ACOUSTIC SCENE CLASSIFICATION USING CONVOLUTIONAL NEURAL NETWORKS AND DIFFERENT CHANNELS REPRESENTATIONS AND ITS FUSION

Technical Report

Golubkov Alexander

Speechpro
golubkov-a@speechpro.com

Lavrentyev Alexander

Speechpro
lavrentyev@speechpro.com

ABSTRACT

Deep convolutional neural networks has great results in a image classification tasks. In this paper, we used different architectures of DCNN for image classification. As for images we used spectrograms of differenet signal representations, such as MFCC, Mel-spectrograms and CQT-spectrograms. Result was obtained using goemetric mean of all the models.

Index Terms— DCNN, CQT, MFCC, Mel

1. INTRODUCTION

Acoustic scene classification challenge focuses on investigation and explore new algorithms and approaches to detecting, classification and tagging environment audio. For the two past years a lot of different approaches were successfully used to win this challenge: HMM and GMM models, DCNN, RNN, LSTM and ensembling and fusion. For example, in this work of DCASE 2016 [1] authors used just one DCNN with CQT and Fourier ("standard") spectrograms. The have splitted spectograms into patches, then used DCNN on these patches and after all ensembled all the results.

Also, there are two different types of DCNN convolutions. First type, 2D-convolution, uses convolution blocks on all axes of spectrogram time and frequency. It is a good approach with good results and it is a good opportunity to use a lot of different pre-trained models. Second type, 1D-convolutions, uses one-axe convolution blocks only, in particular time axe. This approach saves all information about frequency in a signal, but convolve a time. It names a Time Distributed Convolution. It shows better results than 2D-convolutions. Next approach is using mix of Recurrent Neural Networks (LSTM cells) and DCNN (RCNN). The data from the next-to-last layer of DCNN is used as input data in LSTM networks. It is needed to notice, that only Time Distributed Convolution is used in this approach to reduce a time dimension of a spectrogram. And as always, it is good to ensmeble results of this models with DCNN models.

But the best results were shown by DCNN. These networks needed a big dataset to learn, and a standard image representations (channels and the same size). It is easy to achieve when using spectrogram representations of audio signals. DCASE task 1 challenge gives a large train/test dataset, including 8640 training examples. It is great information for train DCNNs using 1d- and 2d-convolutions and then ensembling them.

In this paper ensembling of diffrent models is shown, using 2d-convolutions with one and two inputs, and using 11 audio signal representations.

2. DATA PREPARATIONS

2.1. Acoustic Scenes Dataset

In this work, TUT Acoustic Scenes 2018 dataset [3] for training, testing and evaluation were used. The dataset is a collection of recordings from various acoustic scenes all from distinct locations. All recordings consist of 10 seconds long wav files. All audio clips are recorded with 48 kHz sampling rate and 24 bit resolution. Training dataset includes 8640 audio files, evaluation (leaderboard) dataset includes 1200 audio files. All training files are labeled and every dataset has his own meta-file.

2.2. Audio Files Preprocessing

Downsampling wasn't used because of 10s files were used and it is needed to save a lot of information about audio data.

Amplitude scaling (from -1 to 1) was used. It is useful to make different spectrograms faster. Special class for data preprocessing was created with different method for every preprocessing and processing actions, such as load data, scaling it and calculate spectrograms.

3. DATA REPRESENATIONS

As mentioned, 2 DCNN architectures and 11 data representatons were used: MFCC (mono), Mel-spectrograms (Left, right, middle, side, harmonic and percusive) and CQT-spectrograms (left and right).

3.1. MFCC

In this work Mel-frequency cepstrum coefficients audio representation was used. It is a specail coefficients useful for speech recognition because of using human-friendly cepstral coefficients. MFCC uses the logarithmic perception of loudness and pitch of human level voice and tries to eliminate speaker dependent characteristics by excluding the fundamental frequency and their harmonics. Python librosa library was used to calculate this coefficients with 40 coefficients per audio files. 40 is a common value for speech recognition tasks. More and less values were tried during experiments, but there are no significant changes in the result. Also full audio clips (10 seconds) and chunks (1 second) were tried and it is shown that using chunks gives a big and fast overfitting, so full 10 seconds audio clips were used. DCNN with one single input for training on this data was used.

3.2. Mel-spectrograms

In this work Mel-spectrograms were used. It is a common approach to solve audio classification and tagging tasks. Librosa library was used to calculate spectrograms. Parameters of this feature are defined as follows: 128 bins, 2048 samples window size, 1024 samples hop size. These parameters were obtained by experiments. Thre resulting spectrogram was normalized by substracting the mean value and dividing by the standard deviation. As in [2] a lot of different representations were used. First, spectrograms of left (L) and right (R) channels were obtained. It gives a better space representations. Space informations also got by middle and side representations, where middle means $L + R$ and side means $L - R$.

Also harmonic and percussive separation was used by HPSS (harmonic-percussive sound separation) algorithm. It gives a better tones and pitch represenation. For example, it gives better difference understanding between tram and park, metro and street pedestrians and other.

Background separation using median filtering were used as a part of data representation. It was done using librosa library using nearest-neighbors filtering. Every sample was mean averaging with n nearest neighbours. This operatin removes noise from the spectrogram, making it more clearly.

As a result, seven different representation were obtained using mel-spectrograms: left, right, middle, side, harmonic, percussive and background separated.

3.3. CQT-spectrograms

The CQT-spectograms are calculated from the CQT features, computed using librosa. Default pararemeters, except of sampling rate, were used, such as 12 bins per octave and 512 as a hop length. Sampling rate was setup as 48000. CQT-spectrograms are good at low and mid-low frequencies [1], so it was useful for low-frequency and high-frequency audio data separation. CQT-spectrograms of two channels (left and right) were used. So, using this method, two additional representations of audio file were obtained.

4. SYSTEM ARCHITECTURE

As it was mentioned above, two types of DCNN were used: with one input and with two inputs. One-input network is used for background substructed audio data representation. Architecture of this networks is shown on Table 1.

The second type of DCNN is the same the first type except of having two inputs. This architecture is shown on Table 2.

As it can be seeing from the two-inputs DCNN architecures, two different representations of audio signal go in as an input. For example, left input is for the left channel, and right input is for the right channel. In this work we have used next pairs: mel-spectrogram from the left channel and from the right channel, mel-spectrograms of harmonic and percussiv and of middle and side channels.

As an ensembling goemetric mean was used. It was a lot of experiments with different types of means, such as simple average, harmonic mean, square mean and the result is that there are no significant differences between them.

Input (40, 938, 1)
ZeroPadding2D (1, 1) BatchNormalization Relu Convolution2D (3, 3) ZeroPadding2D (1, 1) BatchNormalization Relu Convolution2D (3, 3)
MaxPool2D (3)
ZeroPadding2D (1, 1) BatchNormalization Relu Convolution2D (3, 3) ZeroPadding2D (1, 1) BatchNormalization Relu Convolution2D (3, 3)
MaxPool2D (3)
ZeroPadding2D (1, 1) BatchNormalization Relu Convolution2D (3, 3) ZeroPadding2D (1, 1) BatchNormalization Relu Convolution2D (3, 3)
MaxPool2D (3)
GlobalAveragePooling2D Dense (1024) Dense (10) Softmax (10)

Table 1: One-input DCNN architecture

Input (40, 938, 1)
Input (40, 938, 1)
ZeroPadding2D (1, 1)
BatchNormalization
Relu
Convolution2D (3, 3)
ZeroPadding2D (1, 1)
BatchNormalization
Relu
Convolution2D (3, 3)
MaxPool2D (3)
ZeroPadding2D (1, 1)
BatchNormalization
Relu
Convolution2D (3, 3)
ZeroPadding2D (1, 1)
BatchNormalization
Relu
Convolution2D (3, 3)
MaxPool2D (3)
ZeroPadding2D (1, 1)
BatchNormalization
Relu
Convolution2D (3, 3)
ZeroPadding2D (1, 1)
BatchNormalization
Relu
Convolution2D (3, 3)
MaxPool2D (3)
GlobalAveragePooling2D
Concatenation
Dense (1024)
Dense (10)
Softmax (10)

Table 2: Two-inputs DCNN architecture

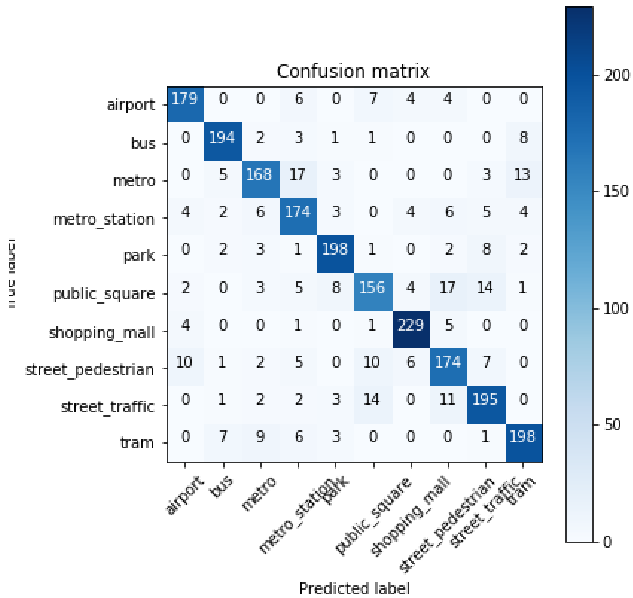

Figure 1: Confusion matrix

5. RESULTS

From feature extraction side such parameters as sampling rate, window length and hop length were used as experiment target. As it was mentioned above, downsampling wasn't used.

A lot of different CNN architectures were tried. As one of the results is that the deep level of CNN is not important. A very deep CNN with more than twenty convolutional layers were used, as well as ResNet, LeNet and VGG. There is no important increasing of the accuracy. Also RNN ans LSTM networks were tried, but there is a very fast overfitting and a lot of time for training, so it was not a good decision for this work.

It was a lot of different features tried to use with different CNNs, such as raw audio, mel-spectrograms, log-mels, MFCC, phase shifts, beam forming, IPD and PCA processing. Raw audio shows accuracy around 0.4, and it is as very bad result. Log-mels, mel-spectrograms and MFCC shows the same result around 0.65. IPD wasn't useful in this work as well as PCA processing. It is a question why it is true, and it is one of the point for the future experiments.

One of hardest classes were shopping mall and street pedestrian, because a lot of variants of futures and cnn gives maximum probabilities for these two classes. Also, one of the most popular confusions were between public square and park, public square and airport, metro and metro station and bus and train. The confusion matrix of the resulting model is shown in Figure 1.

6. REFERENCES

[1] Zheng Weiping, Yi Jiantao, Xing Xiaotao, Liu Xiangtao, Peng Shaohu Acoustic Scene Classification Using Deep Convolutional Neural Network and Multiple Spectrograms Fusion *DCASE 2017 Detection and Classification of Acoustic Scenes and Events*, 2017

[2] Yoonchang Han, Jeongsoo Park, Kyogu Lee Convolutional Neural Networks with binaural representations and background substraction for acoustic scene classification *DCASE 2017 Detection and Classification of Acoustic Scenes and Events*, 2017

[3] `http://dcase.community/challenge2018/task-acoustic-scene-classification#audio-dataset`.