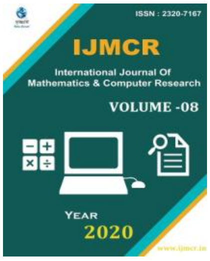
International Journal of Mathematics and Computer Research

ISSN: 2320-7167

Volume 08 Issue 07 July 2020, Page no.- 2066-2070
Index Copernicus ICV: 57.55
DOI: 10.33826/ijmcr/v8i7.01



The Sound Classification System Based on Neural Networks with Attention Mechanism and Autoencoders

Anastasiia Kryvokhata

Department of Software Engineering, Zaporizhzhya National University, Zaporizhzhya, Ukraine

ARTICLE INFO

Published Online:
02 July 2020

Corresponding Author:
Anastasiia Kryvokhata
Zaporizhzhya National University, Zhukovsky str., 66, Zaporizhzhya, 69600, Ukraine
+380972100376

ABSTRACT

This paper discusses the models and methods of machine learning methods used in environmental sound classification systems. A sound classification subsystem could be implemented in the various Smart City, Smart Farming systems, healthcare etc. An automated sound classification system could be decomposed into following subsystems: the audio representation, the features extraction, the classifier and the accuracy estimation. This paper deals with a convolutional neural network with attention mechanism for sound classification and autoencoder for data augmentation. The objective of the paper is to develop the optimal sound classifier for datasets with lack of observations.

KEYWORDS: Sound Classification, Convolutional Neural Network, Autoencoder, Attention.

I. INTRODUCTION

Machine hearing systems [1] are quite useful in such areas as audio surveillance, automatic medicine auscultation, acoustic scene detection in urban environments etc. The main requirements for such systems are real time classification with high accuracy and ability to generalization on unseen data. Environmental sound classification differs from music or speech recognition by higher data variability and lack of frequent pattern repetition.

There are different approaches to environmental sound classification but most suitable method here is neural network classifier. The benefits of using neural networks are as follows. Generalization to new data allows classifying new sounds using a pre-trained model. Deep learning methods allow extracting features from raw sound, so there is no need in additional sound preprocessing modules. However, a large amount of training data is required for neural networks training that is disadvantage. This can be overcome by using autoencoders for data augmentation. There are different approaches for accelerating learning process and improving accuracy: meta-learning, evolutionary algorithms etc. Particularly, attention mechanism for neural networks has been used successfully in recent articles to improve the classifier accuracy. Therefore, our goal is to develop environment sound classification system using autoencoder for data augmentation and deep convolutional neural network with attention mechanism as classifier.

II. THE AIM AND OBJECTIVES OF THE STUDY

There are several essential steps in sound classification systems, we consider following subsystems: audio data preprocessing, features extraction, classification, and accuracy estimation. The audio data preprocessing stage implies different approaches such as segmentation, filtering, transformations. Equal sound files could be obtain by segmentation of the raw signal into shorter chunks using some windowing process. Thus, the original acoustic signal is converted into the frames of a fixed length. Trailing silence frames from an audio signal could be done using sound energy filters.

The feature extraction stage includes methods for different representations of the initial digital signal. This stage exploits special features such as Mel Frequency Cepstral Coefficients (MFCC), Gammatone Mel Frequency Cepstral Coefficients (GFCC), chromagrams and spectrograms. The aim of this stage is to find better signal characteristic for further classification.

The State-of-the-art methods for sound classification are neural networks and other machine learning methods like K-means, support vector machine (SVM), decision trees etc [2-4].The classifiers based on convolutional or recurrent neural networks are quite effective in these problems. There are different approaches for neural networks fine tuning, for instance, meta-learning, genetic algorithms and special layers like attention mechanism.

The aim of this paper is to develop a proof-of-concept system for the environmental sound classification based on convolutional neural networks and autoencoders. We suggest that autoencoders could be used for data augmentation and convolutional model could be improved by attention layer.

In order to reach the mentioned aim, the following objectives were formulated for the study:

– to review state-of-the-art environmental sound classification systems;

– to develop autoencoder model for data augmentation;

– to develop classification system using convolutional neural networks with attention layer;

– to outline a direction for further development of machine hearing systems.

We going to use ESC-50 dataset for training and testing classification model. This dataset consists of 5-second-long recordings organized into 50 semantical classes with 40 examples per class [5].

III. Literature Review

There are numerous survey papers have published recently on the topic of sound classification, for instance, articles [2, 3] provide a description of the components of an automatic sound classification system, which contains preprocessing modules, feature extraction, training algorithm and evaluation module.

The signal feature extraction methods are discussed in details [2]. The physical properties of the signals and the characteristics of the human perception of sounds are two basic category here. Feature extraction methods represent the acoustic signal in the time, frequency, cepstral and wavelet domains.

The analysis of general approaches to automatic music classification by genre is described in [3, 6].

The methods of neural networks application in the feature extraction and the classification are given in [1, 6, 7]. Deep convolutional neural networks are the main classifier in this area, which is explained by the complex layered architecture with several layer types. The disadvantages of this approach include the complexity of neural networks fine tuning, demand for computing resources and large amount of observations in the learning datasets.

Attention mechanism is described in [8-10]. This approach demonstrates high accuracy and become an essential part of neural network classification architectures.

IV. Methodology

The workflow of automated environmental sound classification system has shown in the figure 1.

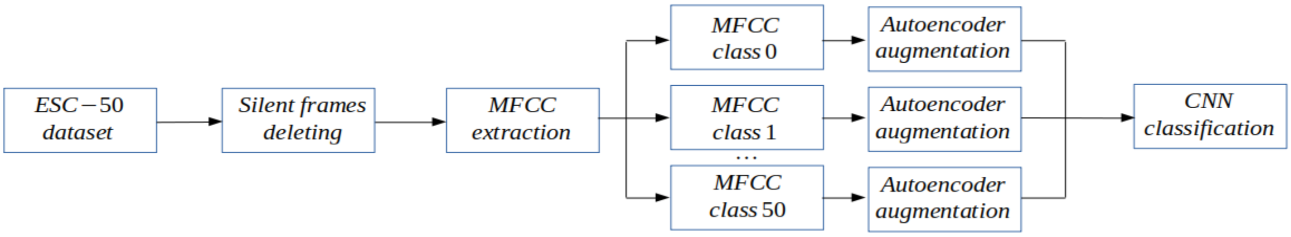
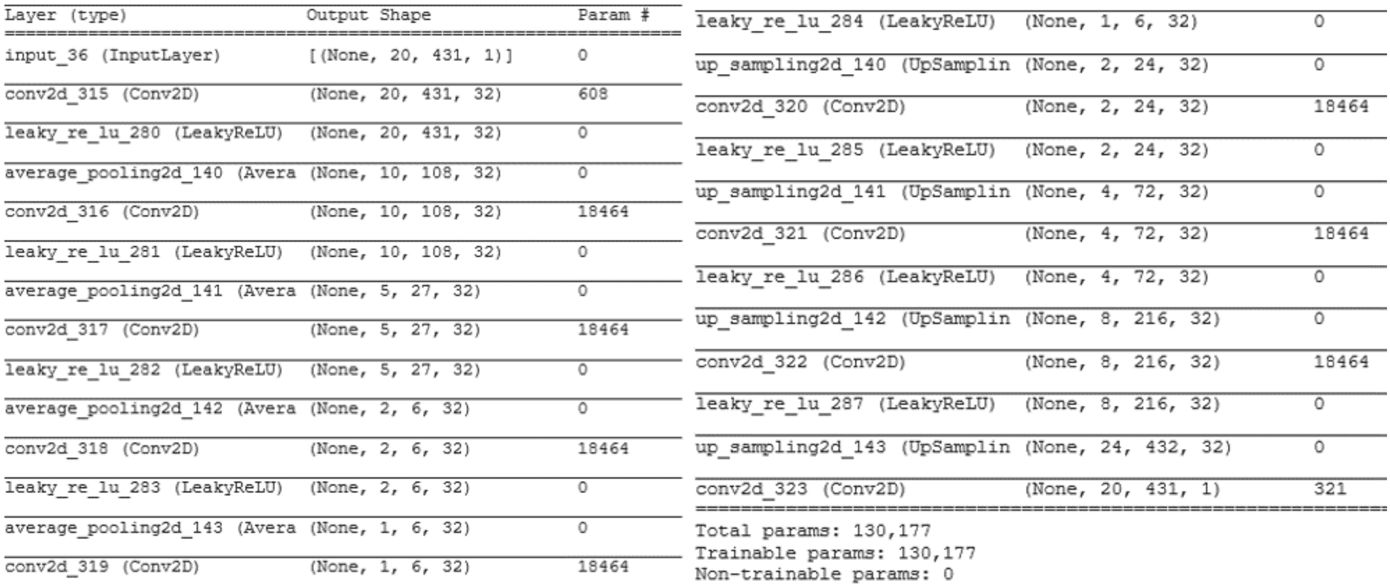


Figure 1: Sound classification system workflow



```
Layer (type)                 Output Shape              Param #
=================================================================
input_36 (InputLayer)        [(None, 20, 431, 1)]      0
_________________________________________________________________
conv2d_315 (Conv2D)          (None, 20, 431, 32)       608
_________________________________________________________________
leaky_re_lu_280 (LeakyReLU)  (None, 20, 431, 32)       0
_________________________________________________________________
average_pooling2d_140 (Avera (None, 10, 108, 32)       0
_________________________________________________________________
conv2d_316 (Conv2D)          (None, 10, 108, 32)       18464
_________________________________________________________________
leaky_re_lu_281 (LeakyReLU)  (None, 10, 108, 32)       0
_________________________________________________________________
average_pooling2d_141 (Avera (None, 5, 27, 32)         0
_________________________________________________________________
conv2d_317 (Conv2D)          (None, 5, 27, 32)         18464
_________________________________________________________________
leaky_re_lu_282 (LeakyReLU)  (None, 5, 27, 32)         0
_________________________________________________________________
average_pooling2d_142 (Avera (None, 2, 6, 32)          0
_________________________________________________________________
conv2d_318 (Conv2D)          (None, 2, 6, 32)          18464
_________________________________________________________________
leaky_re_lu_283 (LeakyReLU)  (None, 2, 6, 32)          0
_________________________________________________________________
average_pooling2d_143 (Avera (None, 1, 6, 32)          0
_________________________________________________________________
conv2d_319 (Conv2D)          (None, 1, 6, 32)          18464
_________________________________________________________________
leaky_re_lu_284 (LeakyReLU)  (None, 1, 6, 32)          0
_________________________________________________________________
up_sampling2d_140 (UpSamplin (None, 2, 24, 32)         0
_________________________________________________________________
conv2d_320 (Conv2D)          (None, 2, 24, 32)         18464
_________________________________________________________________
leaky_re_lu_285 (LeakyReLU)  (None, 2, 24, 32)         0
_________________________________________________________________
up_sampling2d_141 (UpSamplin (None, 4, 72, 32)         0
_________________________________________________________________
conv2d_321 (Conv2D)          (None, 4, 72, 32)         18464
_________________________________________________________________
leaky_re_lu_286 (LeakyReLU)  (None, 4, 72, 32)         0
_________________________________________________________________
up_sampling2d_142 (UpSamplin (None, 8, 216, 32)        0
_________________________________________________________________
conv2d_322 (Conv2D)          (None, 8, 216, 32)        18464
_________________________________________________________________
leaky_re_lu_287 (LeakyReLU)  (None, 8, 216, 32)        0
_________________________________________________________________
up_sampling2d_143 (UpSamplin (None, 24, 432, 32)       0
_________________________________________________________________
conv2d_323 (Conv2D)          (None, 20, 431, 1)        321
=================================================================
Total params: 130,177
Trainable params: 130,177
Non-trainable params: 0
```

Figure 2: Autoencoder architecture

We use MFCC as features for further classification but there are only 40 5-second-long observations per class, so some method of data augmentation is required here. Thus, for 2000 initial observations we obtain 2000x20x431 matrix representation for 20 Mel Frequency Cepstral Coefficients per each sound file. Autoencoder could be used for generating new sounds for each class. The convolutional autoencoder architecture has shown in the figure 2.

Keras library is used for neural network implementation. We use Conv2D layers for convolutions and AveragePooling layers for dimension reduction and Leak ReLu is use as activation function.

The loss functions for different autoencoders classes change during learning process as has shown in the figure 3.

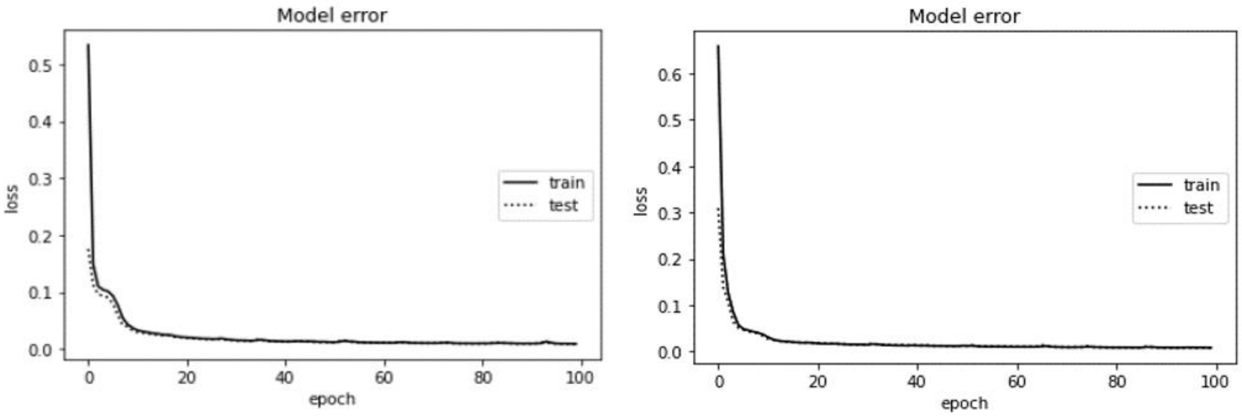
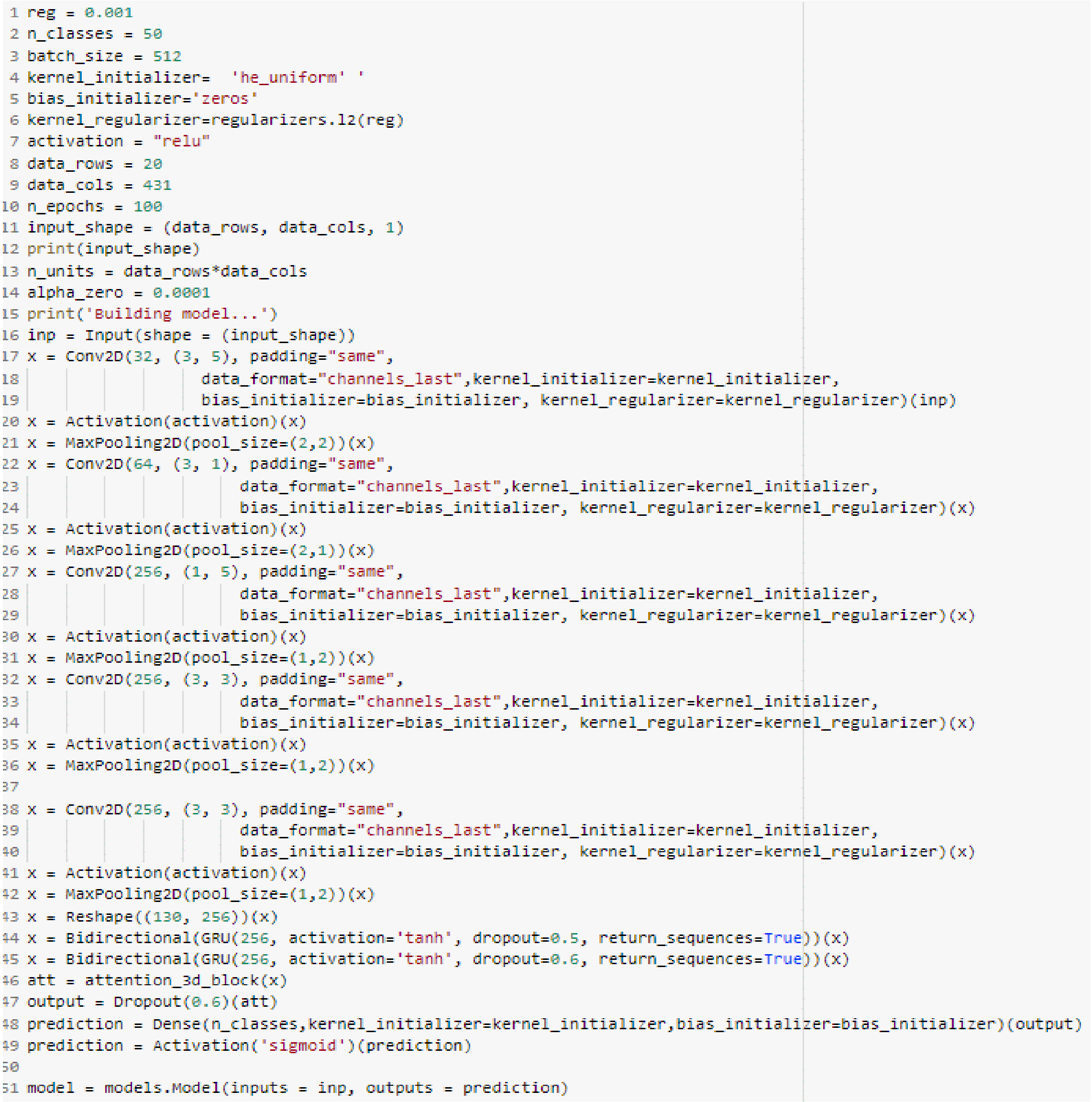


Figure 3: Autoencoders loss function



```
reg = 0.001
n_classes = 50
batch_size = 512
kernel_initializer=  'he_uniform' '
bias_initializer='zeros'
kernel_regularizer=regularizers.l2(reg)
activation = "relu"
data_rows = 20
data_cols = 431
n_epochs = 100
input_shape = (data_rows, data_cols, 1)
print(input_shape)
n_units = data_rows*data_cols
alpha_zero = 0.0001
print('Building model...')
inp = Input(shape = (input_shape))
x = Conv2D(32, (3, 5), padding="same",
                  data_format="channels_last",kernel_initializer=kernel_initializer,
                  bias_initializer=bias_initializer, kernel_regularizer=kernel_regularizer)(inp)
x = Activation(activation)(x)
x = MaxPooling2D(pool_size=(2,2))(x)
x = Conv2D(64, (3, 1), padding="same",
                      data_format="channels_last",kernel_initializer=kernel_initializer,
                      bias_initializer=bias_initializer, kernel_regularizer=kernel_regularizer)(x)
x = Activation(activation)(x)
x = MaxPooling2D(pool_size=(2,1))(x)
x = Conv2D(256, (1, 5), padding="same",
                      data_format="channels_last",kernel_initializer=kernel_initializer,
                      bias_initializer=bias_initializer, kernel_regularizer=kernel_regularizer)(x)
x = Activation(activation)(x)
x = MaxPooling2D(pool_size=(1,2))(x)
x = Conv2D(256, (3, 3), padding="same",
                      data_format="channels_last",kernel_initializer=kernel_initializer,
                      bias_initializer=bias_initializer, kernel_regularizer=kernel_regularizer)(x)
x = Activation(activation)(x)
x = MaxPooling2D(pool_size=(1,2))(x)

x = Conv2D(256, (3, 3), padding="same",
                      data_format="channels_last",kernel_initializer=kernel_initializer,
                      bias_initializer=bias_initializer, kernel_regularizer=kernel_regularizer)(x)
x = Activation(activation)(x)
x = MaxPooling2D(pool_size=(1,2))(x)
x = Reshape((130, 256))(x)
x = Bidirectional(GRU(256, activation='tanh', dropout=0.5, return_sequences=True))(x)
x = Bidirectional(GRU(256, activation='tanh', dropout=0.6, return_sequences=True))(x)
att = attention_3d_block(x)
output = Dropout(0.6)(att)
prediction = Dense(n_classes,kernel_initializer=kernel_initializer,bias_initializer=bias_initializer)(output)
prediction = Activation('sigmoid')(prediction)

model = models.Model(inputs = inp, outputs = prediction)
```

Figure 4: Convolutional neural network implementation

The program implementation of classifier has shown in the figure 4. The neural network hyperparameters like number of epochs, batch size, learning rate are determine empirically after numerical experiments.

The data preprocessing stage includes trailing silence from an audio signal and calculating of mel frequency cepstral coefficient (MFCC) for the giving sound files. This approach allows to unify and simplify the sound files presentation in the memory. Further we feed MFCC arrays to the convolutional neural net.

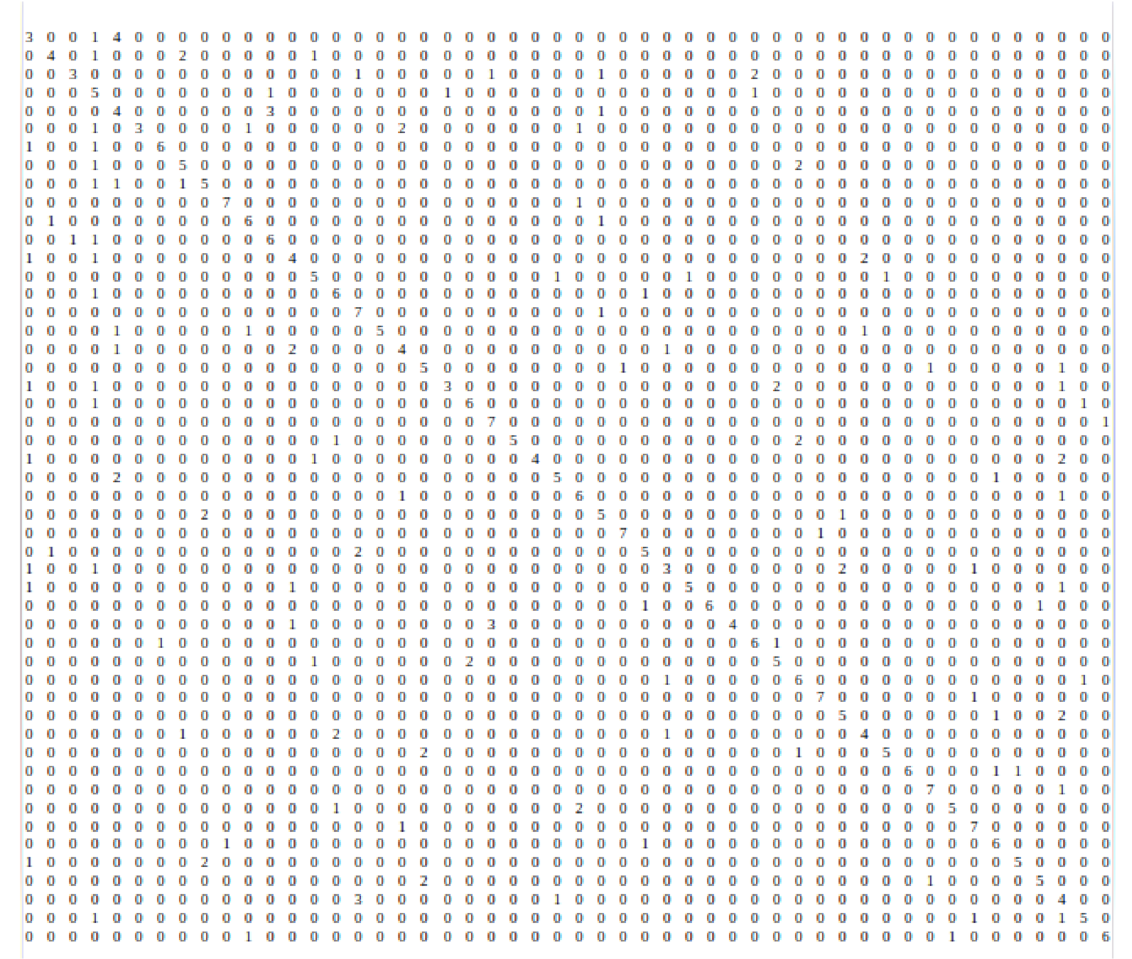
V. Numerical Results

The confusion matrix is most appropriate metric for model testing in sound classification. The testing dataset includes 8 raw sound files per each class from ESC-50 dataset. The resulting confusion matrix has been shown in the figure 5.

At the matrix each row and column means following categories:

0: {'dog'}, 1: {'rooster'}, 2: {'pig'}, 3: {'cow'}, 4: {'frog'}, 5: {'cat'}, 6: {'hen'}, 7: {'insects'}, 8: {'sheep'}, 9: {'crow'}, 10: {'rain'}, 11: {'sea_waves'}, 12: {'crackling_fire'}, 13: {'crickets'}, 14: {'chirping_birds'}, 15: {'water_drops'}, 16: {'wind'}, 17: {'pouring_water'}, 18: {'toilet_flush'}, 19: {'thunderstorm'}, 20: {'crying_baby'}, 21: {'sneezing'}, 22: {'clapping'}, 23: {'breathing'}, 24: {'coughing'}, 25: {'footsteps'}, 26: {'laughing'}, 27: {'brushing_teeth'}, 28: {'snoring'}, 29: {'drinking_sipping'}, 30: {'door_wood_knock'}, 31: {'mouse_click'}, 32: {'keyboard_typing'}, 33: {'door_wood_creaks'}, 34: {'can_opening'}, 35: {'washing_machine'}, 36: {'vacuum_cleaner'}, 37: {'clock_alarm'}, 38: {'clock_tick'}, 39: {'glass_breaking'}, 40: {'helicopter'}, 41: {'chainsaw'}, 42: {'siren'}, 43: {'car_horn'}, 44: {'engine'}, 45: {'train'}, 46: {'church_bells'}, 47: {'airplane'}, 48: {'fireworks'}, 49: {'hand_saw'}.

From this confusion matrix we can see that approach with autoencoder augmentation can lead to quite accurate classification. The maximum row element in the matrix is on the diagonal for most classes. Thus, there are 253 observations predicted as true positive from total 400, therefore, there is room for improvement the accuracy.



```
3 0 0 1 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 4 0 1 0 0 0 2 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 5 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 4 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 3 0 0 0 0 1 0 0 0 0 0 0 2 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 1 0 0 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 0 0 1 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 0 0 0 0 0 0 0 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 1 0 0 0 0 0 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0 0 0 2 0 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0
1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0
0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 0 0 0 2 0 0 0 0 0 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 6 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 6 0 0 0 0 0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 1 0 0 2 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 5 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 6 0 0 0 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 0 0 0 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 6 0 0 0 0 0
1 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 5 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 4 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 5 0
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 6
```

Figure 5: Sound classification confusion matrix

VI. Conclusion

A convolutional neural network with attention layer could be an effective classifier even in the case of train data lack. Convolutional auto encoder was used to generate new observations.

The basic distinction of this paper from papers [9] is that we tried to use a data set with only 40 observations per class. Obtained accuracy in 63% allows to continue research in this area.

The prospect of further research is related to the extension of the considered approach with meta-learning methods, for instance MAML or Reptile.

REFERENCES

1. Kong Q., Xu Y., Wang W. & Plumbley M.D. (2017). Convolutional gated recurrent neural network incorporate spatial features for audio tagging. The 2017 International Joint Conference on Neural Networks (IJCNN 2017), Anchorage, Alaska, DOI:https://doi.org/10.1109/IJCNN.2017.7966291.
2. Alias F., Socoro J.C. & Sevillano X. (2016). A review of physical and perceptual feature extraction techniques for speech, music and environmental sounds. Applied Sciences, № 6(5):143.
3. Bertin-Mahieux T., Eck D. & Mandel M. (2011). Automatic tagging of audio: the state-of-the-art. Machine audition: principles, algorithms and systems. Visnyk IGI Global, pp. 334–352.
4. Camastra F. & Vinciarelli A. (2015). Machime learning for Audio, Image and Video analysis. London: Springer-Verlag.
5. Piczak K.J. (2015). ESC: Dataset for Environmental Sound Classification. Proceedings of the 23rd Annual ACM Conference on Multimedia, Brisbane, Australia, DOI: http://dx.doi.org/10.1145/2733373.2806390.
6. Sturm B.L. (2014). A Survey of Evaluation in Music Genre Recognition. Adaptive Multi-media Retrieval: Semantics, Context, and Adaptation. Lecture Notes in Computer Science, Vol. 8382, pp. 29–66, DOI: https://doi.org/10.1007/978-3-319-12093-5_2.
7. Xu Y., Huang Q., Wang W., Foster P., Sigtia S., Jackson P.J.B & Plumbley M.D. (2017). Unsupervised Feature Learning Based on Deep Models for Environmental Audio Tagging. IEEE/ACM transactions on audio, speech and language processing, Vol. 25, Issue 6, pp. 1230–1241, DOI:https://doi.org/10.1109/TASLP.2017.2690563
8. Sharma J., Granmo O.-C. & Goodwin M. (2020). Environment Sound Classification using Multiple Feature Channels and Attention based Deep Convolutional Neural Network. Preprint arXiv.org, 13 p, URL: https://arxiv.org/abs/1908.11219.
9. Tang B., Li Y., Li X., Xu L. & etc. (2019). Deep CNN Framework for Environmental Sound Classification using Weighting Filters. International Conference on Mechatronics and Automation (ICMA 2019): Proceedings of 2019 IEEE International Conference. (Tianjin, 4-7 August 2019). Tianjin, China, pp. 2297-2302.
10. Zhang Z., Xu S., Qiao T. & etc. Attention based Convolutional Recurrent Neural Network for Environmental Sound Classification. Pattern Recognition and Computer Vision (PRCV 2019): Lecture Notes in Computer Science. (Xi’an, 8-11 November 2019). Vol. 11857, pp. 261-271. URL: https://arxiv.org/abs/1907.02230