local classifier per parent node [4]. This model uses a different classifier in each node of the hierarchical tree, except at the leaf level. Each of those classifiers can be either a CNN or a LSTM, and every one of them is trained with the whole train dataset.

We decided to use local classifiers as they can use the structured information of the data, and it does not need complex and detailed tailoring for our concrete problem, as a global hierarchical classifier would need. Even if it is not too complex, it is very time consuming, as it requires the training of multiple classifiers. That is why we choose to skip the *order*, so we do not have to build even more additional classifiers. For the same reason, we do not try to classify for levels that are under *family*. With the model that uses *phylum*, *class* and *family*, 10 different classifiers are needed.

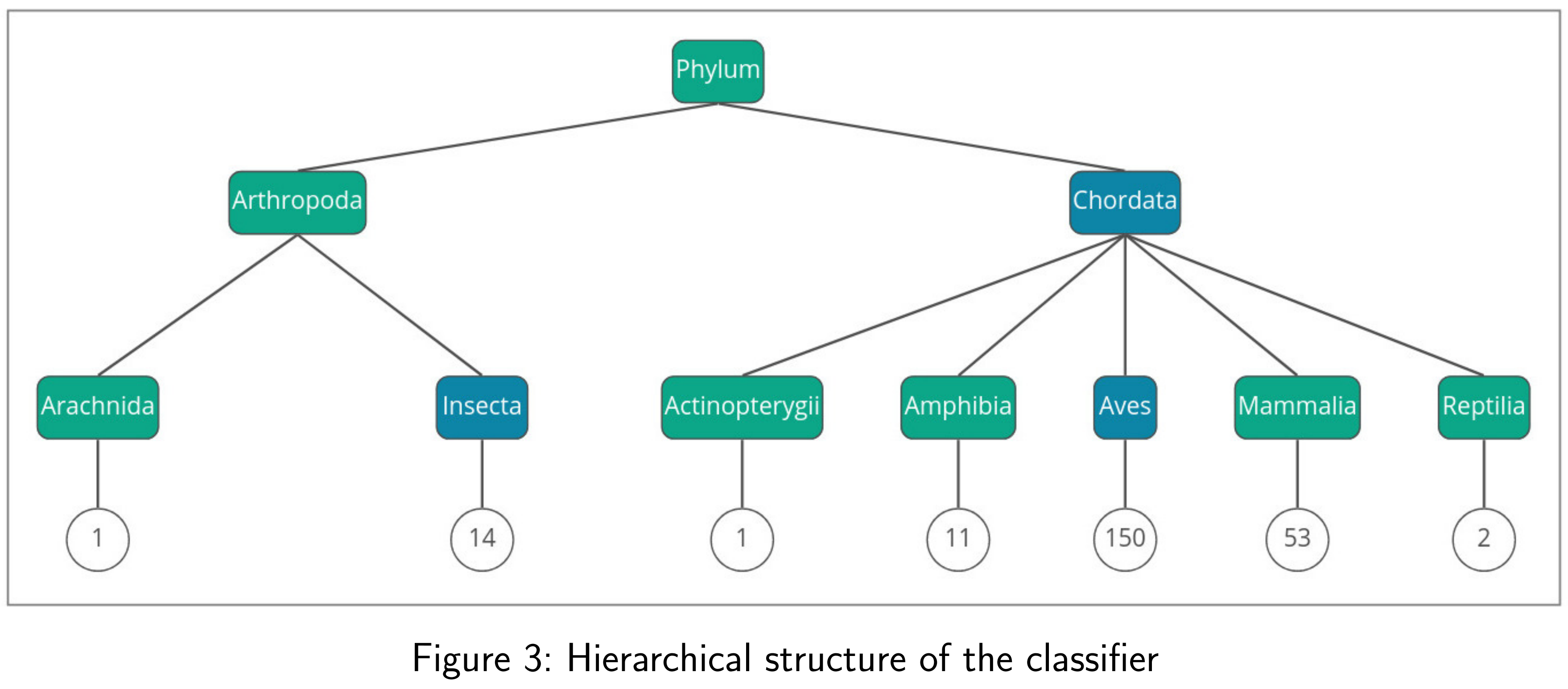

Figure 3: Hierarchical structure of the classifier

Preprocessing

The final data set consists of over 16000 samples of different species. The original data format is stored in *.mp3*, and to prepare it for the CNN's, it is first transformed to waveform format and then into a Mel Spectogram [3]. The former is a representation that uses a specific logaritmic scale modelled after how human hearing works.

To train the LSTM's, the waveforms are transformed into MFCCs (Mel Frequency Cepstral Coefficients). These are the discrete values of the Mel Spectogram more fitting for the LSTM [3].

As seen in Figure 4, most of the samples already have some noise, and we decided not to add more.

Instead, in an effort to generalize the model more, we use data augmentation removing random frequencies or timeslots from the Mel Spectogram. This is done by drawing a black line across random places in the image either horizontally (frequency) or vertically (timeslot).

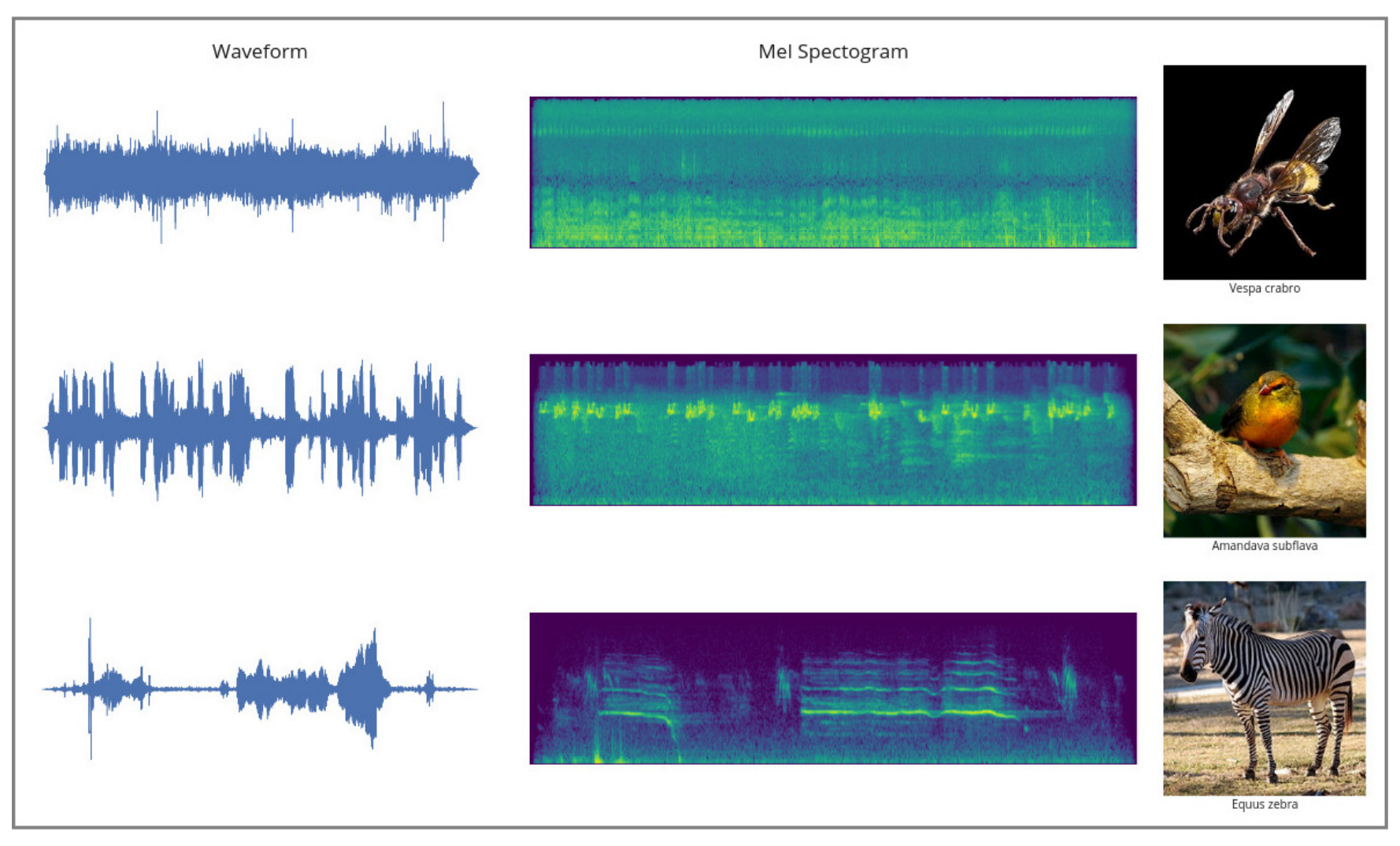

Figure 4: Visualization of the preprocess of the data for the CNN model

Models performance

The difference of accuracy between the Flatten methods is notably, as the *FlattenCNN* gets better results than the *FlattenLSTM*. Even so, the values show a high degree of overfitting. While the Hierarchical models show much less overfitting, it looks like the Flatten models may get better results solving this problem.

Table 1: Train and test accuracy achieved by the best version of each model

Models	Train Accuracy	Test Accuracy
FlattenCNN	0.79	0.45
FlattenLSTM	0.61	0.12
HierarchicalCNN	0.27	0.27
HierarchicalLSTM	0.17	0.09
HierarchicalMix	0.09	0.09

Accuracy per class

As our dataset is highly umbalanced (there are much more ocurrences of *Aves* than of any other), the raw accuracy is not an adequate measure of how good the models are. For further testing of the models, we build a confusion matrix for the *class* level, to check whether or not this models solve with enough accuracy the problem. The measure we take into account is the average of *Recall*. Note that in our test set there were no *Arachnida*, *Actinopterygii* and *Reptilia*, as the ocurrences of those classes were lower, and thus did not appear in the test set.

Table 2: Confusion matrix of the test dataset on the *FlattenCNN* model

Class	Arachnida	Insecta	Actinopterygii	Amphibia	Aves	Mammalia	Reptilia	Recall
Arachnida	0	0	0	0	0	0	0	-
Insecta	0	13	0	0	17	3	0	0.39
Actinopterygii	0	0	0	0	0	0	0	-
Amphibia	0	0	0	0	13	2	0	0.87
Aves	0	6	1	0	2552	189	0	0.93
Mammalia	0	3	0	0	134	296	0	0.68
Reptilia	0	0	0	0	0	4	0	0.00

From the *Recall* of *FlattenCNN* (0.57) in the test set we can say that it is quite good as it does not classify everything as *Aves*, but it is able to classify properly many different *classes*.

Conclusions

- The dataset was created and the data augmentation applied, although despite this, we suffer from overfitting.
- Classifiers achieved decent results down to *family* level, needing to remove *order* level in the Hierarchical models.
- We were able to compare the capabilities of CNNs and LSTMs for solving audio classification problems with Flatten models. Discovering that for our case CNNs are better.
- We got worse results in accuracy for Hierarchical models, but not enough training has been done yet with this models, due the complexity and time they require to train and optimize the hyperparameters.

Next steps

- Keep training the Hierarchical models to try to achieve at least an accuracy close to the flatten methods.
- Use weights to give more relevance to the top layers classifiers in the Hierarchical model, so the outputs takes more into account the previous decission in the tree.
- Try to do undersampling over the *Aves* data so we get more balanced labels in an attempt to reduce more the overfitting of our models.

References

[1] M. für Naturkunde Berlin. Animal sound archive. *Global Biodiversity Information Facility*, 2016. URL https://doi.org/10.15468/0bpalr.

[2] D. Gerhard. Audio signal classification: History and current techniques. 2003.

[3] S. K. Pandey, H. S. Shekhawat, and S. R. M. Prasanna. Deep learning techniques for speech emotion recognition: A review. In *2019 29th International Conference Radioelektronika (RADIOELEKTRONIKA)*, pages 1–6, 2019. doi: 10.1109/RADIOELEK.2019.8733432.

[4] C. N. J. Silla and A. x. Freitas. A survey of hierarchical classification across different application domains. doi: 10.1007/s10618-010-0175-9.

[5] E. Şaşmaz and F. B. Tek. Animal sound classification using a convolutional neural network. In *2018 3rd International Conference on Computer Science and Engineering (UBMK)*, pages 625–629, 2018. doi: 10.1109/UBMK.2018.8566449.

https://github.com/Pheithar/Animal-Sound