Animal Sound Classification Using A Convolutional Neural Network

Emre Şaşmaz
Department of Computer Engineering
Işık University
İstanbul, Turkey
emre.sasmaz@isik.edu.tr

F. Boray Tek
Department of Computer Engineering
Işık University
İstanbul, Turkey
boray.tek@isikun.edu.tr

***Abstract*—In this paper, we investigate the problem of animal sound classification using deep learning and propose a system based on convolutional neural network architecture. As the input to the network, sound files were preprocessed to extract Mel Frequency Cepstral Coefficients (MFCC) using LibROSA library. To train and test the system we have collected 875 animal sound samples from an online sound source site for 10 different animal types. We report classification confusion matrices and the results obtained by different gradient descent optimizers. The best accuracy of 75% was obtained by Nesterov-accelerated Adaptive Moment Estimation (Nadam).**

***Keywords*—Animal sound classification, Mel Frequency Cepstral Coefficient (MFCC), Convolution Neural Network (CNN), Confusion Matrix (CF)**

I. Introduction

Deep learning based methods are successfully applied to many pattern recognition problems such as object detection, speech recognition. Recognizing environmental sounds is an important problem from pattern recognition perspective because artificial intelligence is becoming more important for protecting biodiversity and conversation [1] [2].

In Piczak [3] study, the potential of convolutional neural networks (CNN) in a classification of environmental sounds was investigated using a deep model that consists of 2 convolutional layers with max pooling and 2 fully connected (FC). Accuracy was evaluated on 3 different public environmental datasets and urban recordings. Given the limited datasets, they evaluated whether convolutional neural networks could be successfully applied to environmental voice classification tasks. They have reported that CNN using log Mel-spectrograms performed best with the overall accuracy of 64.5%.

Valenti, Diment, and Parascandolo [4] presented an acoustic scene classification (ASC) study that proposes detection and classification of acoustic scenes and events. They use a convolutional neural network (CNN) by using short sequences of audios. They obtained a 79.0% accuracy on DCASE 2016 development dataset. Their CNN model consists of two-layer convolutional and max-pooling layer. A two-layer convolution network can achieve higher accuracy when compared to a two-layer multi-layer perceptron (MLP), a single-layer CNN, and a Gaussian mixture models-Mel Frequency Cepstral Coefficient (GMM-MFCC) system.

Mane, Rashmi, and Tade [5] presented animal sound classification and retrieval method for bioacoustics and audio retrieval applications. They use ZeroCross-Rate (ZCR), Mel-Frequency Cepstral Coefficients (MFCC), Dynamic Time Warping (DTW) algorithms jointly to recognize the voice of the particular animal. The results also show that different animal states such as normal, hunger, sleep, heat could be recognized.

In this study, we investigate an animal identification system from a deep learning perspective. Deep learning networks usually require huge datasets to learn thousands or millions of parameters. Unfortunately, there is no ready-made animal sound samples dataset in the literature. Thus we had to construct our own dataset which is made public with this paper online [6]. Our dataset consists of 875 sound samples of 10 different animal classes which are all collected from [7] sound source site. The number of samples per-animal class is given in Table I. For implementation, we used Keras based on Tensorflow and a practical sound recognition library "LibROSA" which is a python comprehensive package for music and audio analysis [8]. We computed Mel Frequency Coefficients (MFCC) using LibROSA library. MFCC [9] is a commonly used feature representation for acoustic modeling. We construct a deep convolutional architecture which consists of convolution, max-pooling, and dropout layers to classify the features into animal classes.

TABLE I
Number of Samples per-animal class

Animals	Number of Sounds2
Bird	200
Cat	200
Dog	200
Cow	75
Lion	45
Sheep	40
Frog	35
Monkey	30
Chicken	25
Donkey	25

The rest of the paper is organized as follows. In Section II, we describe the animal sound samples dataset, describe the learning architecture and feature extraction. Experimental

978-1-5386-7893-0/18/$31.00 ©2018 IEEE

work such as training, tests, and results are given in Section III. The conclusion and future work are given in Section IV.

II. Methodology

In this section, we first introduce the dataset. Then we explain the feature extraction pre-process. Then, we introduce the learning architecture.

A. Dataset

Since we do not have an audio data set ready for use we have constructed our own dataset. We have collected animal voices in WAV sound format from an internet site [7]. These audio files were cleaned and used in 3 different ways. Firstly, the audio files were collected in 1050 pieces. Each audio file was individually manually examined to prevent errors. Some audio files were removed because they contained dominant noise or because the animal sound level was very low. Secondly, type of all sounds is WAV because helper tools (MFCC) that we will use in feature extraction stage support WAV sound format. Thirdly, the size of the audio files has been set. The files were brought to at least 3 kilobytes in size. If it is less than 3 kilobytes, the data could not be processed. No upper limit was set. After these 3 eliminations, 175 audio files were deleted. Our final dataset consisted of 875 sound sample of 10 different animal classes which is published online [6]. The number of samples per-animal class is given in Table I.

B. Feature Extraction

One can represent raw audio using different techniques [10]. Sound samples are time-varying and, each sample has non-local dependencies. Hence, though it has been tried before, learning from raw audio samples can be difficult [11]. In this work, due to the limited dataset, we use a commonly used lossy representation system called Mel Frequency Cepstral Coefficient which models the shape of sound frequency spectrum [9]. By using LibROSA, all sound files in the dataset were preprocessed, features were extracted and saved as binary files to be used in training/tests.

C. Network

The deep learning network consists of three convolution layers with a max pooling layer, and three dense layers (aka fully connected layer). Three convolution layers and two dense layers use rectified linear units (ReLU) as the activation function. The final dense layer uses softmax as the activation function. In convolution and max-pooling layers, 2x2 kernel size was used as shown in Figure 1.

III. Experiments

For this study, experiment setup consists of three subsections which are Feature Extraction, Training, and Test. The data was divided into two subsets at random. Also, all data split to train and test part. For this splitting, train data selected 80% and test data selected 20% of all data. We use a randomly selected 20% percent of the training set used as the validation set to tune learning hyperparameters.

A. Training

In training, we first loaded the extracted features and labels of classes. Then, we have used a util (”to_categorical”) of Keras to convert the array of labeled data(from 0 to the number of classes - 1) to one-hot vector. The designed CNN model was trained with given hyper-parameters in Table II. Additionally, the trained model was saved to use it later in the test stage. In training, we observed that increasing training iteration (i.e. epoch) affects the overall classification accuracy positively for both training and validation sets as illustrated in Figure 2 3 4 for different optimizers. This training/validation was repeated many times to manually search the most appropriate hyper-parameters for the model. As it can be seen in Figure 2 3 4 you can clearly see that validation accuracy has not changed after 100th or more epochs. At that rate, we can say the network is overfitting or overtraining.

TABLE II
Hyper parameters of CNN model

Key	Value
epochs	100
batch_size	100
verbose	1
optimizers	Adadelta, Adagrad, Adamax, Adam, Nadam, RMSprop, SGD

B. Test

In the test stage, we expect the CNN model to classify an unknown sound file into one of the animal classes. As in the training the sound files were converted to MFCC. In order to evaluate the performance of the classification models, a confusion matrix can be used. The matrix summarizes the classification errors in terms of target and the predicted labels [12]. The diagonal entries in the matrix show correct classification whereas off-diagonals are incorrect, see Figures 5 and 6. As an example among 43 birds, our model identified 36 bird sounds like birds. Four bird sounds were incorrectly classified as cats. Another observation is that dog sounds were mostly correctly classified class, most probably due to its number of training instances being larger. It can be seen that imbalanced data distribution affected the results such that the accuracy percentage in the less populated animal classes were poorer.

We have repeated the same training and tests to compare different hyper-parameters. We have seen the batch size of 100 was the best. Batch size defines the number of samples that going to be propagated through the network. Training more than 100 epochs did not improve validation results. If we consider the performance of different optimizers, the best was obtained using NADAM [13], whereas Stochastic gradient descent (SGD) performed worse Table III. The learning rate was set as 0.01.

IV. Conclusions

In this paper, we proposed a system which classifies given animal sound audio files into animal classes. We proposed

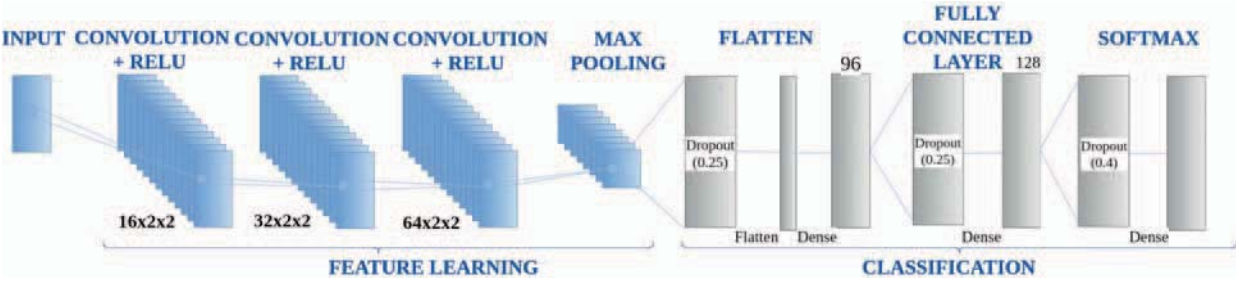

Fig. 1. CNN Architecture

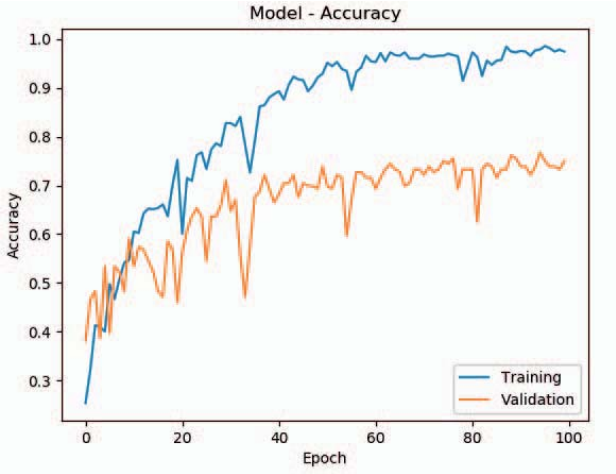

Fig. 2. Nadam Epoch - Accuracy Chart

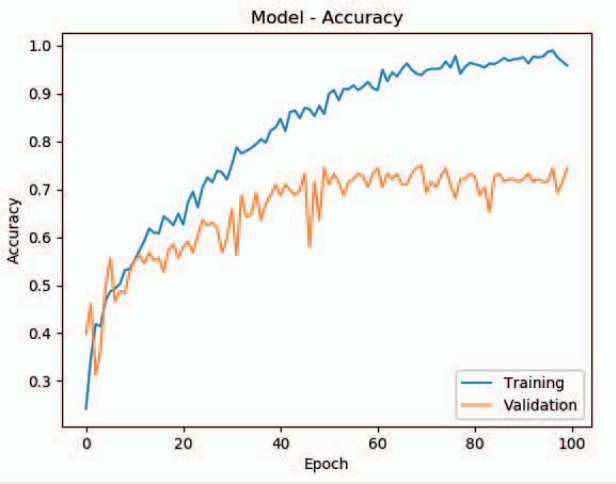

Fig. 3. Adadelta Epoch - Accuracy Chart

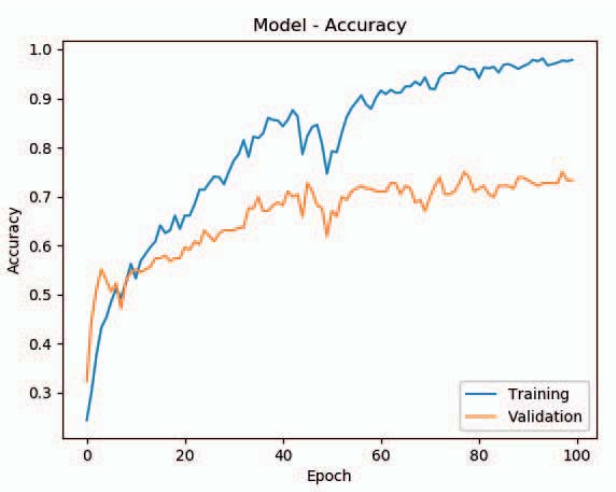

Fig. 4. Adam Epoch - Accuracy Chart

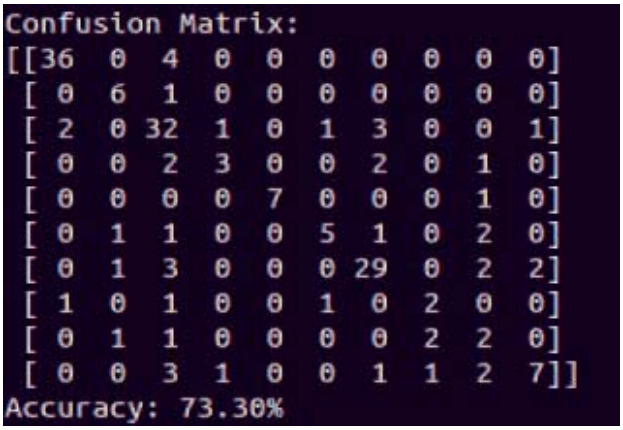

Fig. 5. Adam Optimizer Confusion Matrix

a system which is on a deep convolutional neural network architecture where MFCC features were used as the input. As an input to the network, sound files were preprocessed to extract. To train and test the system we have collected 875 animal sound files from online sound source site [7] of 10 different animal types. The classification confusion matrices showed that the accuracies were affected by the class imbalance. We compared different optimizers and the best accuracy of 75% was obtained by Nesterov-accelerated Adaptive Moment Estimation.

It is well known that deep learning requires a large number of samples to be efficient. The low number of training samples in our study had a negative effect on the recognition performance. In a future work, we would try increasing number of samples and use unsupervised data augmentation to increase

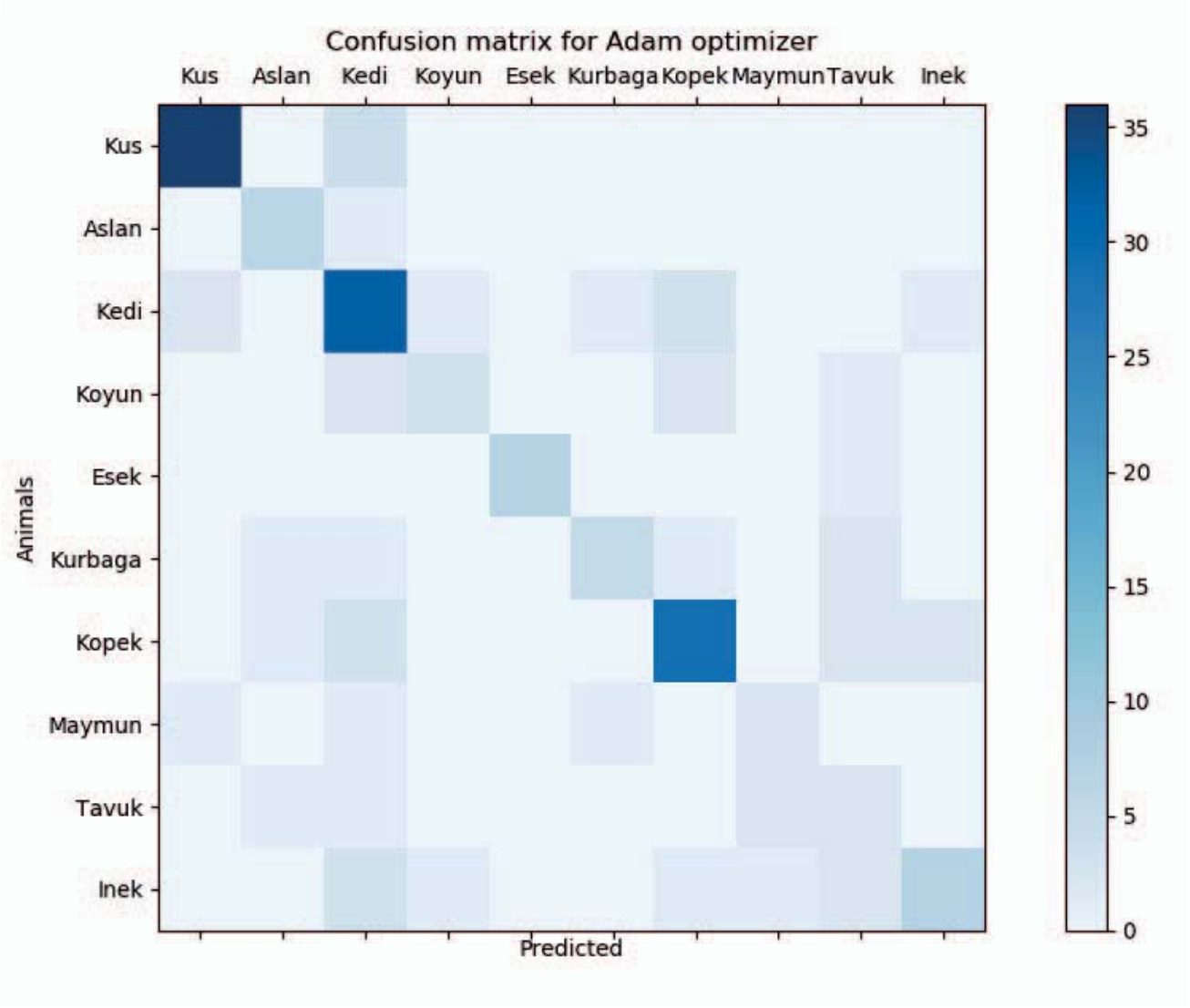

Fig. 6. Adam Optimizer Confusion Matrix on Test Set

TABLE III
RESULTS WITH RESPECT OPTIMIZERS

Optimizer	Accuracy
Nadam	**75.00**
Adadelta	74.43
Adam	73.30
Adagrad	72.73
Adamax	72.73
RMSprop	71.59
SGD	60.23

the recognition performance.

REFERENCES

[1] C. Herweijer, D. Waughray, "Harnessing Artificial Intelligence for the Earth," Fourth Industrial Revolution for the Earth Series, Jan 2018

[2] F. B. Tek, F. Cannavo, G. Nunnari, I. Kale, "Robust localization and identification of African clawed frogs in digital images," Ecological Informatics, vol 23, pp 3-12, Sept 2014

[3] K. J. Piczak, "Environmental sound classification with convolutional neural networks.," Machine Learning for Signal Processing (MLSP), 2015 IEEE 25th International Workshop on. IEEE, Sept 2015.

[4] M. Valenti, A. Diment, G. Parascandolo, S. Squartini, T. Virtanen, "DCASE 2016 acoustic scene classification using convolutional neural networks," Detection and Classification of Acoustic Scenes and Event, Sept 2016

[5] A. D. Mane, Rashmi R. A., S. L. Tade. "Identification & Detection System for Animals from their Vocalization," International Journal of Advanced Computer Research, Vol. 3, Issue 11, No 3, Sept 2013

[6] "Animal-Sound-Classification-Using-A-Convolutional-Neural-Network" [Online]. https://github.com/emresasmaz/Animal-Sound-Classification-Using-A-Convolutional-Neural-Network

[7] "THE Source for Free Sound Files and Reviews," [Online]. Available: http://www.wavsource.com/

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

[9] L. Rabiner, B. H. Juang, "Fundamentals of speech recognition," Englewood Cliffs: PTR Prentice Hall, Vol. 14, 1993

[10] L. Wyse, "Audio Spectrogram Representations for Processing with Convolutional Neural Networks," Proceedings of the First International Workshop on Deep Learning and Music joint with IJCNN, Anchorage,

US. May, 2017. 1(1). pp 37-41
[11] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. “WAVENET: A Generative model for raw audio,” CoRR, abs/1609.03499, 2016. Available: https://arxiv.org/abs/1609.03499.
[12] A. K. Santra, C. Josephine Christy. “Genetic Algorithm and Confusion Matrix for Document Clustering,” IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 2, January 2012
[13] Timothy Dozat. “Incorporating Nesterov Momentum into Adam,” Incorporating Nesterov Momentum into Adam, paper 107, Feb 2016