CONVOLUTIONAL RECURRENT NEURAL NETWORKS FOR BIRD AUDIO DETECTION

Emre Cakir, Sharath Adavanne, Giambattista Parascandolo, Konstantinos Drossos, Tuomas Virtanen

Department of Signal Processing, Tampere University of Technology

ABSTRACT

In this paper, we propose using convolutional recurrent neural networks on the task of bird audio detection in real-life environments. Different data augmentation, model ensemble and regularization methods are proposed for the present problem and evaluated in this regard. We evaluate our results using area under curve measure (AUC). Our best achieved AUC score on five fold cross-validation of the development data is 95.3% and 89.4% on the unseen evaluation data.

***Index Terms*—** Bird audio detection, convolutional recurrent neural network

1. INTRODUCTION

Bird audio detection (BAD) is defined as identifying the presence or absence of bird call/tweet in a given audio recording. This task acts as a preliminary step in the automatic monitoring of biodiversity. Post identifying the presence of bird call activity, a species based classifier can recognize the bird call more accurately. In this regard, the bird audio detection challenge [1] was organized with an objective to create robust and scalable algorithms which can work on real life bioacoustics monitoring projects without any manual intervention. The challenge provided annotated and non-annotated bird call recordings. The former were selected from a wide range of field and crowd-sourced recordings and is utilized as the training dataset. The latter are recordings from a completely different geographical location and employed as the test dataset.

Convolutional neural networks (CNN) are able to extract higher level features that are invariant to local spectral and temporal variations. Recurrent neural networks (RNNs) are powerful in learning the longer term temporal context in the audio signals. CNNs and RNNs as classifiers have recently shown improved performances over established methods in various sound recognition tasks. We combine these two approaches in a Convolutional Recurrent Neural Network (CRNN) and apply it on a bird audio detection detection task. CRNN has provided state-of-the-art results on various polyphonic sound event detection and audio tagging tasks [2].

The rest of the paper is organized as follows. Acoustic features representing the harmonic and non-harmonic content of the audio used in our BAD system are discussed in Section 2. The proposed CRNN and its configuration for the BAD is explained and presented in Section 3. Dataset setings, metrics and the results are reported in Section 4.

2. FEATURES

In this work, we experiment with log mel-band energy features, which has been shown to be effective in various audio tagging and sound event detection tasks. First, magnitude spectrum of the audio signals have been obtained using short-time Fourier transform (STFT) over 40 ms audio frames with 50% overlap using Hamming window and 1024 bins. Duration of each audio file in the challenge dataset is 10 seconds, resulting with 500 frames for each file. Then, 40 log mel-band energy features have been extracted from the magnitude spectrum. Keeping in mind that bird sounds are often contained in a relatively small portion of the frequency range, it makes theoretical sense to focus on extracting features from that range. However, experiments with features from the whole frequency range provided better results, and therefore utilized in the proposed method. Librosa library [3] has been used in feature extraction process.

3. CONVOLUTIONAL RECURRENT NEURAL NETWORKS

The CRNN proposed in this work, depicted in Figure 1, consists of four parts: (1) at the top of the architecture, a time-frequency representation of the data (40 log mel-band energies over 500 frames) is fed to 4 convolutional layers with 96 feature maps, 5-by-5 filters with rectified linear unit (ReLU) activations and non-overlapping pooling over frequency axis (pooling sizes are [5,2,2,2], respectively); (2) the feature maps of the last convolutional layer are stacked over the frequency axis and fed to 2 gated recurrent unit (GRU) [4] layers with 96 hidden units; (3) a temporal max-pooling layer is applied over the extracted representation over each time frame (4) a

The research leading to these results has received funding from the European Research Council under the European Unions H2020 Framework Programme through ERC Grant Agreement 637422 EVERYSOUND. Part of the computations leading to these results were performed on a TITAN-X GPU donated by NVIDIA. The authors also wish to acknowledge CSC-IT Center for Science, Finland, for computational resources.

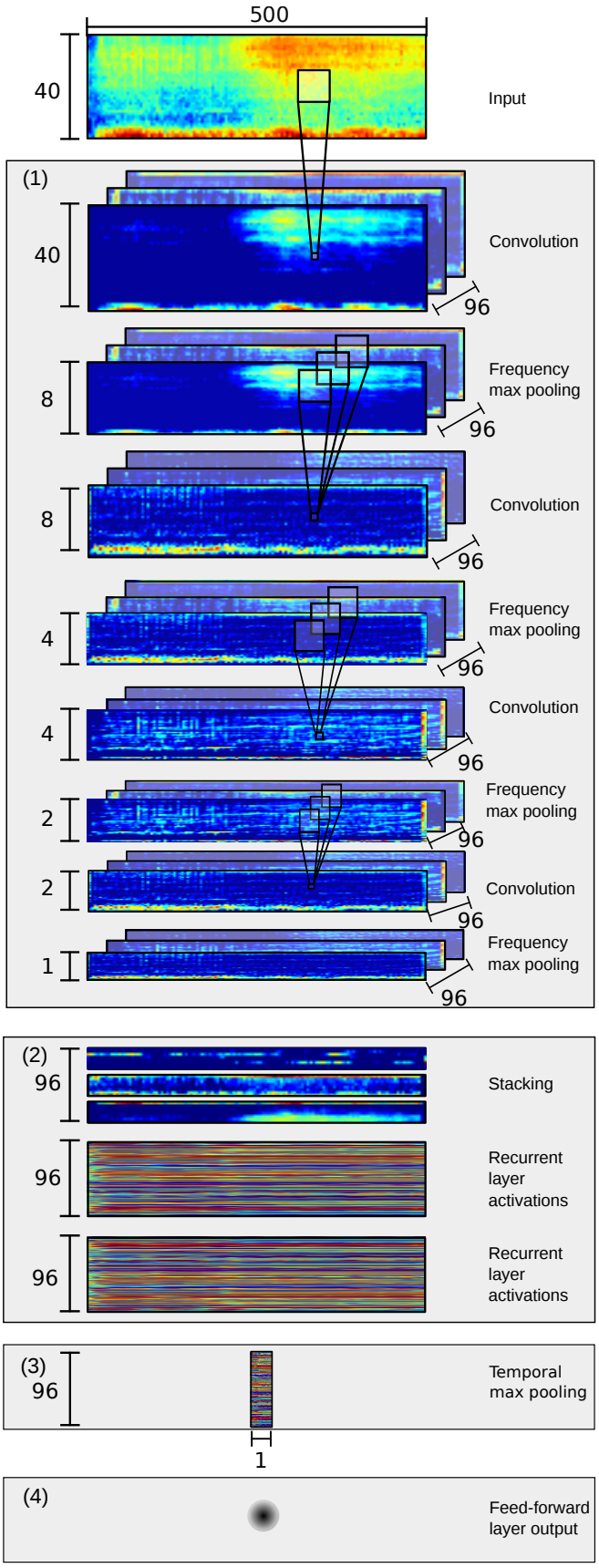

Fig. 1. System overview of CRNN architecture proposed for bird audio detection.

Dataset	Bird call		
	Present	Absent	Total
freefield1010	5755	1935	7690
warblr	1955	6045	8000
chernobyl	?	?	8620
Total	7710 + ?	7980 + ?	24310

Table 1. Bird audio detection challenge [1] dataset statistics

single feedforward layer with 1 unit and sigmoid activation reads the final recurrent layer. The sigmoid activation from the feedforward layer is treated as the bird audio probability for the audio file.

The network is trained with back-propagation through time [5] using Adam optimizer [6] and binary cross-entropy as the loss function. In order to reduce overfitting of the model, early stopping was used to stop training if the area under curve (AUC) measure (Section 4.2) did not improve for 50 epochs. For regularization, batch normalization [7] and dropout [8] with rate 0.25 were employed in convolutional layers. Keras deep learning library [9] has been used to implement the proposed network.

4. EVALUATION AND RESULTS

4.1. Datasets

The bird audio detection challenge [1] consists of a development and an evaluation set. The data comes from three separate datasets: i) field (freefield1010), ii) crowd-sourced (warblr), and iii) remote monitored (chernobyl). While the development set consists of freefield1010 and warblr only, the evaluation set comprises of data unseen in development, predominantly coming from the chernobyl dataset. Recordings in both the sets are 10 seconds long, mono channel and sampled at 44.1 kHz. The labels for the development set are binary - bird calls present or absent. The statistics of the datasets are presented in Table 1.

From the development set, we generate five cross validation (CV) splits of 60% training, 20% validation, and 20% testing. Each split had an equal distribution of birds call present and absent. All development set results in future are the average performance on this five CV split. For the challenge submission, the CRNN is trained on single CV split of 80% training and 20% validation done on development set, with equal distribution of classes. Also for the challenge submission, 11 networks with the same architecture and different initial random weights (obtained by sampling from different random seeds) have been trained. The estimated probabilities for the test audio files from each network has been averaged to obtain the ensemble output.

Dataset	Method	
	CRNN	CRNN + Ensemble
Development	95.3	
Evaluation	88.3	89.4

Table 2. AUC scores on development and evaluation sets

4.2. Metrics

The BAD system output is evaluated from the receiver operating characteristic (ROC) using the area under curve (AUC) measurement [10].

4.3. Results

AUC scores for development and evaluation sets are presented in Table 2. AUC for development set is obtained from the mean AUC of the five-fold CV test data.

5. REFERENCES

[1] "Bird audio detection challenge," 2016. [Online]. Available: http://machine-listening.eecs.qmul.ac.uk/bird-audio-detection-challenge/

[2] E. Cakir, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," in *IEEE/ACM TASLP Special Issue on Sound Scene and Event Analysis*, 2017, accepted for publication.

[3] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th Python in Science Conference*, 2015.

[4] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8)*, 2014.

[5] P. J. Werbos, "Backpropagation through time: what it does and how to do it," in *Proceedings of the IEEE*, 1990.

[6] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *arXiv:1412.6980 [cs.LG]*, 2014.

[7] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015.

[8] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," in *Journal of Machine Learning Research (JMLR)*, 2014.

[9] F. Chollet, "Keras," https://github.com/fchollet/keras, 2015.

[10] "Area under curve." [Online]. Available: https://en.wikipedia.org/wiki/Receiver_operating_characteristic#Area_under_curve