ENVIRONMENTAL SOUND CLASSIFICATION USING MACHINE LEARNING

A Project Report submitted in partial fulfillment of the

M.Sc. Fourth Semester Examination

Session (2020-2021)

To

DEPARTMENT OF COMPUTER SCIENCE

INSTITUTE OF SCIENCE

BANARAS HINDU UNIVERSITY

VARANASI

Under the Supervision of:
Dr. Suresh Selvam

Submitted by:
SAYAN ROY
ROLL-NO.19419CMP019

CANDIDATE'S DECLARATION

I hereby certify that the work, which is being presented in the report/thesis, entitled **Environment Sound Classification using Machine Learning,** in partial fulfillment of **M.Sc. fourth semester Examination Session (2020-2021)** and submitted to the institution is an authentic record of my/our own work carried out during the period Month-Year to Month-Year under the supervision of supervisor(s) name. I also cited the reference about the text(s) /figure(s) /table(s) /equation(s) from where they have been taken.

The matter presented in this thesis as not been submitted elsewhere for the award of any other degree or diploma from any Institutions.

Date: Signature of the Candidate

This is to certify that the above statement made by the candidate is correct to the best of my/our knowledge.

Date: Signature of the Research Supervisor

Annexure 2

ABSTRACT

The main goal of this work is the implementation of Environmental Sound Classification using Machine Learning. Sound Classification is the most widely used application in Deep Learning. It involves learning different categories of sound and as a result it can almost correctly predict the category of the sound. This approach can be used to solve different types of problems, classifying music clips to identify the genre of the music, or classifying short utterances by a set of speakers to identify the speaker based on the voice, Piano skill assessments like this. CNNs or convolutional neural networks are a type of deep learning algorithm that does really well at learning audios. That's because they can learn patterns that are translation invariant and have spatial hierarchies. Here we have enhanced the model and extracted the features and then depending upon the features, implemented the model and using those features model tries to predict the category of the sound. In the domain of audio classification filed there is four types of sound classification, like Acoustic Data Classification, Environmental Sound Classification, Music Classification, and Natural Language Utterance Classification. Here we have implemented Environmental Sound Classification and will explain about that in this paper too. For this purpose we are using UrbanSound8K dataset. This dataset contains above eight thousand different types of sound. The result of my project will depends on the quality of the sounds of the UrbanSound8K dataset. Therefore, to ensure an accurate prediction of audio we need a good volume of high-quality, accurately-annotated data.

TABLE OF CONTENTS

LIST OF FIGURES

Figure No. **Title** **Page No.**

LIST OF ABBREVIATIONS

Hz	Hertz for measuring sound frequency
CV	Computer Vision
NLP	Natural Language Processing
CNN	Convolutional neural network

Annexure 8

Environmental Sound Classification using Machine Learning

By Sayan Roy
Department of Computer science
Institute of Science
Banaras Hindu University, Varanasi

Abstract

The main goal of this work is the implementation of Environmental Sound Classification using Machine Learning. Sound Classification is the most widely used application in Deep Learning. It involves learning different categories of sound and as a result it can almost correctly predict the category of the sound. This approach can be used to solve different types of problems, classifying music clips to identify the genre of the music, or classifying short utterances by a set of speakers to identify the speaker based on the voice, Piano skill assessments like this. CNNs or convolutional neural networks are a type of deep learning algorithm that does really well at learning audios. That's because they can learn patterns that are translation invariant and have spatial hierarchies. Here we have enhanced the model and extracted the features and then depending upon the features, implemented the model and using those features model tries to predict the category of the sound. In the domain of audio classification filed there is four types of sound classification, like Acoustic Data Classification, Environmental Sound Classification, Music Classification, and Natural Language Utterance Classification. Here we have implemented Environmental Sound Classification and will explain about that in this paper too. For this purpose we are using UrbanSound8K dataset. This dataset contains above eight thousand different types of sound. The result of my project will depends on the quality of the sounds of the UrbanSound8K dataset. Therefore, to ensure an accurate prediction of audio we need a good volume of high-quality, accurately-annotated data.

CHAPTER 1

INTRODUCTION

1.1 GENERAL

In our environment we hear a lot of sounds in our daily life. This model will predict an appropriate categories of environment sound. Now a days it is one of the most widely used application in Audio Deep Learning. It involves learning to classify sounds and to predict the category of that sound. This types of problem can be applied to many practical scenarios, e.g. Music Genre Prediction, Classify voice data, Classify short utterances by a speaker to identify the speaker, Piano Skill checking etc.

1.2 OBJECTIVES

The main goal of this work is the implementation of Environmental Sound Classification using Machine Learning. We will understand the approaches used to solve such audio classification problems. We will implement a model also. Mel-frequency will be used to build this model. Our model will try to predict the correct category of sound.

CHAPTER 2
LITERATURE REVIEW

2.1 INTRODUCTION

Although Computer Vision and Natural Language Processing applications gets more attractions but there are many groundbreaking use cases for deep learning with audio data that are transforming our daily lives. We know that sound is a form of signal and signal is produced by variations in air pressure. We can measure the intensity of the pressure variations and plot those measurements over the time. Sound signal often repeat itself in a regular time interval so that each wave has the exact same shape. The height of the signal shows the intensity of the sound and it is also known as the amplitude of the signal.

2.2 Sound

Sound is a form of signal and signal is produced by variations in air pressure. We can measure the intensity of the pressure variations and plot those measurements over the time. Sound signal often repeat itself in a regular time interval so that each wave has the exact same shape. The height of the signal shows the intensity of the sound and it is also known as the amplitude of the signal.

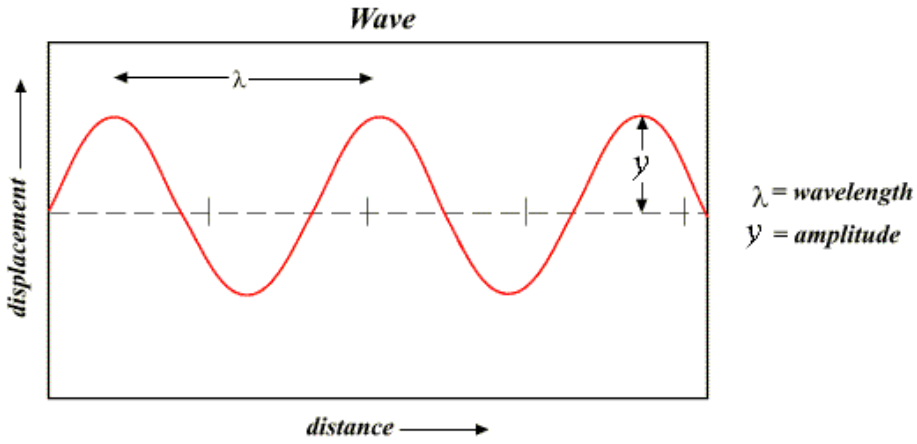

Fig-1 Source: www3.cs.stonybrook.edu

Time taken for the signal to complete one full wave is called the period of the sound. The number of waves made by the sound signal in one second is called the frequency. The unit of the frequency is Hz.

The majority of sounds we encounter may not follow such simple and regular periodic patterns. But signals of different frequencies can be added together to create composite signals with more complex repeating patterns. All sounds that we hear, including our own human voice, consist of waveforms like these. For instance, this could be the sound wave of a music.

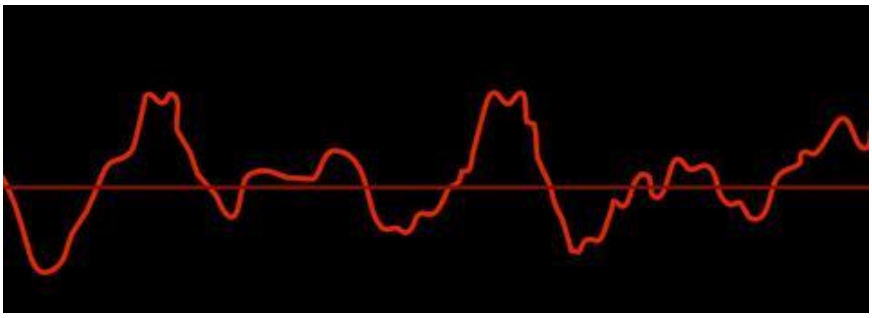

Fig-2 Source: Phiaton

Human ear is able to differentiate between a different sounds based on the quality of the sound which is also known as timbre.

2.3 Digital Sound Representation

To represent a sound digitally we must have to turn the signal into a series of numbers so that we can input it into our models. This is done by measuring the amplitude of the sound at fixed intervals of time.

2.4 Mel Scale

The Mel Scale was developed to take this into account by conducting experiments with a large number of listeners. It is a scale of pitches, such that each unit is judged by listeners to be equal in pitch distance from the next.

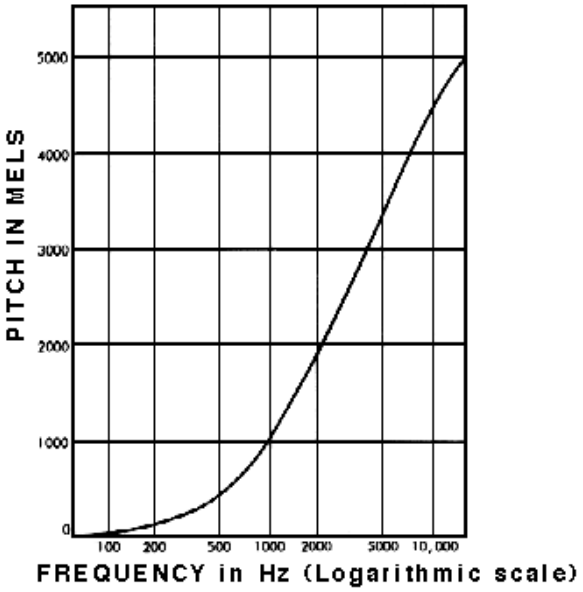

Fig-3 Source: www.sfu.ca

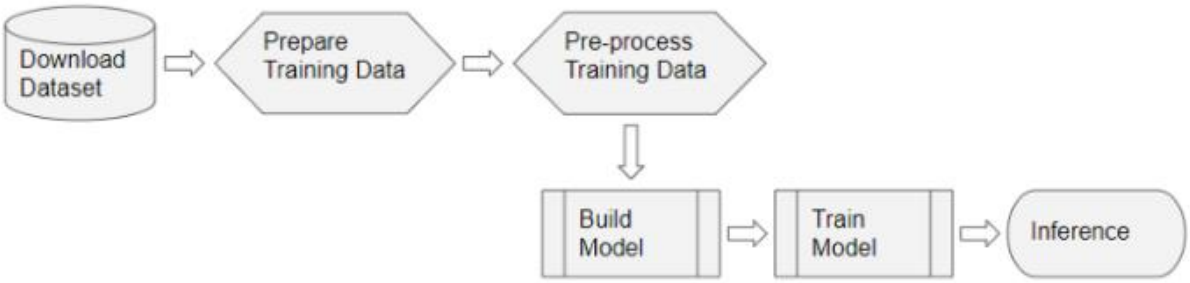

Fig-4 procedure

CHAPTER-3
PROPOSED APPROACH

3.1 Introduction

To implement an environment sound classifier model has four stages to follow: Spectrogram implementation, Sequential Architecture, Feature Maps and lastly linear classifier. Model establishment and training and evaluation of the experimental results as shown.

3.2 Data Source

We have selected UrbanSound8k as experimental data. UrbanSound8K contains approx. 8,200 and more than that different types of sounds.

	slice_file_name	fsID	start	end	salience	fold	classID	class
0	100032-3-0-0.wav	100032	0.0	0.317551	1	5	3	dog_bark
1	100263-2-0-117.wav	100263	58.5	62.500000	1	5	2	children_playing
2	100263-2-0-121.wav	100263	60.5	64.500000	1	5	2	children_playing
3	100263-2-0-126.wav	100263	63.0	67.000000	1	5	2	children_playing
4	100263-2-0-137.wav	100263	68.5	72.500000	1	5	2	children_playing

Fig-5 dataset

Audio files in the ‘audio’ folder: It has 10 sub-folders named ‘fold1’ through ‘fold10’. Each sub-folder contains a number of ‘.wav’ audio samples eg. ‘fold1/103074–7–1–0.wav’
Metadata in the ‘metadata’ folder: It has a file ‘UrbanSound8K.csv’ that contains information about each audio sample in the dataset such as its filename, its class label, the ‘fold’ sub-folder location, and so on. The class label is a numeric Class ID from 0–9 for each of the 10 classes. eg. the number 0 means air conditioner, 1 is a car horn, and so on.

3.3 CNN

Sequential model is the easiest way to build a model in keras. It allows us to build a model layer by layer. We use the 'add ()' function to add layers to our model.

Our first 2 layers are Conv2D layers. These are convolution layers that will deal with our input images, which are seen as 2-dimensional matrices.

64 in the first layer and 32 in the second layer are the number of nodes in each layer. This number can be adjusted to be higher or lower, depending on the size of the dataset. In our case, 64 and 32 work well, so we will stick with this for now.

Kernel size is the size of the filter matrix for our convolution. So a kernel size of 3 means we will have a 3x3 filter matrix. Refer back to the introduction and the first image for a refresher on this.

Activation is the activation function for the layer. The activation function we will be using for our first 2 layers is the ReLU, or Rectified Linear Activation. This activation function has been proven to work well in neural networks.

Our first layer also takes in an input shape. This is the shape of each input image, 28,28,1 as seen earlier on, with the 1 signifying that the images are greyscale.

In between the Conv2D layers and the dense layer, there is a 'Flatten' layer. Flatten serves as a connection between the convolution and dense layers.

'Dense' is the layer type we will use in for our output layer. Dense is a standard layer type that is used in many cases for neural networks.

We will have 10 nodes in our output layer, one for each possible outcome (0–9).

The activation is 'softmax'. Softmax makes the output sum up to 1 so the output can be interpreted as probabilities. The model will then make its prediction based on which option has the highest probability.

Chapter-4
IMPLEMENTATION

4.1 Data preprocessing

We have implemented the proposed environmental sound classification method usPython tensorflow library. We have used the data and divided it into training and test datasets. For the UrbanSound8K dataset. Which is used to training the model and also we are using that for testing the model. Audio data is obtained by sampling the sound wave at regular intervals and measuring the intensity or amplitude of the wave at each sample. The metadata for that audio tells us the sampling rate which is the number of samples per second. When audio is saved in a file it is in a compressed format. When the file is loaded, it is decompressed and converted into a numpy array. This array looks the same no matter which file format started with.

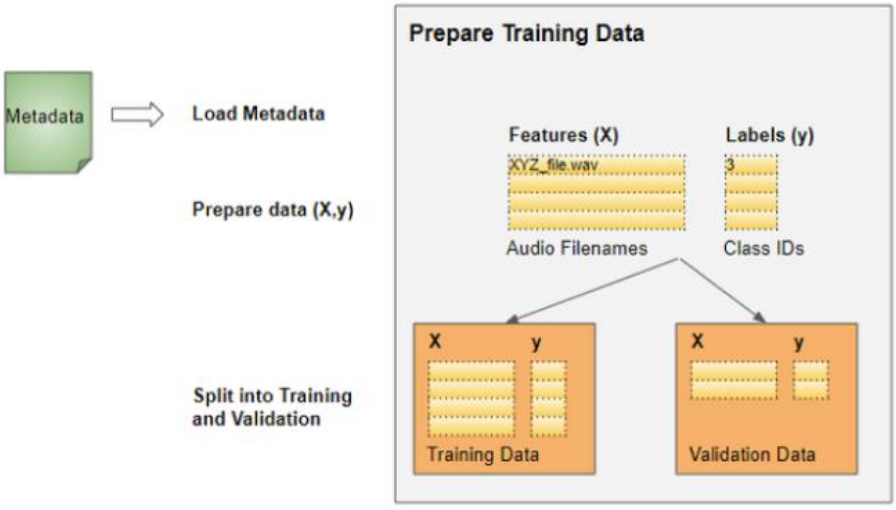

Fig-6 Test data and Validation Data

4.2 Spectrogram

Since a signal produces different sounds as it varies over time, its constituent frequencies also vary with time. In other words, its Spectrum varies with time.

A Spectrogram of a signal plots its Spectrum over time and is like a ‘photograph’ of the signal. It plots Time on the x-axis and Frequency on the y-axis. It is as though we took the Spectrum again and again at different instances in time, and then joined them all together into a single plot.

It uses different colors to indicate the Amplitude or strength of each frequency. The brighter the color the higher the energy of the signal. Each vertical ‘slice’ of the Spectrogram is essentially the Spectrum of the signal at that instant in time and shows how the signal strength is distributed in every frequency found in the signal at that instant.

In the example below, the first picture displays the signal in the Time domain ie. Amplitude vs Time. It gives us a sense of how loud or quiet a clip is at any point in time, but it gives us very little information about which frequencies are present.

4.3 Mel Spectrogram

The Mel Scale was developed to take this into account by conducting experiments with a large number of listeners. It is a scale of pitches, such that each unit is judged by listeners to be equal in pitch distance from the next.

A Mel Spectrogram makes two important changes relative to a regular Spectrogram that plots Frequency vs Time.

It uses the Mel Scale instead of Frequency on the y-axis.

It uses the Decibel Scale instead of Amplitude to indicate colours.

For deep learning models, we usually use this rather than a simple Spectrogram.

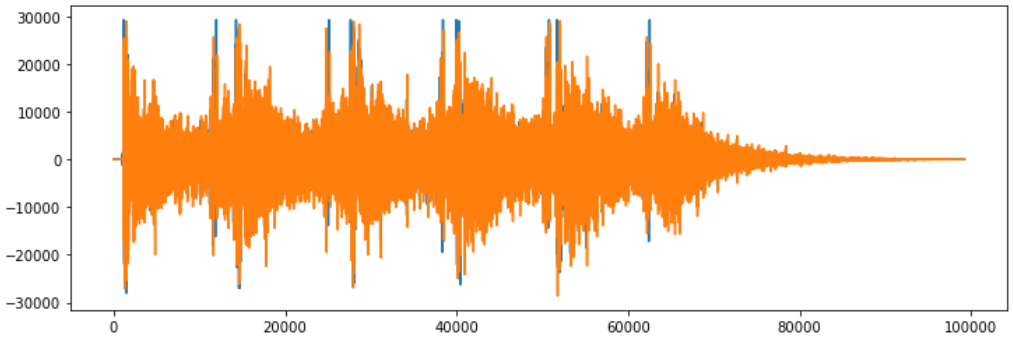

Fig-7 Spectrogram

Chapter-5

Result and Discussion

5.1 Result & Discussion

This is one of the most common use cases and involves taking a sound and assigning it to one of several classes. For instance, the task could be to identify the type or source of the sound. eg. is this a car starting, is this a hammer, a whistle, or a dog barking.
Obviously, the possible applications are vast. This could be applied to detect the failure of machinery or equipment based on the sound that it produces, or in a surveillance system, to detect security break-ins. We have now seen an end-to-end example of sound classification which is one of the most foundational problems in audio deep learning. Accuracy of the model is 76.3%.

```
In [44]: filename="C:/Users/User/Downloads/Compressed/UrbanSound8K.tar/UrbanSound8K/audio/test.wav"
audio, sample_rate = librosa.load(filename, res_type='kaiser_fast')
mfccs_features = librosa.feature.mfcc(y=audio, sr=sample_rate, n_mfcc=40)
mfccs_scaled_features = np.mean(mfccs_features.T,axis=0)

print(mfccs_scaled_features)
mfccs_scaled_features = mfccs_scaled_features.reshape(1,-1)
print(mfccs_scaled_features)
print(mfccs_scaled_features.shape)
predicted_label=model.predict_classes(mfccs_scaled_features)
print(predicted_label)
from sklearn.preprocessing import LabelEncoder
prediction_class = labelencoder.inverse_transform(predicted_label)
prediction_class

[-165.4269       43.89598     -44.51454      13.66336     -31.51437
   16.695425     -8.83037       1.0299591   -17.401968      8.428135
   -5.697943     12.292311     -1.1667279     4.646665    -13.346708
    3.7642868    -6.512463      5.4654503    -7.672402      7.259151
   -1.8075553     5.2677484    -1.801911      1.2806062    -1.5214287
    1.3443055    -9.449034      1.4809908    -4.481703     -4.2806783
   -2.6433184     0.9573297    -2.2087617     1.5173422    -5.6189384
    0.51588243   -3.3322306     0.2297842     3.7476482    -0.96686554]
[[-165.4269       43.89598     -44.51454      13.66336     -31.51437
    16.695425     -8.83037       1.0299591   -17.401968      8.428135
    -5.697943     12.292311     -1.1667279     4.646665    -13.346708
     3.7642868    -6.512463      5.4654503    -7.672402      7.259151
    -1.8075553     5.2677484    -1.801911      1.2806062    -1.5214287
     1.3443055    -9.449034      1.4809908    -4.481703     -4.2806783

    -2.6433184     0.9573297    -2.2087617     1.5173422    -5.6189384
     0.51588243   -3.3322306     0.2297842     3.7476482    -0.96686554]]
(1, 40)
[4]

Out[44]: array(['drilling'], dtype='<U16')
```

Fig-8 Result

Chapter-6
Future Work

6.1 Future Work

This model is build using CNN (Convolutional neural network) and it gives us accuracy up to 76.38 point after this point it’s not accurate. Besides this model can be improved to build music genre prediction, speech recognition this type of projects.

REFERENCES

[1] https://towardsdatascience.com/audio-deep-learning-made-simple-sound-classification-step-by-step-cebc936bbe5

[2] G.Tzanetakis and P. Cook. “Musical genre classification of audio signals, Speech and Audio Processing”, IEEE Transactions, July 2002.

[3] Chandsheng Xu, Mc Maddage, Xi Shao, Fang Cao, and Qi Tan, ”Musical genre classification using support vector machines”, IEEE Proceedings of International Conference of Acoustics, Speech, and Signal Processing, Vol. 5, pp. V-429-32, 2003.

[4] Matthew Creme, Charles Burlin, Raphael Lenain, “Music Genre Classification ”, Stanford University, December 15, 2016.

[5] T. Feng, “Deep learning for music genre classification”, 2014.

[6] Daniel Grzywczak, Grzegorz Gwardys, “Deep image features in music information retrieval”, 10th international conference, AMT 2014, Warsaw, Poland, August 11-14, 2014 proceedings, pp 187-199.

[7] Muhammad Asim Ali, Zain Ahmed Siddqui, “Automatic Music Genres Classification using Machine Learning”, International Journal of Advanced Computer Science and Applications, Vol 8, No 8, 2017.

[8] Hareesh Bahuleyan, “Music Genre Classification using Machine Learning Techniques”, University of Waterloo, ON, Canada, 2018

[9] Sam Clark, Danny Park, Adrien Guerard, “Music Genre Classification using Machine Learning Techniques”, 2012.

[10] Eve Zheng, Melody Moh, Teng-Sheng Moh, “Music Genre Classification: A N-gram based Musicological Approach”, 7th International Advance Computing Conference, 672-677, 2017.

[11] https://urbansounddataset.weebly.com/urbansound8k.html