Outline

1. Abstract
2. Introduction
3. Objective of the research study
4. Literature Review
5. Preprocessing the data
6. Exploratory Data Analysis
    (a) Urbansound8k Dataset
    (b) Audio Sampling
    (c) Digitization of sound signal
    (d) Visual representation and exploration of the dataset
        i. Time - Amplitude Domain
        ii. Spectrograms
        iii. Converting the properties of dataset into dataframe
        iv. Converting the Sampling Rate
7. Feature Extraction

1 Abstract

Automatic environmental sound classification is a growing area of research with numerous real world applications. Whilst there is a large body of research in related audio fields such as speech and music, work on the classification of environmental sounds is comparatively scarce. Likewise, observing the recent advancements in the field of image classification where convolutional neural networks are used to to classify images with high accuracy and at scale, it begs the question of the applicability of these techniques in other domains, such as sound classification.In this study, we will walk through a simple demo application so as to understand the approach used to solve such audio classification problems. We will be using Mel Spectrograms for deep learning models and how they are generated and optimized.

2 Introduction

With the increasing technological aspects, the detection of the events have also become one of the major attributes in the development of technology. The events can be related to sounds, images, videos etc. The introduction of machine learning has now made it easier for image detection using specific algorithms and it is being used in the market for various applications. But the sound event detection is still struggling. The sound detection can provide more useful information about the incidents that fail with only graphical information. In environments like when it is dark or cloudy the graphical data provided may not be informative or full of noises. But in these conditions the sound detection can be more helpful and informative than graphical ones. The sound detection system is cheaper than the image recognition system and may be more helpful along with graphical information. There are various kinds of sounds in the environment. The detection of every sound may not be possible or useful. But the sound that is related to human interests can be detected to retrieve various useful information. The total number of sounds present in the environment and the diversity among them creates many applications for the sound event detection system. Now, it is very important to know that sound events can be detected by classifying the audio. The audio classification has various applications such as it can be used in classifying the genre

or mood of the musical audio. The classification of the audio is also applicable in industries for monitoring the industrial equipment. In industries the classification of audio can be used to detect the sound that can be faulty and may help in diagnosing the fault. [4] One of the most important uses of audio classification is for hearing impaired people. People exposed to noisy environments are prone to hearing loss. [1] The traditional HPDs(Hearing protection devices) used in these environments suppress all the sounds. [6] So, the people are unaware of the warnings or other important sounds. The noise cancellation system can be used for clearing the background noise and hence enhancing the quality of speech. The classification of the sound system can also be useful in detecting the various heartbeats to measure the cardiac condition of the patient. It is among the very useful applications of the sound system classification in the field of biotechnology. Furthermore, the audio classification can also be used for the detection of sirens. In this study, we would be classifying the audio for the detection of the siren. It has various applications such as it can be used in the traffic management system, in industries for the hearing impaired people working in the industries, emergency services etc. In this study, the detection of some of the environmental sounds has been done. These environmental sounds include the sound of the air conditioner to predict its maintenance period. The sound of the drilling machine, jackhammer, barking of dogs, emergency vehicle sounds, siren, gunshots, etc. has also been detected using audio classification.

3 Objective of the research study

There are many objectives of the sound classification such as the detection of the sirens or emergency vehicle sounds could be the reason of traffic management in high traffic areas. If at the intersection of the road the approaching non-emergency vehicle comes to know about the crossing emergency vehicle before it reaches to the intersection then it can give a safe passage to that vehicle. [3] It is also helpful in assisting the hearing impairment people walking on the road. The detection of the sound of air conditioner could be helpful in determining the maintenance time. The sound classification can be very useful for security purpose as if the sound of gunshot can be detected then it can be very useful for the cops in collecting the

proofs on the crime scene. The sound of the drilling machines and jackhammers detection could be useful for the construction purpose.

4 Literature Review

Various methods have been used for the classification of sound and extracting the features to detect a particular sound.

[7] has proposed a method in which the real time sound is recorded and then converted to .wav format. After this the file is transferred to the cloud to check if the frequency matches the threshold frequency. The proposed method has been implemented in the scenarios where sound is above 13 dB (threshold frequency). The audio file that crosses the threshold is sent to the machine for the classification process. This method uses the IOT technology so the user gets notified about the type of sound. This method is easy to operate and it can be very useful to the hearing impaired people. It provides real time experience as the user gets notified about the type of sound.

[6] has proposed a method in which an experimental setup includes the deployment of two mics in two different reverberant rooms. The experiment is performed in two different scenarios i.e. in one room the SNR is high and in another room SNR is very low. The proposed method used the MFCCs feature extraction through MLP(Multi-layer perceptron) method of VAD for the real world scenario consisting of high SNR ratio environments. The demixing of the mixture of signals is done by the CBSS(Convolutive Blind Source Separation) algorithm. Adaptive filter techniques have been used to enhance the quality of the required signal.

[1] uses a method in which the dataset is band pass filtered to remove the noises from the dataset. It uses the supervised machine learning approach for the classification purpose. The SVM classifier has been used in this literature that is trained using the laballed signals. The statistical analysis has also been used for the feature extraction. The selection of the required features is done by the ReliefF algorithm. The authors have used the k-fold cross validation for the evaluation. It has achieved satisfactory results for the real time problems such as doppler effect, low amplitude noise etc. But the dataset used in this literature is

very limited.

[8] employs a method in which it removes the noisy signals through peak detection in which the peaks that are deviated from the maximum height or the threshold gets eliminated. Mainly it focuses on the cut off of 25% elimination. The average distance between the peaks is calculated to distinguish between the noisy and not noisy signals. The average distance between the noisy signals is less in comparison to the not noisy signals. The ReLu function is used as the activation function to solve the gradient descent problem. The Aadam optimization algorithm has been proposed for the optimization of the result obtained from the noise detection algorithms.

[2] has proposed a novel approach for the detection of the siren. This literature has converted the time series signal into cloud points of n-dimensional space using sliding window transformation. After then to lower the dimensional space the dimension reduction algorithm has been used. The simplicial filtration technique has been used for the filtration process. The persistent homological technique is used to obtain the persistent diagram and the numerical marginals are computed from the persistent diagrams.

5 Preprocessing the audio data

The dataset cannot be given to the model as input directly. The data should be loaded and processed so that it could be converted into the format that the model can accept. The processing of the audio dataset will be done during the runtime during the reading and kloading of the audio files. It is similar to that of what we do with the image dataset. Instead of reading the memory at once we will be loading only the name of the files that we require to train the model.

6 Exploratory Data Analysis

6.1 Urbansound8k Dataset

The datset used in this study has been taken fron the internet named urbansound8k. The sound frames are of less than four seconds. All the sounds have been digitized to be used by

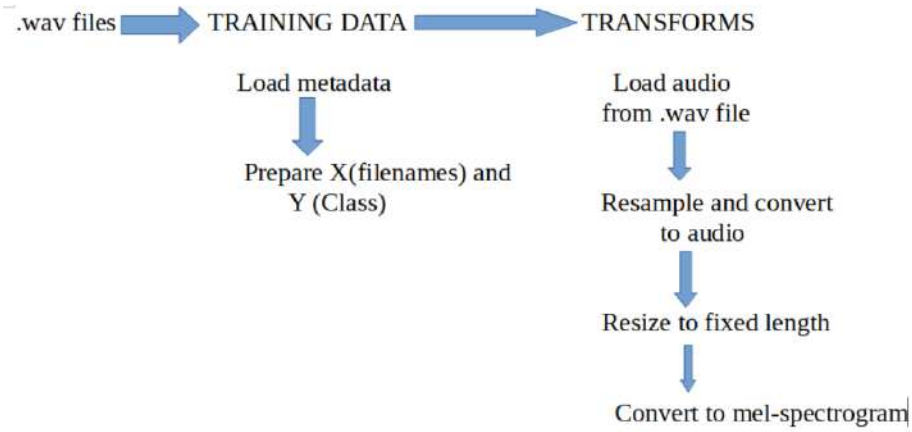

Figure 1: Audio Preprocessing

the machine. The sound files are in .wav file format. The different categories of the sounds that are included in this dataset are following

- Air Conditioner Sound
- Barking Dog
- Gunshot
- Vehicle horns
- Drilling Machines
- Jackhammer
- Street Music
- Emergency Sounds
- Children Playing
- Engine Idling

6.2 Audio Sampling

In simple words, the sound can be defined as the wave due the disturbance in the air pressure. In other words, the sound can be defined as the longitudinal wave that is caused due the

variation in the pressure of the medium in which it is transmitting.It has been shown in fig 2.

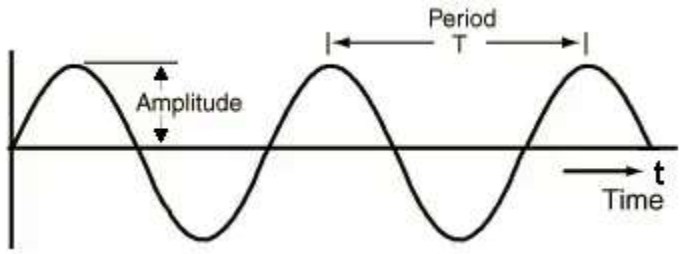

Figure 2: Amplitude vs Time graph

In fig 2 the representation of a signal with its amplitude varying with time is the simplest signal that can be represented. But in the real world the sound signals that could be heard daily can't be represented simply. The environmental sounds that we hear daily is more complex than the above shown fig 3. For example, this can be the sound of the street music or the sound consisting of various signals composed in it.

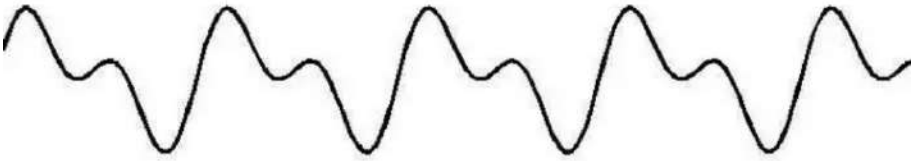

Figure 3: Representation of environmental sound signal

6.3 Digitization of sound signal

The sound waves can be digitized by the process called sampling. The sampling of sound can be defined as obtaining the sound waves in the fixed interval of time. The sampling rate is obtained for the digitization of the sound wave. The average sampling rate that has been computed is 44,100 samples per second. The fig 4 and fig 5 represents the sampling rate. Each sample is the amplitude of the sound wave computed at a particular interval of time.

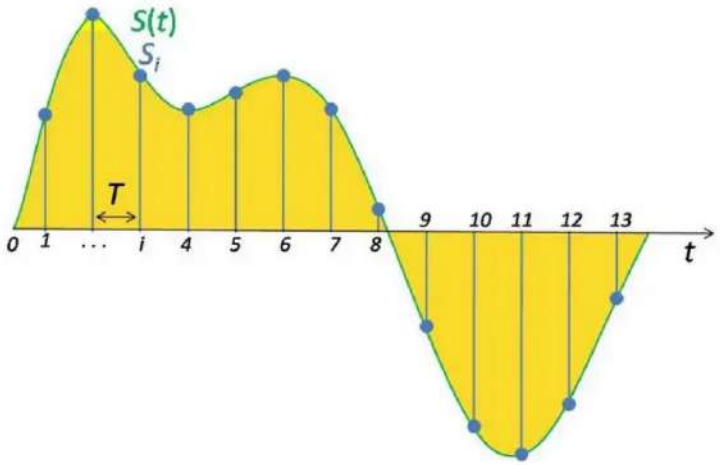

Figure 4: Representation of sample rating

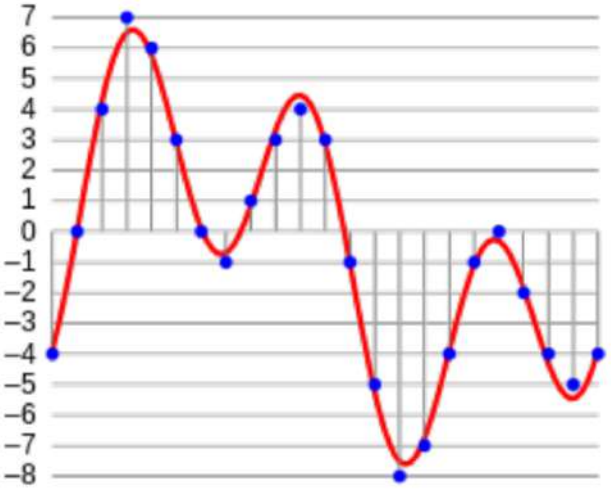

Figure 5: Representation of sample rating

6.4 Visual representation and exploration of the dataset

The dataset can be visually represented in the machines by generating the images of the signal

6.4.1 Time - Amplitude Domain

In the following figures the signals are represented in the time and frequency domain. The x-axis represents the frequency whereas the y-axis represents the amplitude of the signal.

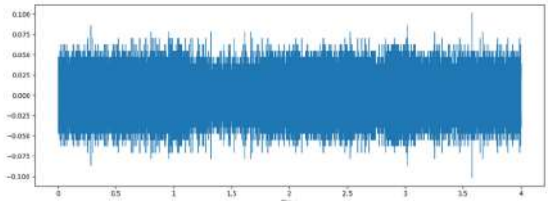

Figure 6: Air conditioner

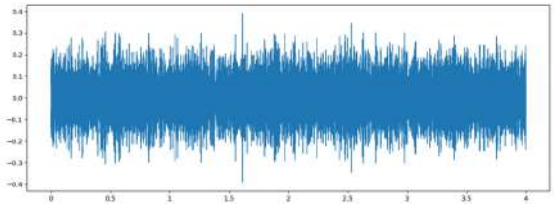

Figure 7: Engine Idling

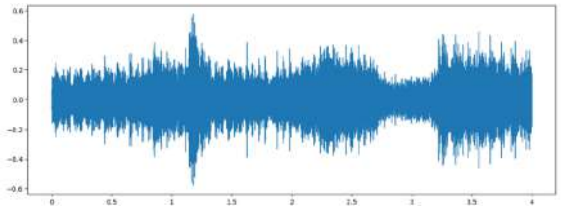

Figure 8: Jackhammer

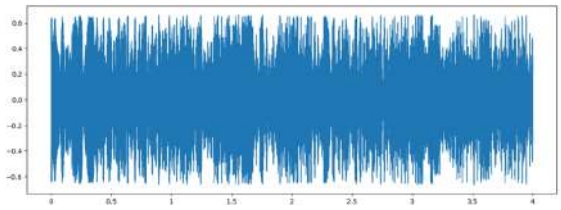

Figure 9: Emergency Sound

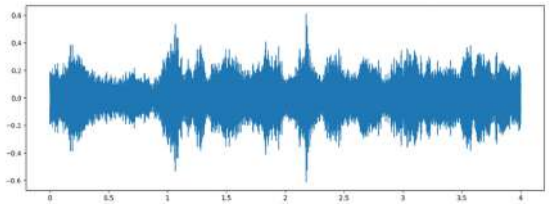

Figure 10: Children Playing

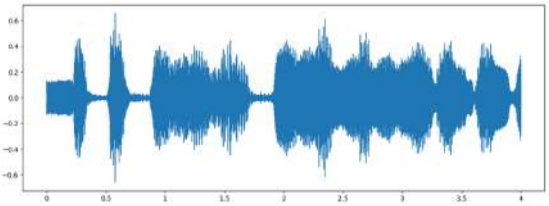

Figure 11: Street Music

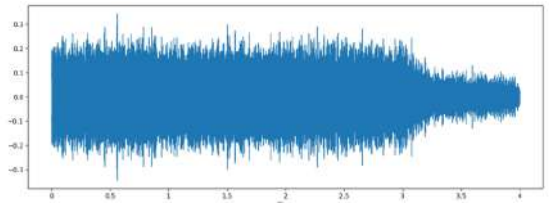

Figure 12: Drilling

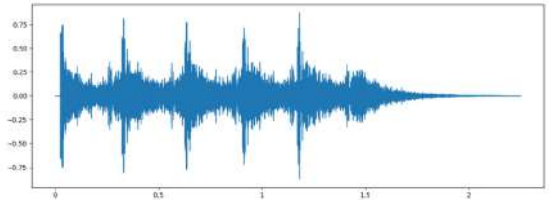

Figure 13: Gunshot

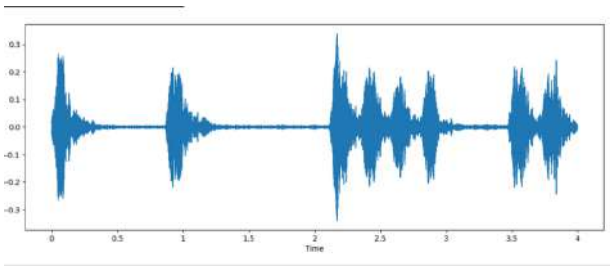

Figure 14: Barking Dog

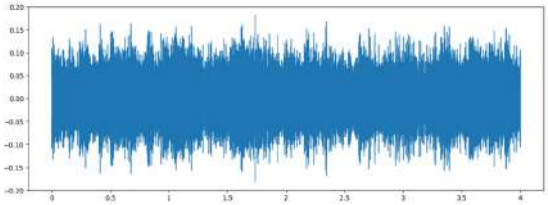

Figure 15: Vehicle Horn

6.4.2 Spectrograms

The time-domain frequency only represents a single signal so it may be difficult to extract more details from this feature. So the new concept of spectrogram has been introduced to solve this problem. The spectrum can be defined as the set of more than one signals or we can say that composite signals. The real world sound that we hear around us can be best explained in the context of the spectrum. The spectrogram is the feature of the signal. Here signal refers to the set of more than one signals. So this data can be more useful in classifying the audio signals. The spectrograms are used to represent the frequencies of all the signals that are superimposed. it uses different types of colors to represent the frequency of the superimposed signals. It also represents the energy of the signals. If the energy of the signal is higher then that frequency will appear brighter on the spectrogram whereas on the other side if the energy of the signal is lower then its frequency can be represented by the dull color on the spectrogram. Some of the following spectrogram features that has been extracted have been shown by the following figures.

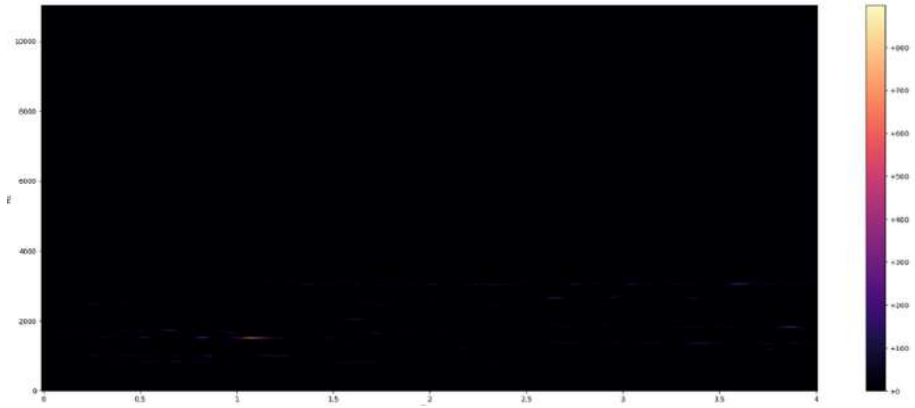

Figure 16: Spectrogram

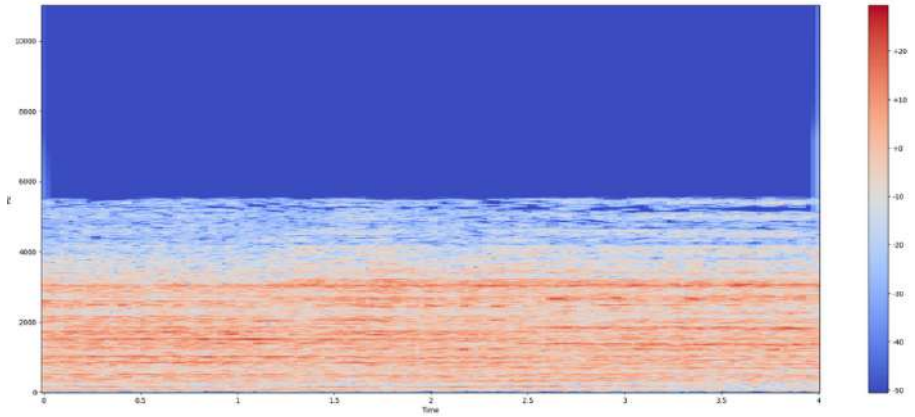

Figure 17: Log-Amplitude Spectrogram

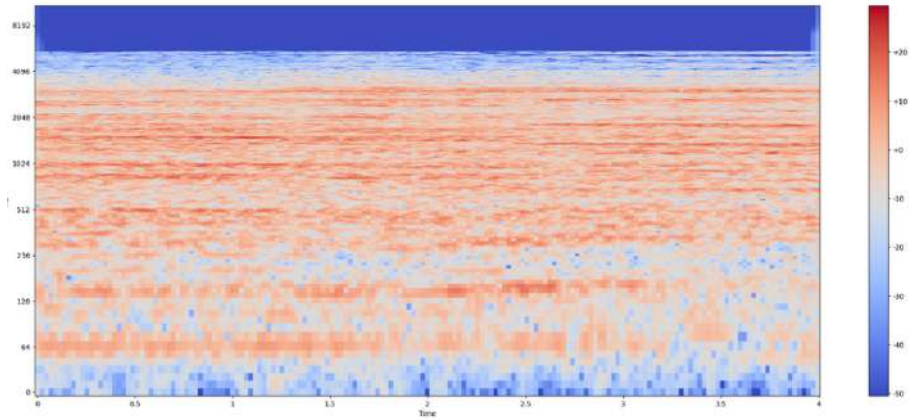

Figure 18: Log-Frequency Spectrogram

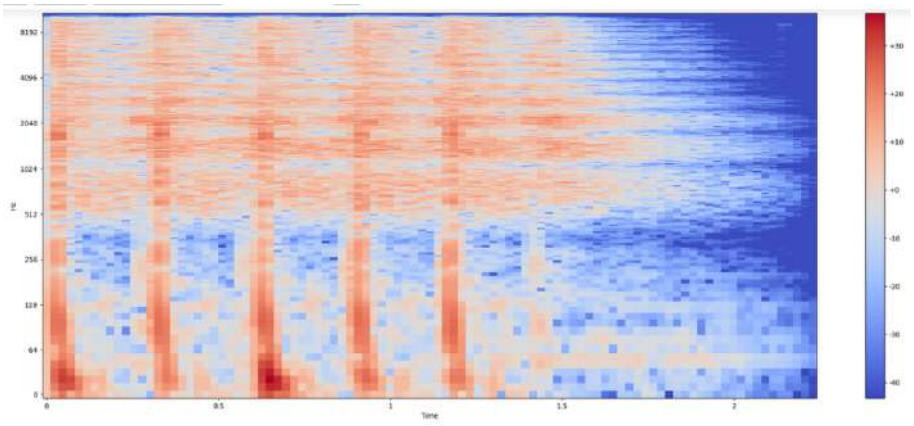

Figure 19:

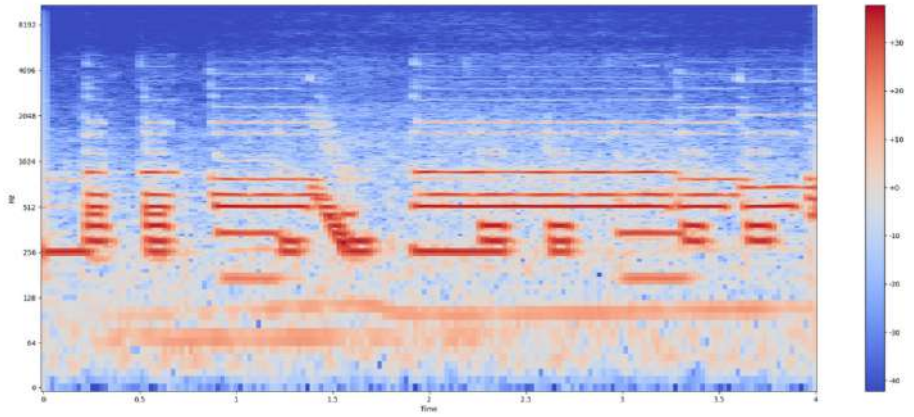

Figure 20:

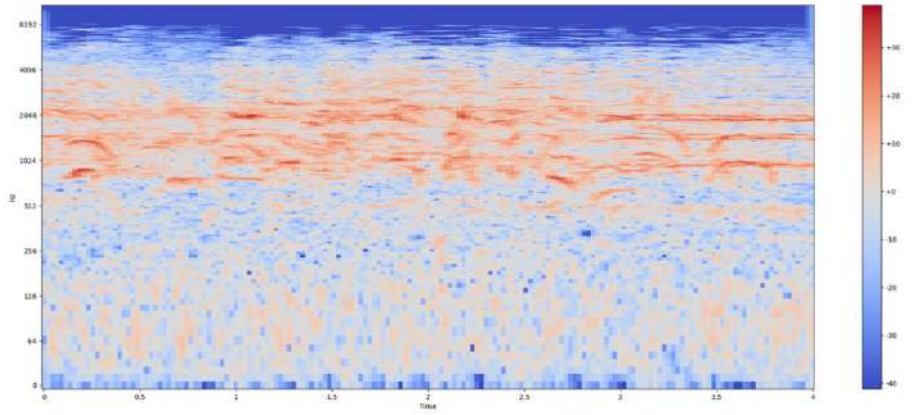

Figure 21:

6.4.3 Converting the properties of dataset into dataframe

It may be very difficult in distinguishing between the type of sound by just using the above generated images of the signal. The signal may consist of more than on type of sounds. Also the images obtained appear similar visually. So for the classification of the sound type, we need to convert the given data into the numerical dataset. So we take the dataset into the table frame along with the classes. But in fig22, it can be seen that the data set seems to be unbalanced.

	slice_file_name	fsID	start	end	salience	fold	classID	class
0	100032-3-0-0.wav	100032	0.000000	0.317551	1	5	3	dog_bark
1	100263-2-0-117.wav	100263	58.500000	62.500000	1	5	2	children_playing
2	100263-2-0-121.wav	100263	60.500000	64.500000	1	5	2	children_playing
3	100263-2-0-126.wav	100263	63.000000	67.000000	1	5	2	children_playing
4	100263-2-0-137.wav	100263	68.500000	72.500000	1	5	2	children_playing
5	100263-2-0-143.wav	100263	71.500000	75.500000	1	5	2	children_playing
6	100263-2-0-161.wav	100263	80.500000	84.500000	1	5	2	children_playing
7	100263-2-0-3.wav	100263	1.500000	5.500000	1	5	2	children_playing
8	100263-2-0-36.wav	100263	18.000000	22.000000	1	5	2	children_playing
9	100648-1-0-0.wav	100648	4.823402	5.471927	2	10	1	car_horn

Figure 22:

```
dog_bark           1000
children_playing   1000
air_conditioner    1000
street_music       1000
engine_idling      1000
jackhammer         1000
drilling           1000
siren               929
car_horn            429
gun_shot            374
Name: class, dtype: int64
```

Figure 23: Class Distributions

From the fig 23 it can be seen that most of the classes(dog_bark, children_playing, air_conditioner, street_music, engine_idling, jackhammer, drilling) have the same counts while in the case of other cases there are only few data available so it may create confusion between the classes. If the data remains unbalanced then there is more probability of the errors in the model.

6.4.4 Converting the Sampling Rate

We have seen that the average sampling rate is 44,100 and it gives better quality sound. But we need to transform the sample rate to its half so that it can be used for the sound classification purpose. Now, the resampling of the audio file will be done to transform the sampling rate of 44.1 Khz to 22.05 Khz. By doing this transformation every other frequencies will be removed from the sample and only the required frequency will be shown. In order to explain this, Nyquist theorem can be used which states that if the sampling rate is halved then the frequency with the highest can be represented more accurately in comparison to all the other frequencies. The signals with more than one channel or we can say that stereotype signals will be transformed into mono type signal. This will be more helpful in determining the particular frequency.

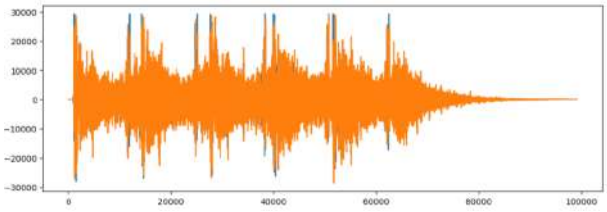

Figure 24: Original Audio with more than one channels

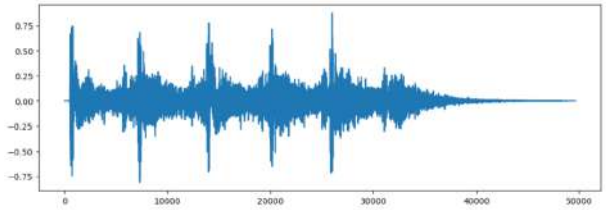

Figure 25: Audio converted to mono

From the above figures 24 and 25 it can be seen that the image that has been converted to mono more clear. The image obtained from the two channel audio is showing the superimposed signals so it is difficult to classify the sound with the help of the signals of stereotype audios.

7 Feature Extraction

7.1 MFCCs

MFCCs(Mel-Frequency Cepstral Coefficients) are the visual representation of the audio signals. Now, these visual representation will be used for the extraction of the required features for the classification sound. The spectrograms obtaind from the datset can also be used for extracting the features of the audio signal. But in this study, MFCCs will be used for the feature extraction. [7] The MFCCs can also be used for the speech recognition. Now, the question arises that how can be the MFCCs generated. The following are the steps for generating the MFCCs of the audio file

- The first step is to find the fourier transform of the audio signal
- The second step for generating the MFCCs is mapping the power to the mel-scale using cosine overlapping windows
- After the above steps the log of the powers of each mel-frequency is obtained
- The next step would be to find the Discrete Fourier Transform of the log of powers at the mel-frequency is
- And now the MFCCs has been obtained in the form of amplitude of the resulting spectrum

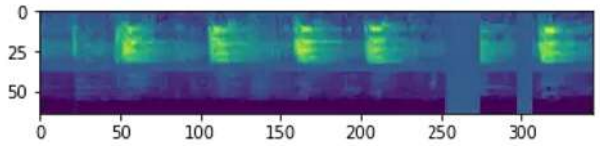

Figure 26: Mel-spectrogram

8 Models

8.1 Existing Model

We will start with constructing a Multilayer Perceptron (MLP) Neural Network using Keras and a Tensorflow backend. We will begin with a simple model architecture, consisting of three layers, an input layer, a hidden layer and an output layer. All three layers will be of the dense layer type which is a standard layer type that is used in many cases for neural networks. The First layer will receive the input shape. As each sample contains 40 MFCCs (or columns) we have a shape of (1x40) this means we will start with an input shape of 40. The First two layers will have 256 nodes. 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. We will also apply a Dropout value of 50% on our first two layers. This will randomly exclude nodes from each update cycle which in turn results in a network that is capable of better generali- sation and is less likely to overfit the training data. Our output layer will have 10 nodes (num_labels) which matches the number of possible classifi- cations. The activation is for our output layer 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.

8.2 Proposed model

The convolution layers are designed for feature detection. It works by sliding a filter window over the input and performing a matrix multiplication and storing the result in a feature map. This operation is known as a convolution. The filter parameter specifies the number of nodes in each layer. Each layer will increase in size from 16, 32, 64 to 128, while the kernel_size

parameter specifies the size of the kernel window which in this case is 2 resulting in a 2x2 filter matrix. The first layer will receive the input shape of (40, 174, 1) where 40 is the number of MFCC's 174 is the number of frames taking padding into account and the 1 signifying that the audio is mono. The activation function we will be using for our convolutional layers is ReLU which is the same as our previous model. We will use a smaller Dropout value of 20% on our convolutional layers. Each convolutional layer has an associated pooling layer of MaxPooling2D type with the final convolutional layer having a GlobalAveragePooling2D type. The pooling layer is do reduce the dimensionality of the model (by reducing the parameters and subsquent computation require- ments) which serves to shorten the training time and reduce overfitting. The Max Pooling type takes the maximum size for each window and the Global Average Pooling type takes the average which is suitable for feeding into our dense output layer. Our output layer will have 10 nodes (num_labels) which matches the number of possible classifi- cations. The activation is for our output layer 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.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_4 (Dense)              (None, 100)               4100

activation_4 (Activation)    (None, 100)               0

dropout_3 (Dropout)          (None, 100)               0

dense_5 (Dense)              (None, 200)               20200

activation_5 (Activation)    (None, 200)               0

dropout_4 (Dropout)          (None, 200)               0

dense_6 (Dense)              (None, 100)               20100

activation_6 (Activation)    (None, 100)               0

dropout_5 (Dropout)          (None, 100)               0

dense_7 (Dense)              (None, 10)                1010

activation_7 (Activation)    (None, 10)                0

=================================================================
Total params: 45,410
Trainable params: 45,410
Non-trainable params: 0
_________________________________________________________________
```

Figure 27: model

9 Results

The final model achieves the classification accuracy of 90% on the testing data while the accuracy given by the benchmark model was 68%. The final solution performs well when presented with a .wav file with a duration of a few seconds and returns a reliable classification. However, this model can not provide the accurate results for real time audio detection. A study has shown an accuracy of 92% for the real time audio event detection with noisy dataset. [5] It has used gammatone filterbank for the feature extraction. Audio features are extracted using a Raspberry Pi edge computing device and those are fed to a local data server to detect audio events.

Model	Classification Accuracy
CNN	92%
MLP	88%
Benchmark SVM_rbf	68%

Figure 28: model

10 Conclusion

It was previously noted in our data exploration, that it is difficult to visualise the difference between some of the classes. In particular, the following sub-groups are similar in shape:

- There were repetitive sounds of air conditioner, jackhammer and machines.
- The peaks of the images obtained from gun shot and barking dog were quite similar.
- Similar pattern was also observed between the children playing and street music.

But, the use of CNN model has improved the performance and the classification accuracy to a better extent.

11 Future Scope

If we were to continue with this project there are a number of additional areas that could be explored:

- As previously mentioned, test the models performance with Real-time audio.
- Train the model for real world data. This would likely involve augmenting the training data in various ways such as:
  - Adding a variety of different background sounds.
  - Adjusting the volume levels of the target sound or adding echos.
  - Changing the starting position of the recording sample, e.g. the shape of a dog bark.
- Experiment to see if per-class accuracy is affected by using training data of different durations.
- Experiment with other techniques for feature extraction such as different forms of Spectrograms.