A Project Report on

New-born's Cry Analysis using Machine Learning Algorithm

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

in

Information Technology

by

Parvi Agrawal(18104018)
Vaishnavi Sriramoju(18104014)
Manish Kumar(18104035)

Under the Guidance of

Prof. Kiran Deshpande
Prof. Nahid Shaikh

Department of Information Technology
NBA Accredited
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2021-2022

Approval Sheet

This Project Report entitled ***"New-Born's Cry Analysis using Machine Learning"*** Submitted by ***"Parvi Agrawal (18104018),Vaishnavi Sriramoju (18104014),Manish Kumar (18104035)"***is approved for the partial fulfillment of the requirenment for the award of the degree of ***Bachelor of Engineering*** in ***Information Technology*** from ***University of Mumbai***.

(Prof. Nahid Shaikh)
Co-Guide

(Prof. Kiran Deshpande)
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***"New-Born's Cry Analysis using Machine Learning"*** submitted by ***"Parvi Agrawal (18104018),Vaishnavi Sriramoju (18104014),Manish Kumar (18104035)"*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Information Technology***,to the University of Mumbai,is a bonafide work carried out during academic year 2021-2022.

(Prof. Nahid Shaikh)
Co-Guide

(Prof. Kiran Deshpande)
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Acknowledgement

We have great pleasure in presenting the report on **New-Born's Cry Analysis using Machine Learning.** We take this opportunity to express our sincere thanks towards our guide **Prof. Kiran Deshpande** & Co-Guide **Prof. Nahid Shaikh** Department of IT, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Kiran B. Deshpande** Head of Department,IT, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Vishal S. Badgujar** BE project co-ordinator, Department of IT, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

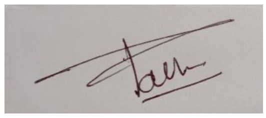
Student Name1: Parvi Agrawal
Student ID1: 18104018

Student Name2: Vaishnavi Sriramoju
Student ID2: 18104014

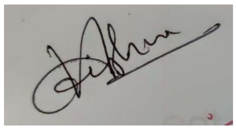
Student Name3: Manish Kumar
Student ID3: 18104035

Declaration

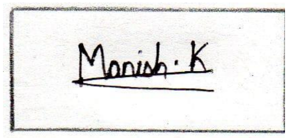
We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.



(Parvi Agrawal - 18104018)



(Vaishnavi Sriramoju - 18104014)



(Manish Kumar - 18104035)

Date:

Abstract

Crying is a newborn's main way of telling you what they need. It's a sound that can spur you into action, even when you're asleep. Newborns usually spend 2 to 3 hours a day crying. Normal as it may be, a bawling baby can be distressing for infants and parents alike. The analysis of infant cry has been the subject of research for many years. In this paper, we classify the newborn's cry signals into five categories: hunger, anger, discomfort, tiredness and belly pain. The proposed system involves preprocessing the cry signal and finding the MFCC coefficients followed by feeding the retrieved data into CNN for training and classification. Experimental results shows 72% that the proposed method achieves high classification accuracy.For a new-born, the first communication with the outside world is in the form of crying of the infant. Crying plays a crucial role in ensuring the survival, health and development of the infant. The most important aspect of taking up this project is to help unskilled parents/trainee paediatricians/babysitters who would be aware of their baby's needs. Therefore, it is important to understand the reason behind the cry. According to Dunstan's theory, before crying, the babies try to communicate their needs using a "special language" that consists of five "words" (or specific utterances) associated with five basic needs (like being hungry, tiredness etc.). The theory states that these five utterances are universal and innate. This research develops a system to classify the infants cry sound and focuses on generating an automated and non-invasive platform to monitor baby's status by performing feature extraction using MFCC (Mel-Frequency Cepstrum Coefficients); the coefficients obtained from the feature extraction would be normalized and will be sent as input to the classifier: Convolutional Neural Network (CNN) for classification based on audio type. It will be classified into hungry, tiredness, low belly-pain, anger and physical discomfort and provide a suitable solution.

Keywords: Newborn's cry analysis, MFCC, CNN.

Contents

List of Figures

List of Tables

List of Abbreviations

MFCC: Mel Frequency Cepstrum Coefficient
CNN: Convolutional Neural Network
ANN: Artificial Neural Network
FFT: Fast Fourier Transform
IDE: Integrated Development Editor
RMSE: Root Mean Square Energy
FIR: Finite impulse response
DCT: Discrete Cosine Transform
CONV2D: Two-dimensional convolutional layers

Chapter 1

Introduction

For infants, crying is one of few ways to communicate or express how they feel or need to their parents. Because newborns may cry for several reasons, this may make new parents feel tough to handle or respond to their baby's need. Generally infants are incapable of expressing their physical, physiological and psychological status to their surrounding environment as they cannot communicate with words. The cry of baby cannot be predicted accurately as it is very hard to identify for what it cries for. Experienced parents and specialists in the area of child care such as paediatrician and paediatric nurse can distinguish different sort of cries by just making use their individual perception on auditory sense. This is totally subjective evaluation and not suitable for clinical use. Non-invasive method has been widely used in infant cry signal analysis and has shown very promising results. Various feature extraction and classification algorithms used in infant cry analysis have been mentioned in the referred research papers. The main reason is because it is difficult to understand the meaning of the infant cries. So our aim is to generate an automated and non-invasive platform to monitor baby's status to diagnose their emotional status. Many researches have been done regarding feature analysis the infant cry but the most accurate for feature extraction is MFCC and also for classification of infant cries use of different types of ANN have been used. Experimental results show that CNN based deep learning achieves high performance of 84% [1]. Our system will incorporate preprocessing including frame blocking and windowing followed by MFCC (Mel-Frequency Cepstrum Coefficients) feature extraction whose coefficients will be normalized [4] and sent to CNN (Convolutional Neural Network) for classification based on voice type and will classify the infant crying sounds into five classes: hunger, anger, Discomfort, tired and belly Pain. Coefficients of MFCC are obtained by performing FFT (Fast fourier transform) which converts time domain to the frequency domain.FFT is performed in order to obtain the magnitude frequency of each frame. The coefficients of MFCC are then normalized and sent to Convolutional Neural Network for further classification. The output neurons will depict the classes and hence the infant cry sounds can be classified into five classes namely hungry,tired,discomfort, belly pain and anger.

Chapter 2

Literature Review

- In paper [1], MFCC is used and its secondary characteristics derived from different newborn-crying signals to classify into three types; hungry, sleepy and burping. From table 3, CNN can learn information from the complicated MFCCs spectra with a total accuracy of 84accuracy is 80%, 80% and 92% for burping, hungry and sleepy respectively. For future work, more data need to be collected to enhance classification accuracy and consistency, which will be integrated into the infant care system.

- In paper [2], we describe the significant research work in infant cry analysis and classification, providing details and resources that are helpful for both researchers and medical professionals who work in this area. It is shown that the limited database resources hinder the development of the infant cry research. Large databases with diverse samples fitting the need of deep neural networks is imperatively desired. The current tendency for feature extraction is to generate a mixed feature set and takes advantages of different domains to achieve better discriminating ability. The relevant research results show promising improvement with combined features. In addition, new neural network-based architectures become the mainstream methods. It proves better robustness and performance than traditional machine learning approaches. In the future, we are interested in creating a large database, extracting more robust features, combining features with good ratio, establishing novel neural network architectures with the use of prior knowledge as well as other space information from interdisciplinary areas.

- In paper [3], the convolutional neural networks is adopted to train the infant crying data. Accordingly, the trained CNN is capable to classify the crying into hungry, pain, and sleepy. The proposed system assists parents to understand the demand of the infants. After running 25000 training iterations, the network achieves 78.5% validation accuracy. We can found in our testing process, CNN successfully distinguish between different spectrograms. Experimental results showed the proposed method achieves high classification accuracy. In the future, we will continue collect more data to promote classification accuracy.

- In paper [4], they have presented and discussed the results of three cry sound recognition methods using the MFCC and ML classifiers such as multi-class support vectors machines, fully connected feed-forward neural networks, and one- dimensional convolutional neural networks for indoor cry sound monitoring applications. Evaluation results showed that the 1D-CNN with frame length of 500 ms provides promising results as compared to that of the frame lengths 100 ms and 250 ms. The 1D-CNN based based method had class-wise F1-score of above 98%. Our preliminary results demonstrated that the CNN based cry sound recognition has a great potential applications in cry sound pattern analysis. In future directions, we attempt to implement our methods on real-time hardware for edge computing applications.

Objectives

- To help parents, nannies or paediatrician who don't have any prior experiences, especially for orphanages and young parents who will feel uncomfortable when the infant is crying.

- To generate dataset from Baby language website like Dunstan baby language, Donate A Cry Corpus and from other sources like through parents hospitals as self-recorded datasets.

- To perform feature extraction noise threshold detection.

- To classify cries using CNN algorithm into hungry, angry, tired, low belly pain physical discomfort.

- To create a web portal using Django Framework for users to provide a solution based on the type of cry.

Chapter 3

Project Design

3.0.1 Existing System

- Taking good care of new-born is a big challenge, especially for first time parents/ nannies/ paediatrician. Infant communication is the only means from which the emotional status of the infant can be identified.
- Young parents get frustrated and have trouble calming down their babies because all cry signals sound the same to them.
- In the early years, researches have determined that different types of cries can be differentiated auditory by trained adult listeners. But training human perception for infant cry is much harder than training machine learning models.

3.0.2 Proposed System

From Figure 3.2, the proposed system consists of the Input audio files which are in .wav format which are the infant cry sounds. These cry sounds are then sent for preprocessing which involves the frame blocking and windowing. The frame blocking is the process of breaking down the signal into frames. Signal analysis is done on short span periods of time which are called frames to get the accurate values. Typical values for frame blocking are dividing the frames into 115 samples and taking the overlapping length as 1s. This signal is then passed onto for windowing. Windowing is basically done in order to reduce the distortion of the signal. Finally the signal is then multiplied with the hamming windowing function and this windowing signal serves as an input to calculate the Mel Frequency Cepstrum coefficients. Coefficients of MFCC are obtained by performing FFT (Fast fourier transform) which converts time domain to the frequency domain. FFT is performed in order to obtain the magnitude frequency of each frame. The coefficients of MFCC are then normalized and sent to Convolutional Neural Network for further classification. The CNN takes the MFCC as the inputs and does the classification of the cries. More the number of coefficients, better the system will be able to classify the data. CNN is a supervised learning algorithm that typically has one input layer, one hidden layer, and one output layer. The number of iterations in the training process is also influenced by the use of neurons in the hidden layer. As a result of the output neurons displaying the classes, the infant cry noises may be divided into five categories, as indicated in figure 3.1, namely hunger, discomfort, anger, tiredness, and belly pain.

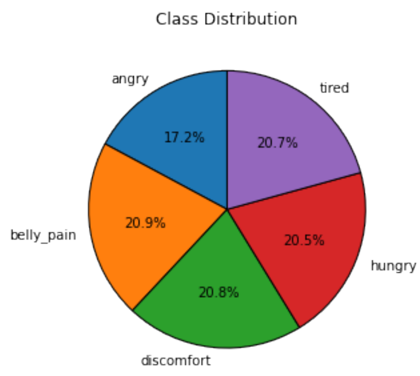


Figure 3.1: Class Distribution

Programming language used here is python and Google Collab as our Integrated Development Editor (IDE) are used for code and for matrix calculation and display, use key modules such as pandas, Numpy, Scipy, and Matplotlib. Librosa is a programme that allows you to process audio and calculate features. Tensor Flow for classification model and Tqdm for calculating the features required for model analysis.

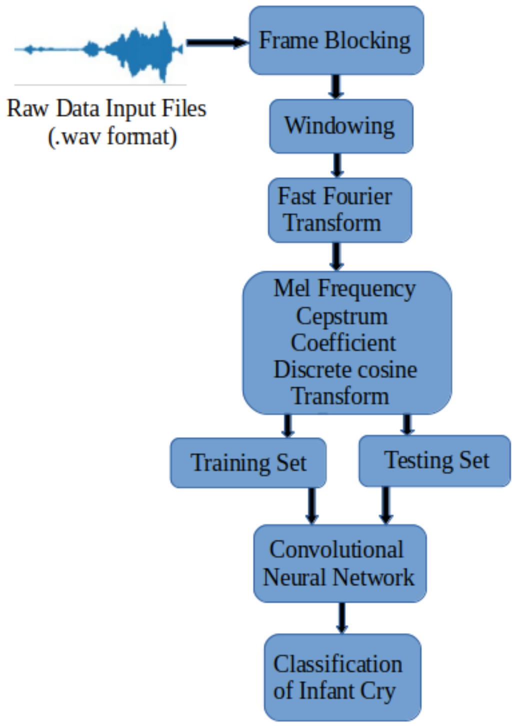


Figure 3.2: Proposed Method Flow

3.0.3 System Diagram

- Activity Diagram

The activity diagram shown is for the working of the proposed system, First the input audio data would be preprocessed and then Mfcc coefficients would be extracted which would be sent for classification to the Convolutional Neural Network . Higher the Mfcc coefficients greater the accuracy, The classification of three types of cries:hungry,anger and fear woud be evaluated and the output will be containing of the audio,text and image solution based on the type of infant cry.

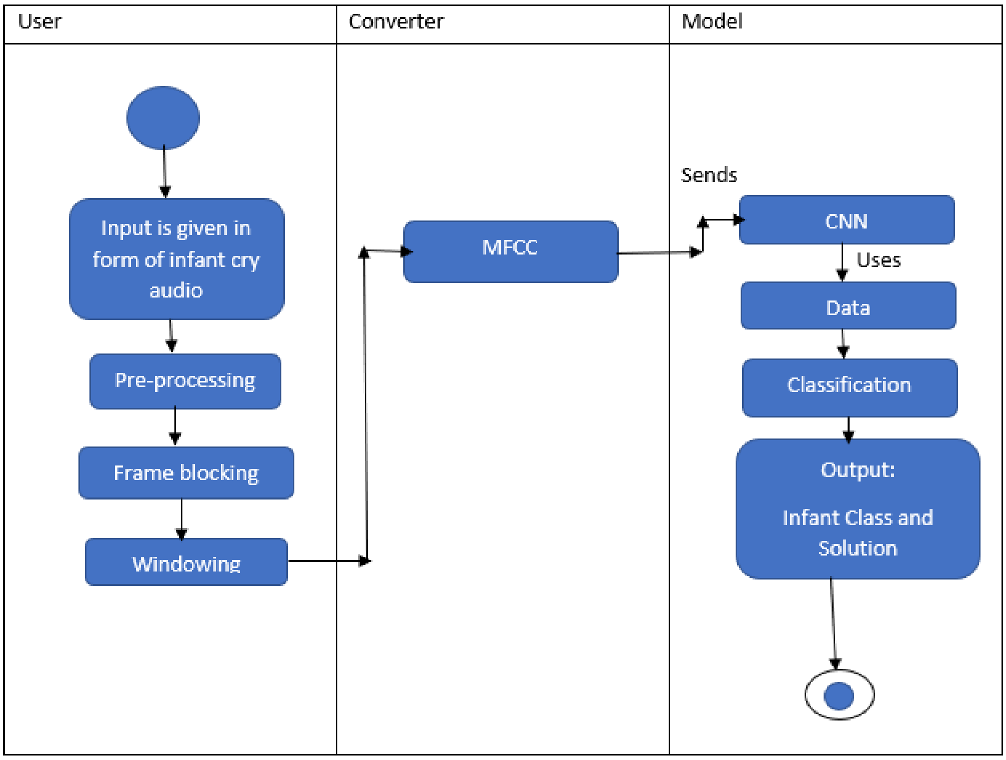


Figure 3.3: Activity Diagram For Infant Cry Analysis

- Use Case Diagram

The Use Case Diagram below shows different actors as Pediatricians, Nannies and Untrained Parents. The relation between actors and what they can do with the system. The classification of three types of cries:hungry,anger and fear would be evaluated and the output will be containing of the audio,text and image solution based on the type of infant cry. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements.

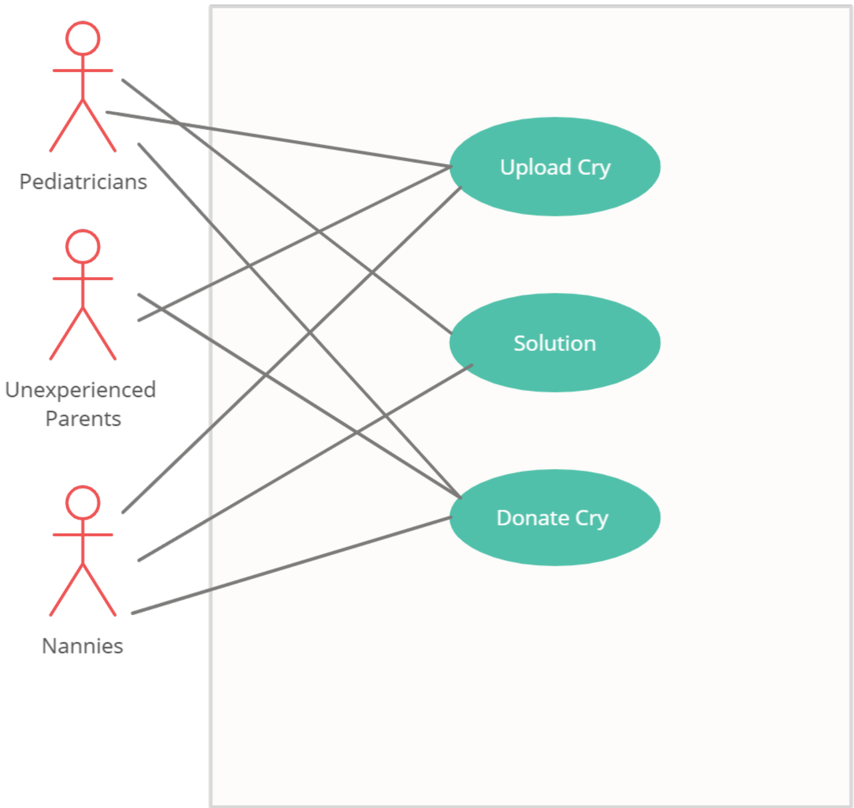


Figure 3.4: Use Case Diagram For Infant Cry Analysis

- Sequence Diagram

The sequence diagram below shows how objects operate with one another and in what order in our system. First the input audio data would be pre-processed using frame blocking and windowing and then Mfcc coefficients would be extracted which would be sent for classification to the Convolutional Neural Network. Higher the Mfcc coefficients greater the accuracy, The classification of three types of cries:hungry,anger and fear would be evaluated and the output will be containing of the audio,text and image solution based on the type of infant cry.

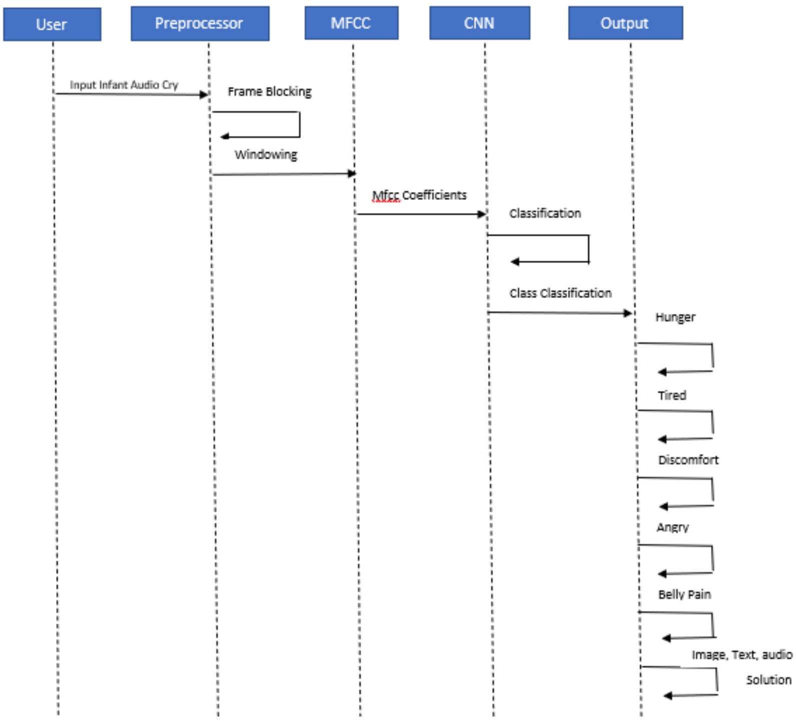


Figure 3.5: Sequence Diagram For Infant Cry Analysis

Chapter 4

Project Implementation

4.0.1 Code Snippets

Imports Requirements.

```
import sys
from warnings import catch_warnings


from cfg import Config

import os
from tqdm import tqdm
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt


from scipy.io import wavfile
from python_speech_features import mfcc
import librosa

from keras.layers import Conv2D, MaxPool2D, Flatten, LSTM
from keras.layers import Dropout, Dense, TimeDistributed
from keras.models import Sequential
from tensorflow.keras.utils import to_categorical
from sklearn.utils.class_weight import compute_class_weight

import pickle
from keras.callbacks import ModelCheckpoint
```

Here, fn. is used to perform feature extraction from a small segment of the audio file.

```
def build_rand_feat():
    tmp = check_data()
    if tmp:
        return tmp.data[0], tmp.data[1]
    X = []
    y = []
    _min, _max = float('inf'),-float('inf')
    for _ in tqdm(range(n_samples)):
        rand_class = np.random.choice(class_dist.index, p = prob_dist)
        file = np.random.choice(df[df.label == rand_class].index)
        rate, wav = wavfile.read(Folderpath + 'clean//'+ file)
        label = df.at[file, 'label']
        rand_index = np.random.randint(0, wav.shape[0]- config.step)
        sample = wav[rand_index:rand_index+config.step]

        X_sample = mfcc(sample, rate, numcep= config.nfeat, nfilt= config.nfilt, nfft= config.nfft)
        _min = min(np.amin(X_sample),_min)
        _max = max(np.amax(X_sample),_max)
        X.append(X_sample)
        y.append(classes.index(label))
    config.min = _min
    config.max = _max
    X, y = np.array(X), np.array(y)
    X = (X - _min)/ (_max - _min)

    X = X.reshape(X.shape[0], X.shape[1], X.shape[2], 1)
    y = to_categorical(y, num_classes=5)
    config.data = (X,y)
    try:
        with open(config.p_path, 'wb') as handle:
            pickle.dump(config, handle, protocol = pickle.HIGHEST_PROTOCOL)
            #return tmp
    except(FileNotFoundError) as e :
        print(e)
        pass
    return X,y
```

Convolution model and its architecture.

```
def get_conv_model():
    model = Sequential()
    model.add(Conv2D(16, (3, 3), activation='relu', strides = (1, 1), padding = 'same', input_shape = input_shape))
    model.add(Conv2D(32, (3, 3), activation='relu', strides = (1, 1), padding = 'same'))
    model.add(Conv2D(64, (3, 3), activation='relu', strides = (1, 1), padding = 'same'))
    model.add(Conv2D(128, (3, 3), activation='relu', strides = (1, 1), padding = 'same'))
    model.add(MaxPool2D((2, 2)))
    model.add(Dropout(0.5))
    model.add(Flatten())
    model.add(Dense(128, activation='relu'))
    model.add(Dense(64, activation='relu'))
    model.add(Dense(5, activation='softmax'))
    model.summary()
    model.compile(loss='categorical_crossentropy', optimizer = 'adam', metrics=['acc'])
    return model
```

Loading of pickle, .model files for classifying the given input audio file.

```python
FILEPATH = os.path.join(PARENT_DIR, 'clean', file)

classes = ['angry', 'belly_pain', 'discomfort', 'hungry', 'tired']
p_path = os.path.join(PARENT_DIR, 'pickles', 'conv.p')
print(p_path)
config = None
with open(p_path, 'rb') as handle:
    config = pickle.load(handle)

model = load_model(os.path.join(PARENT_DIR, 'model', 'conv.model'))

y_pred, fN_pE = predict_cry(config, FILEPATH)
print("Predicted result for filepath %s is %s"%(FILEPATH, classes[y_pred]))
print()
print("Individual probability scores:")
for classN, proba in zip(classes, fN_pE[FILEPATH]):
  print("%s:\t\t%.2f"%(classN, proba))

return classes[y_pred]
```

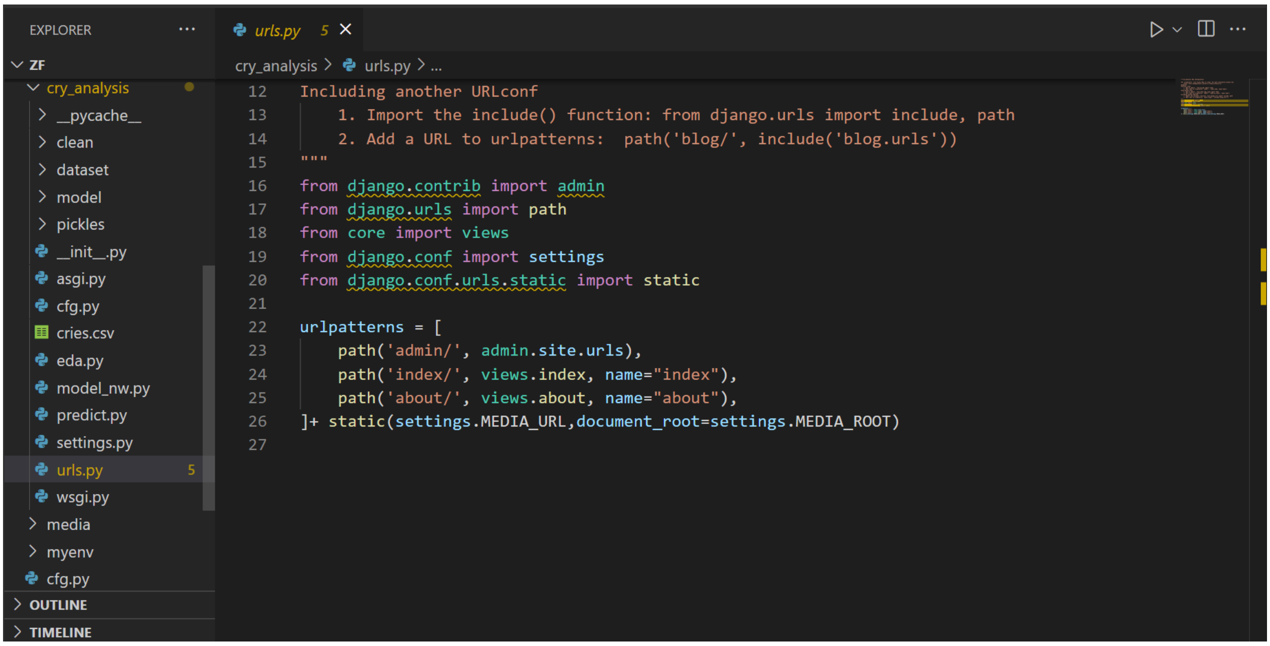
In views.py, respective functions have been defined where index function includes form validation, passing the audio file to the model and retrieval of result.

```python
from django.http import HttpResponse
from core.demo import something
import cry_analysis.predict as model
# Create your views here.
def index(request):
    result=False
    if request.method == 'POST':
        form = AudioForm(request.POST,request.FILES or None)
        if form.is_valid():
            form.save()
            result=True
            filename = str(request.FILES['record'])
            print("filename:", filename)
            output=model.main(filename)

            return render(request, 'index.html', {'form' : form, 'result': result, 'ou
    else:
        form = AudioForm()
    return render(request, 'index.html', {'form' : form, 'result': result})


def about(request):
    return render(request, 'about.html')
```

We have defined paths in urls.py for index, about and support pages



Chapter 5

Testing

5.1 Functional Testing

5.1.1 Unit Testing

Unit testing is the first level of testing, which is typically performed by the developers themselves. At the code level, it is the process of ensuring that individual components of software are functional and work as intended. Unit testing can be done manually, however automating the process will reduce delivery times and boost test coverage. Because flaws will be detected earlier in the testing process and will take less time to fix than if they were discovered later, debugging will be easier as a result of unit testing. It helped us understand the desired output of each module, which we had broken down into separate units.

Chapter 6

Result

6.0.1 Datasets

We have taken out infant data from Donate a cry corpus, Dunstan Baby Language and some are self-recorded from few sources. We have acquired a list of available datasets that are available globally out of which we currently have the "Donate a cry Corpus" dataset. Audio features of infant cry signals were obtained in time and frequency domains, and were used to perform infant cry language recognition. The proposed system consists of the Input audio files which are in .wav format which are the infant cry sounds. The following data is divided as 80% training set and 20% testing set. A total of five categories are being considered: Hunger, Anger, Belly Pain, Discomfort and Tiredness. Total no. of datasets used are 115 and distribution is shown in figure 6.1:

	Angry	**Hungry**	**Belly Pain**	**Discomfort**	**Tired**
Train	12	22	12	20	14
Test	6	8	4	7	10
Total	18	30	16	27	24

Table 6.1: Dataset Distribution

6.0.2 Pre-processing

After accumulating newborn-crying signal, these cry sounds are then sent for pre- processing which consists of the frame blocking and windowing. It is also an important step in data mining as we cannot work with raw data. The quality of the data should be checked before applying machine learning or data mining algorithms. Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format. Data preprocessing is essential before its actual use. Data preprocessing is the concept of changing the raw data into a clean data set. The dataset is preprocessed in order to check missing values, noisy data, and other inconsistencies before executing it to the algorithm. This procedure employ a union of short-time methods including Energy Banks as well as Root Mean Square Energy (RMSE), with the backdrop of 10 s frame length and 1 s overlapping. Since the baby cry duration is of 5-10 seconds, perceptible threshold values equals to 0.005. Downsampling results are shown in figure 6.2.

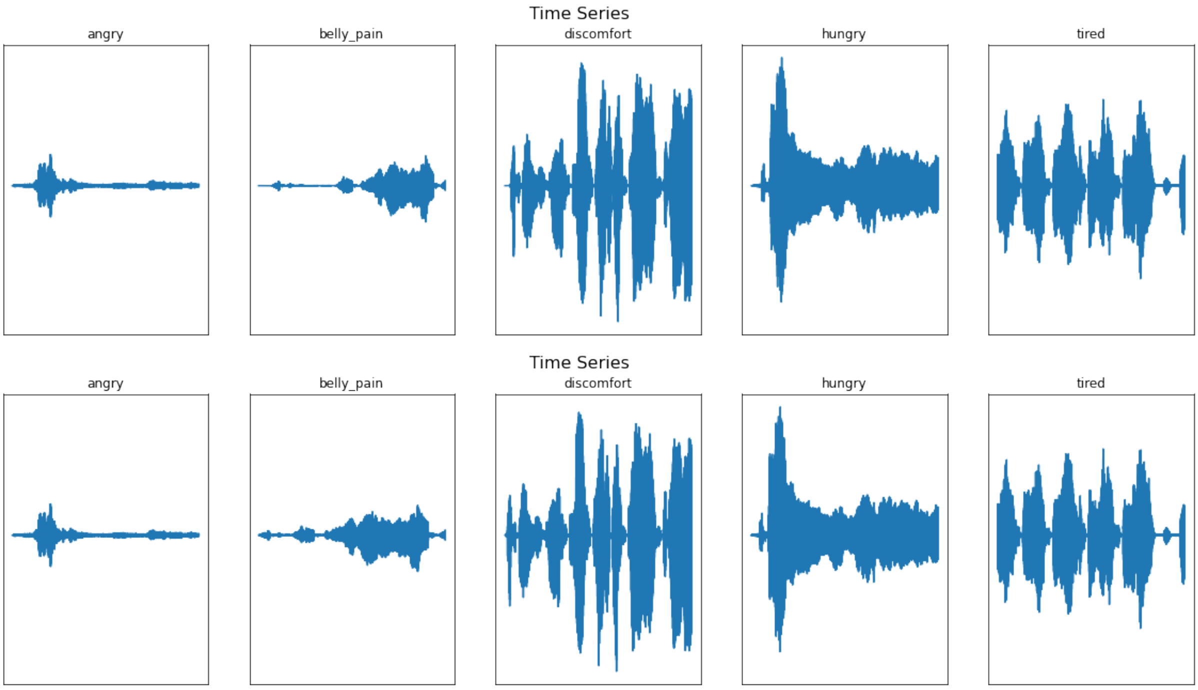


Figure 6.1: Data Before & After Downsampling

6.0.3 Mel-Frequency-Cepstrum Coefficient

Ultimately, the signal is then multiplied with the hamming windowing function and this windowing signal serves as an input to calculate the Mel Frequency Cepstrum coefficients. Remove the leftover signal time duration if it exceeds the starting point; otherwise, pad the signal with zero. The Librosa module's MFCC method is then used to compute variable by setting value of Mel Frequency Cepstrum Coefficient, value of Fast Fourier Transform, and no. of filters to 13, 1103, and 26 correspondingly. Aside from that, delta feature approaches are used to boost the amount of vital data in MFCC. Figure 6.3 depicts the MFCC results.

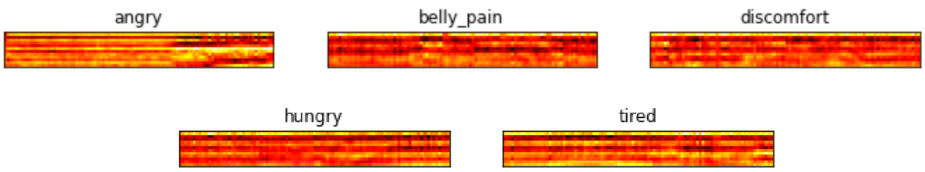


Figure 6.2: MFCC for Cry Categories

Obtained the coefficients of MFCC by executing FFT which is done in order to obtain the vast frequency of each frame and to convert time-series signal into frequency range is shown in figure 6.4.

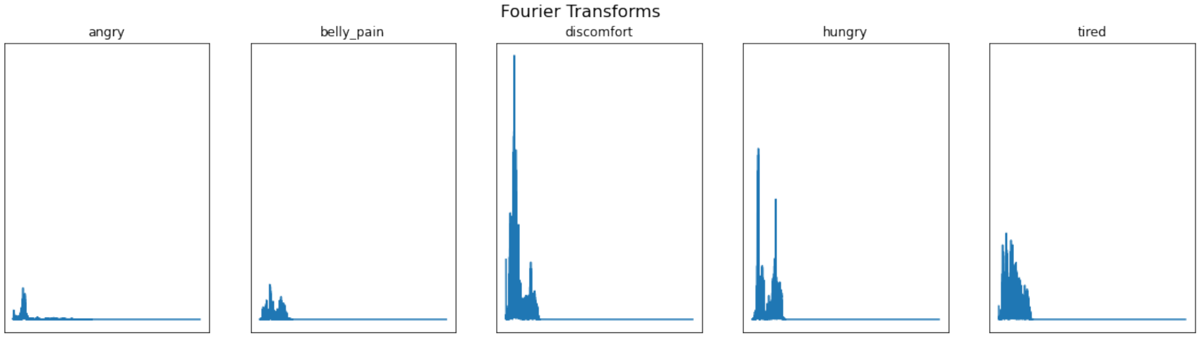


Figure 6.3: Fast Fourier Transform for Cry Categories

6.0.4 Convolutional Neural Network(CNN)

The coefficients of MFCC are then normalized and sent to Convolutional Neural Network (CNN) classifier for further categorization. From figure 6.5, The model consists of four deep two-dimensional convolutional layers (CONV2D). 16-128 filters shows boundary which is set in each layer and 3x3 size of kernel. The padding in the convolutional layer is unchanged. After the fourth convolutional layer, the max pooling procedure is used to help reduce computing demand by lowering dimensionality; however,now, keep up principal details. Dropout technique which helps in decreasing overfitting which is set to be 0.5, thus 50% of the nodes are dropped out randomly from the neural network. Other two variables for making system are reformer and loss function which are respectively Adam and categorical-cross entropy. After the model is constructed, training dataset of 13x9x1 input shape , 32 batch size and 10 epochs in figure . After ten rounds, the network obtains a test accuracy of 74%.

Layer (type)	Output Shape	Param #
conv2d_12 (Conv2D)	(None, 13, 9, 16)	160
conv2d_13 (Conv2D)	(None, 13, 9, 32)	4640
conv2d_14 (Conv2D)	(None, 13, 9, 64)	18496
conv2d_15 (Conv2D)	(None, 13, 9, 128)	73856
max_pooling2d_3 (MaxPooling2	(None, 6, 4, 128)	0
dropout_3 (Dropout)	(None, 6, 4, 128)	0
flatten_3 (Flatten)	(None, 3072)	0
dense_9 (Dense)	(None, 128)	393344
dense_10 (Dense)	(None, 64)	8256
dense_11 (Dense)	(None, 5)	325

Total params: 499,077
Trainable params: 499,077
Non-trainable params: 0

Figure 6.4: CNN Architecture

Chapter 7

Conclusions and Future Scope

Exploratory outcome conveyed that the presented method contains high categorization correctness. According to the MFCC estimated result, the CNN with a frame duration of 10 ms, after having 10 iterations, the system gets 72% test correctness. Our primary outcome shown that cry sound based on CNN has higher possibility of getting required results. In the future scope, A stronger and a massive datatset of infant cries should be made available. The goal is for people to be able to grasp the meaning of infants' cries more easily and quickly by using the model. For future studies, a larger amount of audio data will be used for training and testing.

Bibliography

[1] K. Teeravajanadet1 , N. Siwilai2 , K. Thanaselanggul3 , N. Ponsiricharoenphan4 , S. Tungjitkusolmun5 , P. Phasukkit6," An Infant Cry Recognition based on Convolutional Neural Network Method", The 2019 Biomedical Engineering International Conference (BMEiCON-2019)

[2] Chunyan Ji, Thosini Bamunu Mudiyanselage, Yutong Gao Yi Pan, "A review of infant cry analysis and classification", EURASIP Journal on Audio, Speech, and Music Processing volume 2021, Article number: 8 (2021), Georgia State University, 25 Park Place, Atlanta, USA.

[3] Chuan-Yu Chang, Jia-Jing Li," Application of Deep Learning for Recognizing Infant Cries", 2016 International Conference on Consumer Electronics-Taiwan

[4] Karinki Manikanta, K.P. Soman, M. Sabarimalai Manikandan," Deep Learning Based Effective Baby Crying Recognition Method under Indoor Background Sound Environments", IEEE 2019

Appendices

Appendix-I: FORMULAS USED

7.0.1 Root-Mean-Square Energy

A weeping signal consists of weeping expression and terminated voice as well as non-voice, quietness or other antiquity. To determine the threshold, a root-mean-square-energy computation is utilised. The following is the RMSE equation:

$$E[p] = \sqrt{\frac{\sum_{s}^{Q} |z[p-s]|^2}{Q}}$$

where E[p] is the rmse function, z[p] is the signal equation, Q is the frame length, s means index of window, s starts with zero and p is the sample index.

7.0.2 Pre-emphasis

To amplify the high frequencies, the first step is to apply a pre-emphasis filter to the signal. A type of filter that is necessary to minimise noise and provide a spectrum that is not wrinkled. The input is a time domain signal that is expressed as follows:

$$T[p] = z[p] - cz[p-1]$$

Constant of emphasis filter is denoted by c , c should be greater than 0.9 and less than 1.

7.0.3 Frame blocking and Windowing

The total speech is fragmented into frames. Because speech is a time-varying signal, framing is essential because the values are not dynamic when viewed over a short period of time. To remove alias signal, this split frame will be multiplied using the FIR filter. Spectral leakage is reduced by the windowing technique. A proper Hamming window multiplies each frame.

$$T_1[p] = z_1[p]W[p]$$

Here 'p' should be greater than 0 and less than 1. The Hamming window is represented by W[p]:

$$W[p] = 0.54 - 0.46 \cos \left[\frac{2\pi p}{Q-1} \right]$$

Here 'p' should be greater than zero and less than Q-1

7.0.4 Fast Fourier Transform

Multiply the retaliation with collection of band pass filters which are triangular so that a smooth spectrum can be achieved. To convert time-series signal into frequency spectrum:

$$D_p = \sum_{l=0}^{Q-1} z_l e^{\frac{-2\pi elp}{Q}}$$

7.0.5 Mel-Frequency Wrapping

Using triangular overspreading windows and mel scale, map the spectrum's strengths onto the mel scale. Mel-scaled filter banks — triangle filters that imitate human perception:

$$F_{mel} = 2595 * log_{10} \left[1 + \frac{F_{HZ}}{700} \right]$$

7.0.6 Log spectrum computation

Log of the powers should be taken at each mel frequency. Filter banks are created by applying triangular filters to the power spectrum on a Mel-scale to obtain frequency bands.

$$D_v = log_{10} \left[\sum_{l=0}^{Q-1} |z[l]| U_v[l] \right]$$

Here 'v' can be from 1 to M [M refers triangle filters number] and Uv[l] shows the value of v-triangle filter which is having l as acoustic frequency value.

7.0.7 Cepstrum

To obtain the coefficients, which will then be normalised and given to CNN to be categorise. Compute the discrete cosine transform of the list of mel log powers as if it were a signal.

Convert Mel-spectrum into the time domain using a Discrete Cosine Transform (DCT):

$$C_e = \sum_{e=0}^{L} D_e \cos\left[\frac{e(i-e)}{2\frac{\pi}{l}}\right]$$

Here 'Ce' is having the value of coefficient of MFCC, Power Spectrum of Mel frequency is represented by De and e lies between 1 and l [l represents coefficient which is desired].

Publication

Paper entitled **"New-born's Cry Analysis using Machine Learning"** is presented at **"International Conference on Computational Intelligence and Innovative Technologies (ICCIIT – 2022) - Elsevier SSRN series"** by **"Parvi Agrawal, Vaihsnavi Sriramoju, Manish Kumar, Prof. Kiran Deshpande and Prof. Nahid Shaikh"**.