END SEM REPORT

OF

B.E THESIS ENTITLED

AUDIO CLASSIFICATION USING URBAN SOUND DATASET WITH CNN

UNDER THE SUPERVISION OF Prof. Ritu Sibal

TO

FACULTY OF TECHNOLOGY

UNIVERSITY OF DELHI, DELHI-110007, INDIA

Department of Computer Engineering

NETAJI SUBHAS INSTITUTE OF TECHNOLOGY

(University of Delhi)

SUBMITTED BY:
Ritesh Ranjan Nahak, 2017UCO1580
Kanishk Yadav, 2017UCO1591
Jay Sampartak, 2017UCO1593
Ajay Kumar Meena, 2017UCO1594

Problem addressed

What we want to achieve

Removal of Noise from the sound

Nowadays there are increasing applications of audio analysis and classification in every field. Voice assistants also use audio analysis to understand their commands.

Problem addressed

Identifying the Source of the sound

- Another application motivates us to classify sound samples which is that we can identify the source(device) or the performer of the audio or the song that you are listening to.

- This application is different from identifying the sound or song track from their acoustic fingerprint. Our application requires an analysis of features of sound that represents the characteristics of the source (human or device) of that sound. The automatic classification of environmental sound is a growing research field with multiple applications to large-scale, content-based multimedia indexing, and retrieval.

- The objective of this notebook is to train a CNN model to classify sounds into 10 categories, using the Urban Sound Dataset. There are two main tasks in this project, one being feature extraction, and the other is the training of the neural network architecture on those extracted features.

Brief Discussion of Related Work

Although primarily used in visual recognition contexts, convolutional architectures have been also successfully applied in speech and music analysis .

These efforts have shown that approaches taking advantage of data locality can provide viable solutions to problems encountered in other domains.

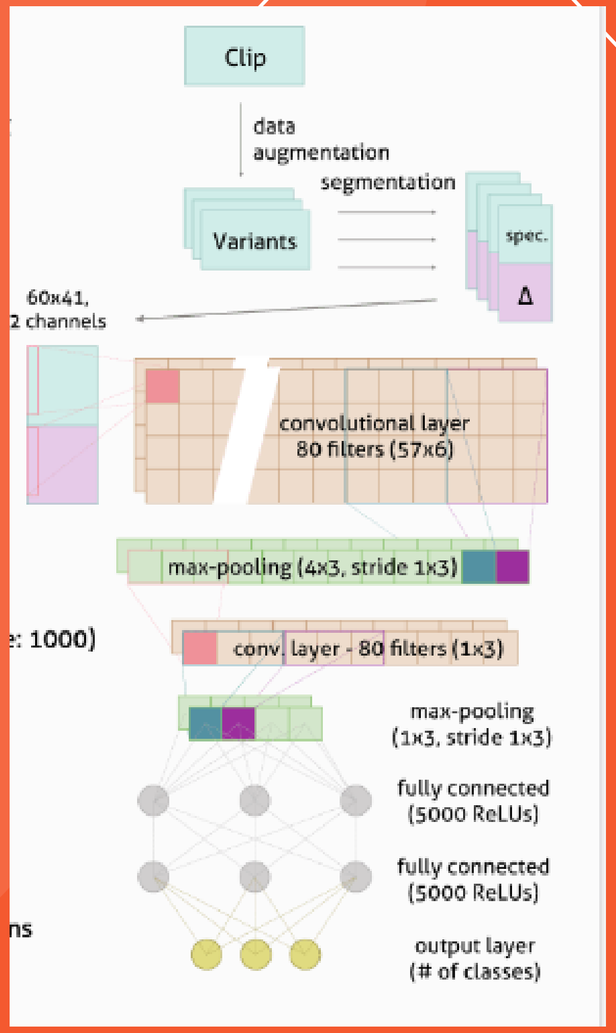
The introduction of deep learning techniques in this context has slowly begun since 2013.

However, these efforts are still mostly limited to analyzing highly pre-processed acoustic features

Related Research Paper

We have followed the research paper *ENVIRONMENTAL SOUND CLASSIFICATION WITH CONVOLUTIONAL NEURAL NETWORKS*, Karol J. Piczak

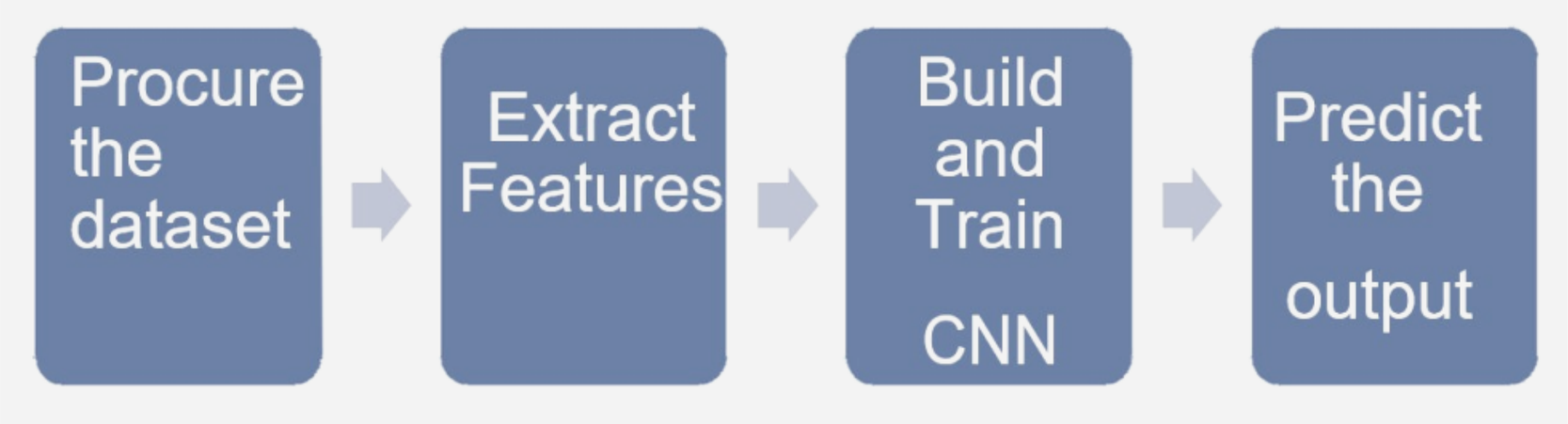
The aim of this work is to evaluate whether this potential of convolutional architecture is also adaptable to other audio tasks- specifically ,classifying short recordings of environmental and urban sound sources



UrbanSound8K:

- 8732 short (<4 seconds) excerpts of various urban sound sources
- 10 classes
- baseline accuracy: 68%

Workflow



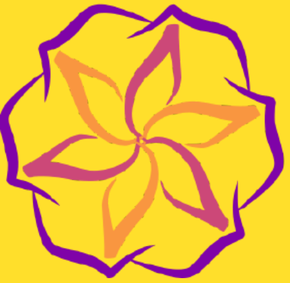
10-fold cross validation using the predefined folds: train on data from 9 of the 10 predefined folds and test on data from the remaining fold. Repeat this process 10 times (each time using a different set of 9 out of the 10 folds for training and the remaining fold for testing). Finally report the average classification accuracy over all 10 experiments (as an average score + standard deviation, or, even better, as a boxplot).

Data Set & Experimental Setup

Dataset

Urbansound8k dataset (8732 audio samples of 4s) categorized into 10 classes:-Air conditioners, car horn, children playing, dog bark, drilling, engine idling, gunshot, jackhammer, siren, and street music.
Dataset – UrbanSound8k Dataset (https://urbansounddataset.weebly.com/)
UrbanSound8k.csv: This file contains meta-data information about every audio file in the dataset. This includes:

- slice_file_name: The name of the audio file. The name takes the following format: [fsID]-[classID]-[occurrenceID]-[sliceID].wav
- fsID: The Freesound ID of the recording from which this excerpt (slice) is taken.
- start: The start time of the slice in the original Freesound recording
- end: The end time of slice in the original Freesound recording.
- salience: A (subjective) salience rating of the sound.
- fold: The fold number (1-10) to which this file has been allocated.
- classID: A numeric identifier of the sound class.



Data Set & Experimental Setup

Experimental Setup

- We have used the Librosa library to convert the clips to a mono(single) channel.
- Keras and TensorFlow
- Jupyter Notebook and Google Colab for executing our code.

Data Preprocessing

- The sounds were sampled and the sampling rate was found to be 22050 samples/sec. The samples are now represented as a 1D array containing float values.
- A few of the signals were decomposed into harmonic and percussive parts.
- Since it is infeasible to use the clips of 4s length as they are as input to the CNN, we needed to reduce the number of features/inputs and bring down the number significantly.

Approach Used & Methodology

Feature Extraction from the Data

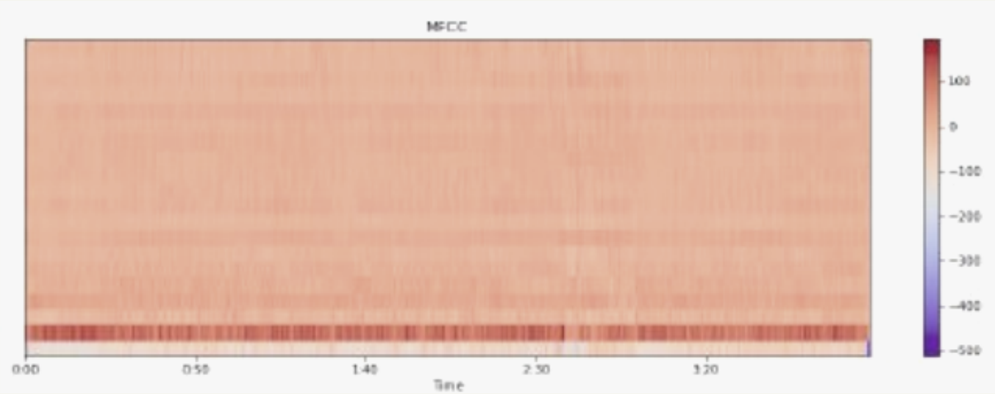
We will extract features of sound files that will help us to classify sound files into different types.

Based on the analysis of the data our objective here is to obtain features for the dataset which can be used to train our neural network architecture.

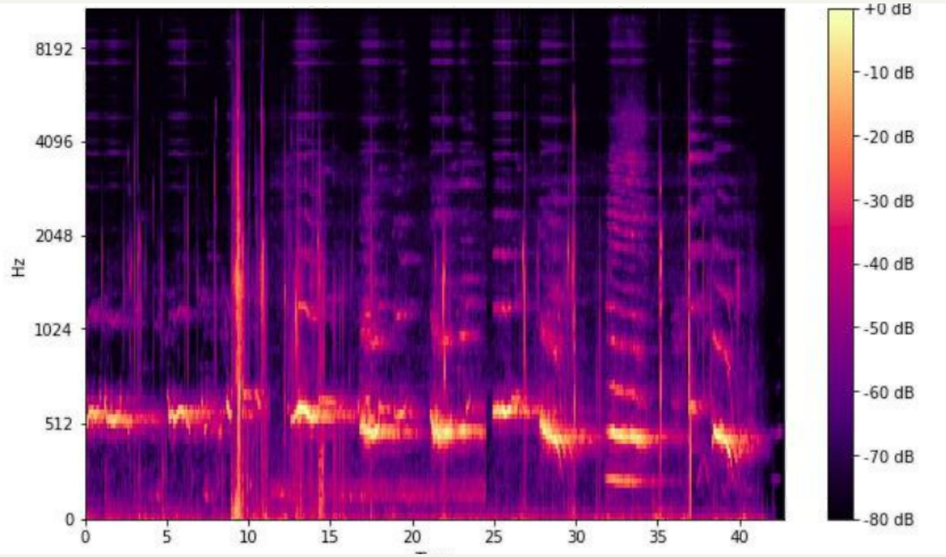
Another objective in this part is to only consider those features that are based on how humans perceive sound information.

The major features used in the CNN are :

- MFCC: Mel Frequency Cepstral Coefficients
- Mel-Spectrogram: The Mel-Spectrogram is a spectrogram in which the y-axis is the Mel - scale and the x-axis is time.
- Chroma-stft: STFT stands for short term Fourier transform. The STFT represents a signal in the time-frequency domain by computing discrete Fourier Transforms.



Spectrogram of MFCCs of Dog's bark



Mel Spectrogram of Dog's bark

Approach Used & Methodology

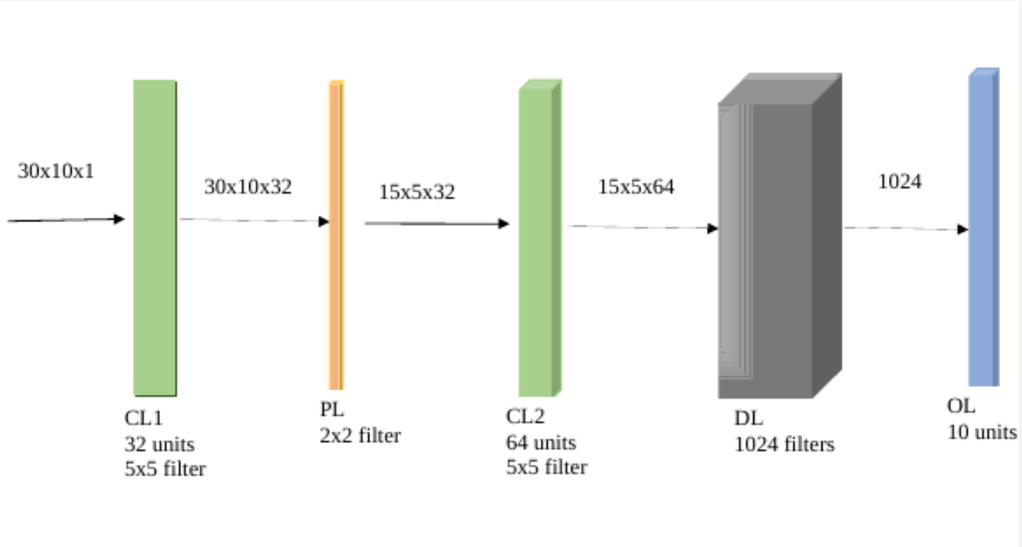
Training CNN Architecture

Using the features extracted we want to train a convolutional neural network and try and refine the architecture and the parameters such as

- Number of layers
- Optimizer used
- Learning rate
- Loss Function

Architecture I

The first step in the training of CNN is the initialization of weights and biases with random values using a normal distribution. This is the model initialization according to the first architecture:



Architecture II Model Diagram

```
def build_model():

    model = Sequential()

    f_size = 3
    model.add(Convolution2D(24, f_size, f_size, border_mode='same', input_shape=(bands, frames, num_channels)))
    model.add(MaxPooling2D(pool_size=(2, 2)))
    model.add(Activation('relu'))
    model.add(Convolution2D(48, f_size, f_size, border_mode='same'))
    model.add(Activation('relu'))
    model.add(Convolution2D(48, f_size, f_size, border_mode='valid'))
    model.add(Activation('relu'))
    model.add(Flatten())
    model.add(Dense(64, W_regularizer=l2(0.001)))
    model.add(Activation('relu'))
    model.add(Dropout(0.5))
    model.add(Dense(num_labels, W_regularizer=l2(0.001)))
    model.add(Dropout(0.5))
    model.add(Activation('sigmoid'))
    sgd = SGD(lr=0.001, momentum=0.9, decay=0.0, nesterov=True)
    model.compile(loss='categorical_crossentropy', metrics=['accuracy'], optimizer=sgd)

    return model
```

Initially, we started with 2 convolutional layers ,one pooling and one dense layer

Approach Used & Methodology

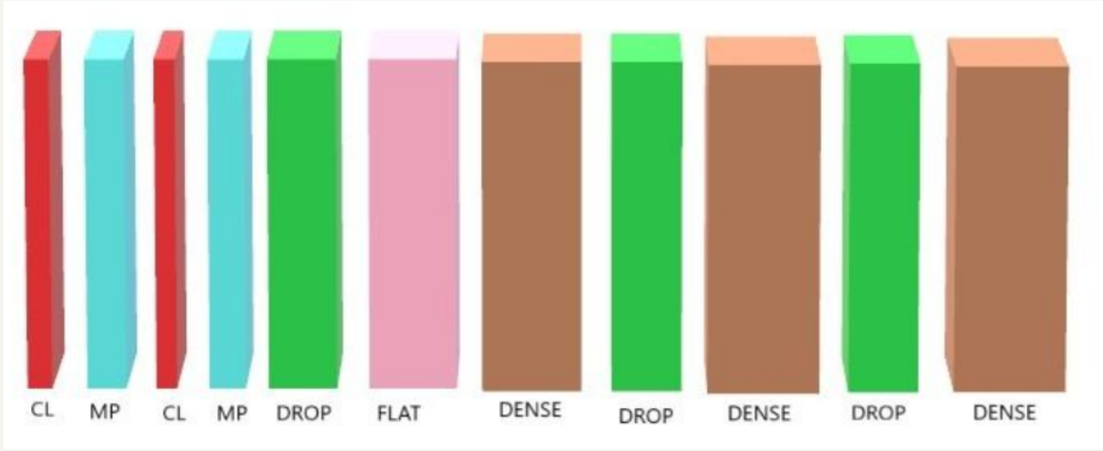
Architecture Refinement

- Optimizers are used to update weights by using gradient descent and propagating the error backward into previous Layers (Back Propagation).
- The number of epochs is decided by experiment.
- The following parameters are fine-tuned for the CNN architecture :
  - Number of conventional Layers
  - Filter size
  - Max pooling size
  - Early Stopping and Checkpointing Callback in Keras

Architecture II

```
#adding layers and forming the model
model.add(Conv2D(64,kernel_size=5,strides=1,padding="Same",activation="relu",input_shape=(40,5,1)))
model.add(MaxPooling2D(padding="same"))

model.add(Conv2D(128,kernel_size=5,strides=1,padding="same",activation="relu"))
model.add(MaxPooling2D(padding="same"))
model.add(Dropout(0.3))

model.add(Flatten())

model.add(Dense(256,activation="relu"))
model.add(Dropout(0.3))

model.add(Dense(512,activation="relu"))
model.add(Dropout(0.3))

model.add(Dense(10,activation="softmax"))
```

Code snippet for model initialization(Architecture II)



Architecture II Model Diagram

After many rounds of exploration and experiments, we found out that a new architecture with the 2 convolutional layers, 2 max-pooling layer, 3 dense layer was optimal for this project.

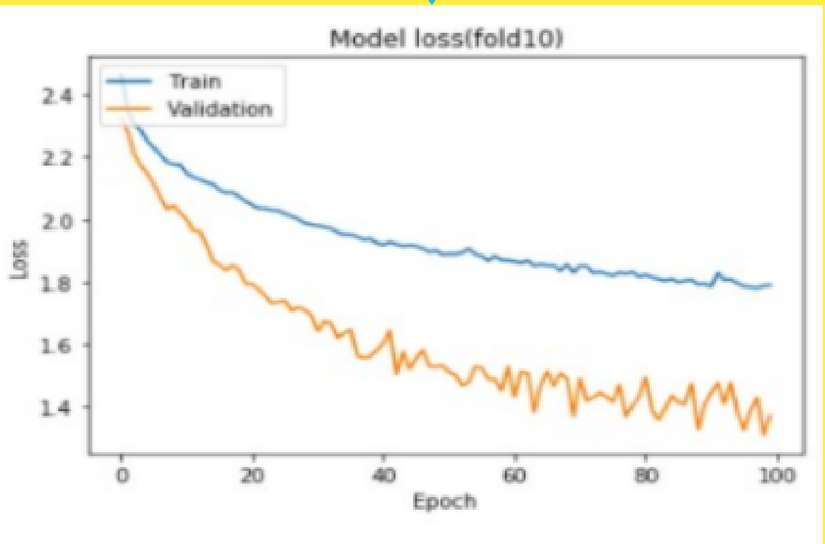
Results

ARCHITECTURE I

Average Accuracy with 100 epochs = 64.487 %

Average Accuracy with 150 epochs = 65.587%

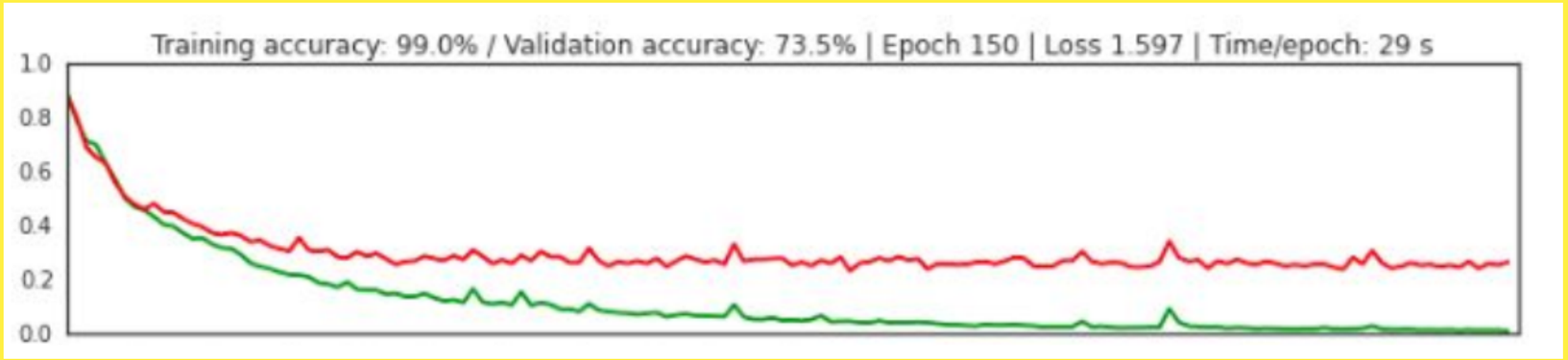
(The training & testing of this architecture was done using 10 fold cross validation)

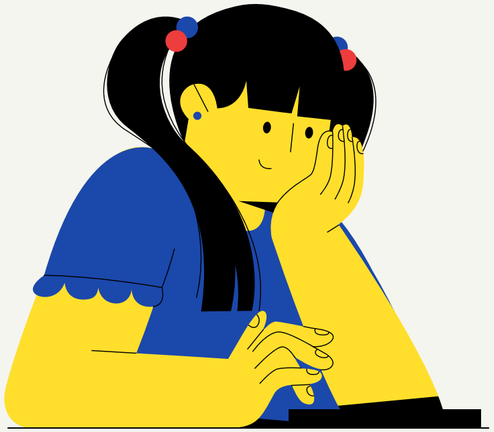


ARCHITECTURE II

Accuracy score: 73.246% (With 150 epochs)

(The training & testing of this architecture was done using 10 fold cross validation)





Challenges

- With large datasets, the training of the model was a time-consuming task. It took hours to train the whole data.

- Lack of labelled audio data posed a problem if we can get bigger datasets we can improve the accuracy of the model.

Plagiarism Report

Thesis Report

ORIGINALITY REPORT

11%	8%	6%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	towardsdatascience.com Internet Source	2%
2	machinelearningmastery.com Internet Source	2%
3	analyticsblog.ravivk.com Internet Source	2%
4	Bhupesh Kumar Mishra, Dhavalkumar Thakker, Suvodeep Mazumdar, Daniel Neagu, Marian Gheorghe, Sydney Simpson. "A novel application of deep learning with image cropping: a smart city use case for flood monitoring", Journal of Reliable Intelligent Environments, 2020 Publication	1%
5	Submitted to Mahidol University Student Paper	<1%
6	Submitted to Indian Institute of Technology, Madras Student Paper	<1%
7	Joy Krishan Das, Arka Ghosh, Abhijit Kumar Pal, Sumit Dutta, Amitabha Chakrabarty. "Urban Sound Classification Using Convolutional Neural Network and Long Short Term Memory Based on Multiple Features", 2020 Fourth International Conference On Intelligent Computing in Data Sciences (ICDS), 2020 Publication	<1%
8	Cichocki. "Multiplicative Iterative Algorithms for NMF with Sparsity Constraints", Nonnegative Matrix and Tensor Factorizations, 09/18/2009 Publication	<1%
9	Submitted to University of Warwick Student Paper	<1%
10	librosa.github.io Internet Source	<1%
11	Xun Jiao, Vahideh Akhlaghi, Yu Jiang, Rajesh K. Gupta. "Energy-efficient neural networks using approximate computation reuse", 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE), 2018 Publication	<1%
12	Submitted to Pusan National University Library Student Paper	<1%
13	Submitted to University of Northumbria at Newcastle Student Paper	<1%
14	link.umsl.edu Internet Source	<1%
15	www.justinsalamon.com Internet Source	<1%
16	Santanu Pattanayak. "Pro Deep Learning with TensorFlow", Springer Science and Business Media LLC, 2017 Publication	<1%
17	export.arxiv.org Internet Source	<1%

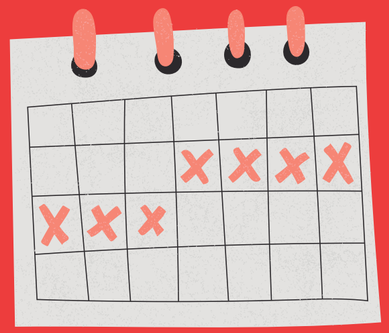
Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		

References

HELPFUL BOOKS AND ARTICLES

- Dataset – UrbanSound8k Dataset (https://urbansounddataset.weebly.com/)
- K.J. Piczak, “Environmental Sound Classification with Convolutional neural networks Proc. 25th Int. Workshop Mach. Learning Signal Process., pp. 1-6, Sep. 2015: This paper helps us understand how we can apply cnn to various public datasets of audio recording.It discusses the basic parameters such as number of units, Activation function used etc.
- J. Salamon, C. Jacoby and J.P. Bello, “A Dataset and Taxonomy for Urban Sound Research”, Proceedings of the 22nd ACM International Conference on Multimedia, November 03-07, 2014, Orlando, Florida, USA: This paper introduces the classification of environmental sound and its applications
- J. Salamon, J.P.Bello, “Unsupervised Feature Learning for Sound Classification”. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, Australia, April 2005
- Salamon Justin and Juan Pablo Bello .” Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification”, Proceedings of the 22nd ACM International Conference on Multimedia, November 03-07-2014, Orlando, Florida, USA: This paper proposes another cnn architecture that can be used for sound classification. This also introduces the approach of data augmentation to resolve the issue of the unavailability of datasets to train on.
- S. Chu, S.Narayanan, C- C. Kuo, “Environmental Sound Recognition with time and frequency audio features”, IEEE Trans, Audio Speech Language Process, vol. 17, No.6 pp. 1142- 1158, Aug 2009.
- Y. Kumar, M.Vyas, S.Garg.” From Image Classification to Audio Classification”.

Project Timeline



01

August

Choose the topic and did Academic Reasearch on it

02

September

Data preprocessing & Feature Extraction

03

October

CNN Training Architecture 1

04

November

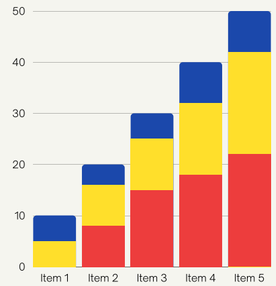
CNN Refinement Architecture 2

05

December

BTP Mid Term Evalutaion

Future Scope



The deep neural networks proposed in this report are dependent on the availability of large training sets to learn a nonlinear function and generalize, scarcity of labeled data for audio classification poses a major problem.

With the improvement in the datasets, the accuracy of the model can be increased.

A possible solution is data augmentation, which allows us to create more training samples by applying deformations on the dataset, by training on the deformed data we can expect the model to generalize better on unseen data. Few possible augmentation techniques that can be applied are time-shifting, pitch shifting, and time stretching.

We can use these models for outdoor environmental sound recognition. This can be used to detect “danger” situations (using sound in a video surveillance system). They also allow us to detect the type of environment, location in which the audio has recorded the use of pre-trained models like VGG, ResNet-50 can reduce the model building time.

Thanks for being here!