AN EMOTION BASED MUSIC RECOMMENDER SYSTEM USING DEEP LEARNING

MSc Research Project
DATA ANALYTICS

Sodiq Adebiyi
Student ID: 19133979

School of Computing
National College of Ireland

Supervisor: Hicham Rifai

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: SODIQ ADEBIYI

Student ID: 19133979

Programme: DATA ANALYTICS **Year:** 2019/2020

Module: RESEARCH PROJECT

Supervisor: HICHAM RIFAI

Submission Due Date: 17th AUGUST 2020……………………………….…………………………………….………

Project Title: AN EMOTION BASED MUSIC RECOMMENDER SYSTEM USING DEEP LEARNING …………………………………………….......................………...................................……...

Word Count: 8384…..............................… Page Count…25…………………………………….……..

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……

Date: ……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

An Emotion Based Music Recommender System Using Deep Learning

Sodiq Adebiyi
19133979

Abstract

Emotions are affective states which show an individual's response to mental stimulus. In this paper, we propose the use of Deep Learning and music therapy to influence negative emotions associated with dementia episodes – (Anger, Fear, Sadness and Confusion) and therefore help the patient achieve neutral or positive emotion states – (Joy and Trust). Targeted emotions were identified with the help of a Convoluted Neural Network (CNN) model built on curated datasets containing annotated emotion-labelled audio clips. These audio clips were converted to Mel Frequency Spectrograms (MFS) for classification according to the emotions they represent. We then selected music according to this emotional state using content-based filtering. We achieved average classification results of accuracy: 95.25%, recall: 72.44%, precision: 87.42% and F1 score: 0.7922.

Keywords:
Deep Learning, Recommender Systems, Music Therapy, Dementia, Convoluted Neural Network

1 Introduction

Deep learning and music recommender systems (MRS) are two fields of research which have had a big impact in the field of Human Computer Interaction (HCI). Artificial Intelligence (AI) systems equipped with capabilities to improve the life of people are becoming more common place. This is also the case in the field of therapy, with researchers coming up with ways to improve the lives of patients.

In the case of Dementia, there is increasing interest in the subject matter of adapting deep learning to improve the lives of patients. As there is no approved cure for dementia, research is geared towards improving the way of life and reducing the danger dementia patients pose to themselves and others. Dementia episodes have been shown to be accompanied by various reactions such as sudden mood changes, confusion, among others, causing delirium.

One method which has been used to help dementia patients enjoy life better is music therapy(Khan et al., 2020). However, music therapy is often administered in the presence of a professional and in a preconceived environment. The implementation of music therapy can however be automated with AI, leading to the conception of this emotion-based music recommender system. Current research into this subject matter requires the use of wearable devices(Domínguez-Jiménez et al., 2020). We aim to achieve similar results with the use of the non-intrusive method of monitoring audio to recognize changes in emotion. The changes in emotion give an insight into the patient's state of mind and hence helps choose the kind of music to play to calm the patient.

The research objective for this work is to build an emotion-based music recommender system (MRS). In doing so, we answer the research question "Is it possible to build classify emotions from audio in the presence of noise distortions?". Answering this question identifies an alternative to implementing noise reduction for noisy input signals.

Limitations encountered during the implementation of this project include the lack of a predefined scoring metric for the audio files in the music recommender system; the overfitting of the model due to prevalence of silent frames in some of the audio files and the choices when building the model such as the optimizer to use, the minimum delta for the callback function, the use of the dropout layer and other parameters.

2 Related Work

2.1 EMOTION RECOGNITION

Automated emotion recognition and classification is an important topic in human-computer interaction. In literature discussing speech emotion recognition (SER), acoustic features are considered the most important features useful in identifying speech patterns for various emotions. Researchers have used various methods in emotion recognition, often relying on the learning the distribution of low-level parameters in audio files using generative models such as the Generative Mixture Model (GMM) and the Hidden Markov Model (HMM) which are then classified using classification algorithms such as Support Vector Machine (SVM) (Muljono et al., 2019), Bayesian Classifiers and Decision Trees; or more complicated methods such as the Boltzmann Machines (Lee et al., 2011). (Goudbeek and Scherer, 2010), discuss the roles played by the acoustic vocal parameters[1] – frequency, intensity, perturbation and range in contribution to emotional dimensions – valence, arousal and potency in an emotion-specific audio profile for twelve basic emotions, as well as the correlation between groups of vocal parameters.

(Lu and Jia, 2012) used an SVM to identify the emotions present in audio files. The authors, Muljono et. al. took advantage of the Mel's Frequency Cepstral Coefficients (MFCC) to train an SVM classifier. The audio files were cropped to two second snippets emphasizing the importance of having audio files of manageable duration. They gave no reason for using two second audio files, but (Medhat et al., 2017) explain that the use of a longer audio file results in uncertainty as to the exact emotion represented by the MFCCs, causing fluctuations and uncertainty in the result of classification.

The current trend in state-of-the-art techniques is the implementation of deep learning algorithms using various techniques to classify emotions. In (de Pinto et al., 2020), a Deep Convolutional Neural Network (DCNN) was used to classify emotions, taking its input as MFCC of snippets of audio files. The proposed classifier was built to classify eight different emotion classes, achieving an average F1 score of 0.91, which performed better when compared to a benchmark model built using a decision tree classifier which made an average score of 0.78.

In (Atsavasirilert et al., 2019), a DCNN was employed to recognize emotion in speech. In this case, the use of a Mel Spectrogram was adopted as input feature. The model architecture was built on an adaptation of the AlexNet model for SER. Although the accuracy of their model did not outperform the original AlexNet model used by Zhang et. al. in (Zhang et al., 2018) which was used as their benchmark, they proved their model to be effective and lightweight, training it on 220 thousand parameters which was considerably less than the 60 million parameters used by the benchmark model in (Zhang et al., 2018).

In (Suganya and Charles, 2019), Sungaya et. al. created a DCNN model which takes long vector values which represent the raw waveforms of the audio file as input to the model. The

[1] https://web.nmsu.edu/~lleeper/pages/Voice/simpson/acousticmeasures.html

audio files were sampled at a Nyquist sampling rate of 16KHz [2]. The DCNN was used to classify 5 emotions, (Anger, Neutral, Happy, Excitement, Sadness) from the IEMOCAP dataset with multiple experiments carried. When used to classify only Anger, Happiness, Neutral and Sadness the accuracies of classification were 0.592, 0.143, 0.799 and 0.764, respectively. Due to the low accuracy of classifying happiness, it was replaced with excitement in the next experiment resulting in accuracies of 0.401, 0.458, 0.758 and 0.747, respectively. A final experiment was carried out, leaving out both happiness and excitement. The resulting accuracies for Anger, Neutral and Sadness were 0.626, 0.822 and 0.811, respectively. The model built by Sungaya et. al. was also tested using the EmoDB dataset, achieving an average accuracy of 0.856. However, the results showed high misclassification rate of the happiness emotion, only achieving an accuracy of 0.476 and misclassifying with the Anger emotion. This is not consistent with the results from the first dataset and shows a need to build a more robust model.

Other means of identifying emotions without the use of audio files as input have also been used, with the use of video and sensors to measure physiological parameters also being applied in emotion recognition.

In (Cao et al., 2019), the use of an Electronic Encephalogram (EEG) was adopted to measure brain record brain activity. Features showing the power level in the frontal region of the brain were extracted and used to train a CNN. The features of the frontal region in [insert footnote] had previously been identified as the main driver in measuring valence and arousal felt in the brain, thus a 3D image of the signal from this part of the brain was extracted, normalized and flattened into a 2D signal image which served as input into the model. It had an average classification accuracy of 84.3±4.0% for Valence classification and 81.2±3.0% for arousal classification.

Video features have also been used to train classifier models, with the GMM finding application in a multitude of experiments. In (Sangapillai and Hegde, 2019), the R-Net model built with a Multi-task Cascaded Convolutional Neural Network (MTCNN) was trained to find joint statistics and identify facial landmarks in static videos. The model had an average accuracy of 0.783, misclassifying 23 emotions out of the 106 used to train the model.

Comparing the models reviewed, the use of Deep learning networks to identify emotions in an audio file produces good results. The challenge would be to build a robust model which would achieve reasonable accuracy even with the presence of noise in the signal. It is hardly likely that a clean audio signal is required to be classified in a real-world scenario.

As seen in (Jingzhou et al., 2019), research into the segmentation of audio signals to better identify noise, silence and actual audio is of great importance as it is imperative for a building robust classifier. Here, Jingzhou et. al. used the double threshold method to identify noise, silence and actual speech after inserting noise from the NOISE92 dataset into the original audio files. The architecture of their model consisted of 4 convolution layers, two pooling layers, a RELU layer and a Softmax layer. After the thresholding to separate noise from the actual signal, their adaptive-CNN model achieved accuracy of 0.894, 0.910, 0.948, 0.981,

[2] https://www.sciencedirect.com/topics/engineering/nyquist-theorem

1.000 and 0.995 in classifying male speech, female speech, speech with music, speech with noise, music, and noisy signals, respectively.

Further adaptations and research in working with noisy and degraded signals has also been implemented, with Chowdhury et. al. building in (Chowdhury and Ross, 2020) a model, the proposed 1D-Triplet-CNN which combines two short term spectral speech features – MFCC and Linear Prediction Cepstral Coefficient (LPCC). Their model outperformed existing baseline models by 14% for all the datasets it was tested on.

2.2 RECOMMENDER SYSTEMS

Recommender systems have been a topic of research for over a decade, finding application in movie streaming, music streaming, automated playlist generation, book recommendation, among others. Traditional methods of building recommender systems are categorized into three – Collaborative Filtering (CF), Content-based Filtering (CBF) and Hybrid Filtering (HF) (Park et al., 2012). CF is further divided into model-based implementation and memory-based implementations (Su and Khoshgoftaar, 2009) while the CBF uses the content of the item to compute similarity scores with methods such as TF-IDF (Term Frequency Inverse Document Frequency) (Achakulvisut et al., 2016) and Matrix Factorization (Mehta and Rana, 2017). Hybrid Filtering combines both the CF and CBF methods to improve the recommendations.

CF is typically used for large scale data found in applications like music and movie recommender systems. However, there is the need for the algorithm to compute similarities for each new item or user entry. There is also the cold start problem and sparsity of user information.

The quest to improve results from recommender engines in recent years is still ongoing with new techniques such as the use of knowledge graphs (KG). In (Wang et al., 2019), Hong Wei et al used multitask learning approach to implement KG enhanced recommendation called the Knowledge Graph Convolutional Network (KGCN). For music recommender systems (MRS), there is the potential use of emotions as contextual information to improve the output of music recommendation.

2.3 EMOTION-AWARE MUSIC RECOMENDER SYSTEMS

The use of music has been shown to evoke emotions, changing the mood of an individual according to the type of song being played, with researchers now taking advantage of this consequence of listening to music. (Deebika et al., 2019) proposed the use of emotions contextual information to facilitate choice of music in a music recommender system. The classifier was a CNN model trained on input parameters extracted from video showing the face of the user. The emotions identified when using the model are then used to curate music with the aim of improving the results of the MRS. The recommendation of songs is then implemented using the SVM algorithm on songs in a database. No mention was made in the paper of the methods of evaluation.

In (Shin et al., 2019), Shin et. al. made built a classifier ‘MusicRecNet’ based on the CNN algorithm to extract acoustic features based on auditory spike code (ASC) of an audio file. This method of feature extraction allows the algorithm to learn specific features from a spectral image of the audio file showing all the information, much like Mel’s Spectrogram.

This model recorded a mean accuracy of 85.0 ± 2.86% on the GTZAN dataset and 90.7% on the ISMIR2004 dataset.

The proposed techniques in (Shin et al., 2019) were modified and implemented in (Elbir and Aydin, 2020) and audio snippets were also used as input to a Deep Convolutional Neural Network (DCNN) in a system also called "MusicRecNet". Acoustic features of the audio signals were converted to images by generating Mel Spectrograms and then used to train a DCNN, achieving an accuracy of 81.8% when used alone and 97.6% when used in ensemble with SVM.

The use of images to train machine learning (ML) algorithms has been shown over the years to allows researchers to classify audio signals with minimal pre-processing of the data. This ensures feature extraction techniques such as Principal Component Analysis (PCA) and Dimensionality Reduction (DR) do not remove useful information from the data.

Due to the black box nature of neural networks, they are difficult to model mathematically. However, they are useful for this project due to the distinct nature of audio signals. When transformed to the frequency domain, spectrograms are used to make a pictorial representation of an audio signal which can then be used to train an neural classifier. As such, this method is heavily employed in the classification of audio signals.

The quest to build and improve emotion-based recommender systems has not been limited to just audio input. As seen in (Suganya and Charles, 2019) and (Cao et al., 2019), the use of wearable gadgets is believed to be a more reliable method of generating features it would require a user to keep an item on to consistently keep track of their emotions. The use of audio-visual methods to generate a feature set is more feasible as it would achieve the constant tracking which would be preferred for always-on music recommender systems.

The state-of-the-art in emotion detection and classification using Deep Convolutional Neural Networks (CNN) is made possible using Digital Signal Processing (DSP) techniques for Audio Spectral Analysis. The transformation of audio signals into Mel Frequency Spectrograms makes it possible to implement image classification algorithms when implementing audio classification.

3 Research Methodology

In this project, the aim is to discover how appropriate spectrogram-based input to CNNs are for identifying emotions in audio files. We also, investigate the effect of noise on the performance of the CNN model. To achieve the set objectives, we create a pipeline as shown in Figure 1 containing the operations described in the following subsections where we present the data used to achieve the objectives and the operations involved in our implementation.

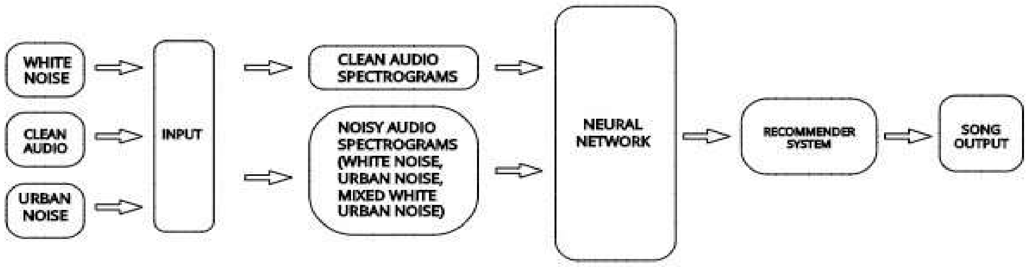

Figure 1: Implementation Pipeline of Emotion Classification and Music Recommendation

3.1 Data Description

This project is executed using the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS), (EMOTIFY) and URBANSOUND8K datasets. The datasets and their roles in the project are as described below.

3.1.1 RAVDESS

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)(Livingstone and Russo, 2018) contains 7356 files and takes up of 24.8 GB total space on disk. The database contains 24 professional actors (12 females, 12 males), pronouncing two linguistically similar utterances in a neutral North American intonation. Regular speech vocalization includes calm, happy, sad, angry, fearful, surprise, and disgust expressions, and song contains calm, happy, sad, angry, and fearful emotions. Each expression is produced at two levels of emotional intensity (normal, strong), with an additional neutral expression. All conditions are available in three modes - Audio-only (16bit, 48kHz .wav), Audio with Video (720p H.264, AAC 48kHz, .mp4), and Video-only (no sound). There are no song files for Actor_18.

Audio-only files

Audio-only files of all actors (01-24) are available as two separate zip files (~200 MB each):

- Speech file (Audio_Speech_Actors_01-24.zip, 215 MB) contains 1440 files: 60 trials per actor x 24 actors = 1440.
- Song file (Audio_Song_Actors_01-24.zip, 198 MB) contains 1012 files: 44 trials per actor x 23 actors = 1012.

Audio-Visual and Video-only files

Video files are provided as separate zip downloads for each actor (01-24, ~500 MB each), and are split into separate speech and song downloads:

- Speech files (Video_Speech_Actor_01.zip to Video_Speech_Actor_24.zip) collectively contains 2880 files: 60 trials per actor x 2 modalities (AV, VO) x 24 actors = 2880.
- Song files (Video_Song_Actor_01.zip to Video_Song_Actor_24.zip) collectively contains 2024 files: 44 trials per actor x 2 modalities (AV, VO) x 23 actors = 2024.

File naming format

Each of the 7356 RAVDESS files has a unique filename. The filename consists of a 7-part numerical identifier, delimited by a hyphen "-" after every two digits until the last digit. (e.g., 02-01-06-01-02-01-12.mp4). These identifiers define the stimulus characteristics:

Table 1: File Identifier for RAVDESS Dataset

IDENTIFIER	DEFINITION
Modality	01 = full-AV 02 = video-only 03 = audio-only
Vocal channel	01 = speech 02 = song
Emotion	01 = neutral 02 = calm 03 = happy 04 = sad 05 = angry 06 = fearful 07 = disgust 08 = surprised
Emotional intensity	01 = normal 02 = strong There is no strong intensity for the 'neutral' emotion.
Statement	01 = "Kids are talking by the door" 02 = "Dogs are sitting by the door"
Repetition	01 = 1st repetition 02 = 2nd repetition
Actor	01 to 24. Odd numbered actors are male, even numbered actors are female

3.1.2 EMOTIFY

The Emotify dataset (Aljanaki et al., 2016) contains 400 song excerpts (1 minute long) in 4 genres (rock, classical, pop, electronic). Each genre has 100 songs, annotated by multiple participants. The annotations were collected using GEMS scale (Geneva Emotional Music Scales). There are 9 possible emotions in this dataset, with each participant able to select a maximum of 3 emotions which they feel from each song. The annotations are spread unevenly among the songs by design. Each line in the file corresponds to one participant (i.e., annotations are not averaged per song). The information found in the file include:

- Id of the music file
- Genre of the music file
- 9 annotations by the participant (whether emotion was strongly felt for this song or not). 1 means emotion was felt, 0 means the emotion was not felt.
- Participant's mood prior to playing the game.
- Liking (1 if participant decided to report he liked the song).
- Disliking (1 if participant decided to report he disliked the song).
- Age, gender, and mother tongue of the participant (self-reported).

3.1.3 URBANSOUND8K

The UrbanSound8K dataset(Salamon et al., 2014) contains 8732 labeled sound excerpts (<=4s) of urban sounds from 10 classes: air_conditioner, car_horn, children_playing, dog_bark, drilling, enginge_idling, gun_shot, jackhammer, siren, and street_music.

File naming Format

The name of the audio file. The name takes the following format: [fsID]-[classID]-[occurrenceID]-[sliceID].wav, where:

Table 2: File Identifier for UrbanSound8K Dataset

IDENTIFIER	DEFINITION
fsID	the Freesound ID of the recording from which this excerpt (slice) is taken
classID	a numeric identifier of the sound class
occurrenceID	a numeric identifier to distinguish different occurrences of the sound within the original recording
sliceID	a numeric identifier to distinguish different slices taken from the same occurrence

The sampling rate, bit depth, and number of channels vary from file to file. The columns included in the dataset are:

Table 3: Available Metadata for UrbanSOund8K Dataset

IDENTIFIER	DEFINITION
slice_file_name	The name of the audio file
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. 1 = foreground, 2 = background
fold	The fold number (1-10) to which this file has been allocated.
classID	A numeric identifier of the sound class: 0 = air_conditioner 1 = car_horn 2 = children_playing 3 = dog_bark 4 = drilling 5 = engine_idling 6 = gun_shot 7 = jackhammer 8 = siren 9 = street_music
class	The class name - air_conditioner, car_horn, children_playing, dog_bark, drilling, engine_idling, gun_shot, jackhammer, siren, street_music.

3.2 Data Pre-processing

3.2.1 RAVDESS DATASET

The following pre-processing operations were carried out on the RAVDESS dataset.

Loading of Audio data into python

The dataset as described above contains both audio and video components. For our purposes, only the audio data is required and hence we discard the video files.
The main features that describe digital audio are:

Table 4: Relevant Features to describe Digital Audio

FEATURE	DEFINITION
Channels	number of audio channels (mono/stereo)
Sample Rate	samples per second expressed in kilohertz (kHz) which determines the quality of audio
Bit Depth	Describes the level of detail present in an audio file
Duration	length of the audio file over time

We load each audio file using the “librosa” library with a sample rate of 22.05KHz and a mono channel setting.

Data Augmentation

To augment the data on which we train the model, as well as increase the robustness of the neural network, we combine the clean audio files with 2 types of artificial noise - additive white gaussian noise (AWGN) and urban noise from the UrbanSound8K dataset.
We create the artificial white noise with a random signal to noise ratio (SNR) between 10 and 20 dB for every audio file. The ‘SNR’ determines the loudness of the noise being laid over the original audio file.
The urban noise is taken from the ‘URBANSOUND8K’ dataset. To create the urban noise augmented dataset, we randomly select types of noise from the UrbanSound8K dataset which allow for better distribution of the sounds. After combining the audio files, we save them to 4 locations, each location containing one of the following:

- Merged files of clean audio files
- Merged files of clean audio files with white noise
- Merged files of clean audio files with urban noise
- Merged files of clean audio files with white noise and urban noise

Audio Trimming

The audio files are trimmed with a noise identification parameter of signal to noise ratio[3] (SNR>45) to remove silence from the frame. This is done to prevent the model from attempting to match silence in training to identify emotions.

Merging of clean audio with noise

For our purposes in this project, the clean audio files have been merged with two kinds of noise, Additive White Gaussian Noise[4] (AWGN) which is generated as a random signal using NumPy and background noise gotten from the UrbanSound4K dataset.

3.2.2 Feature extraction and engineering

Conversion of audio to spectrograms

Using the “librosa” library, we converted the audio snippets into spectrograms, converting all the clean audio files, the audio files augmented with white noise and the audio augmented with urban sounds to spectrograms. The converted spectrogram lots were then saved using the “matplotlib” python library and storing them as appropriate in separate folders. The spectrogram is a visual representation of the audio spectrum as it varies with time. This makes it so a CNN can train on audio data as it would on an image.

Train – Test – Validation split

The train-test-validation split was done using the “split_folders” python library in a 60-20-20 ratio with training having 60%, testing having 20% and validation having 20% of the data.

The files contained in the RAVDESS audio dataset follow the naming condition as described above. The dataset file names are used to create a dataframe which serves as a lookup table containing file names, emotion, modality, along with other metadata.

3.3 MRS DATASETS

3.3.1 Data preprocessing

The EMOTIFY dataset of annotated music is used for this part of the project. The dataset contains information on the mood of the user and emotions which they feel while listening to a song. It also contains 400 music files of 100 files per genre, grouped according to genre for four genres – Rock, Electronic, Pop and Classical. Each user picks multiple emotions which the music evokes in them as in equation 1.

$$\boldsymbol{Score = Amazement\,Weight + Nostalgia\,Weight + Calmness\,Weight + Joyful\_Activation\,Weight + Tenderness\,Weight + Solemnity\,Weight - Tension\,Weight - Sadness\,Weight} - (\mathbf{1})$$

We generate a score by attaching weights to the emotions identified in the dataset. The columns representing the mother tongue and ‘liked’ were dropped as they either added no useful information or duplicated already present information. Positive emotions are attached to a positive weight value while negative emotions have a negative weight value. The emotions and their attached weights are:

[3] https://en.wikipedia.org/wiki/Signal-to-noise_ratio

[4] https://www.sciencedirect.com/topics/engineering/additive-white-gaussian-noise

Table 5: Emotion Classes, Classifications and Attached Weights

EMOTION	CLASSIFICATION	WEIGHT
Amazement	Positive	+1
Calmness	Positive	+1
Nostalgia	Positive	+1
Joyful Activation	Positive	+1
Solemnity	Positive	+1
Tenderness	Positive	+1
Tension	Negative	-1
Sadness	Negative	-1

This weighting method is arbitrary, solely for the purpose of execution of this project. However, the classification of the emotions into positive and negative is as classified by [21].

This is used to calculate an overall score, as an aggregate of the multiple emotions reportedly felt by while listening and then classified as an overall positive or an overall negative emotion invoking song. The 'track id' column in the dataset was renumbered to between 1 and 100 for each genre of music to be consistent with the naming convention in the audio folders. A correlation was performed on the dataset and no correlations were found between any of the columns. The results from the emotion classification are then used to decide the kind of emotion to be invoked and hence the song to play. Negative emotions identified by the CNN prompt the selection of highly positively scored songs (4 to 5) in the dataset. Positive emotions prompt the selection of songs with low positively scored songs (-2 to 2) and neutral emotions prompt neutral songs of score 3.

4 Design Specification

4.1 The CNN Model Architecture

The CNN model architecture for this project is built as described below.

1. Input: The input to the network are spectrograms of shape (256 x 256 x 3). The inputs are fed in mini batches of size 50.
2. Convolutional Layer: The model has three 2D-Convolutional layers both with filter sizes of 64 respectively and utilizing the "RELU' activation function. Each of the convolution layers have a kernel size of (4,4).
3. MaxPooling Layer: The network has three "MaxPooling" layers with pool sizes of (4,4) each following a convolution layer. We use the MaxPool2D layer for this project
4. Fully connected Layer: The model has two fully connected layers each with 32 nodes and using the relu activation function.
5. Dropout Layer: The network utilizes two dropout layers, each after the fully connected layers. Each of the dropout layers have dropout rates of 0.25.
6. Output Layer: The last layer of eight nodes corresponding to the eight emotions being identified. It uses the "softmax" activation function with loss computed with the "categorical_crossentropy" function and an "RMSprop" optimizer for compilation.

The model is trained to a maximum of 50 epochs, being stopped with an early stopping callback function set to monitor validation loss with a minimum acceptable change of 0.0005 and a patience of 2.
The input to the convolutional neural network is spectrograms of dimensions (256,256,3).

The output of a prediction using the CNN is the emotion represented in the spectrogram.

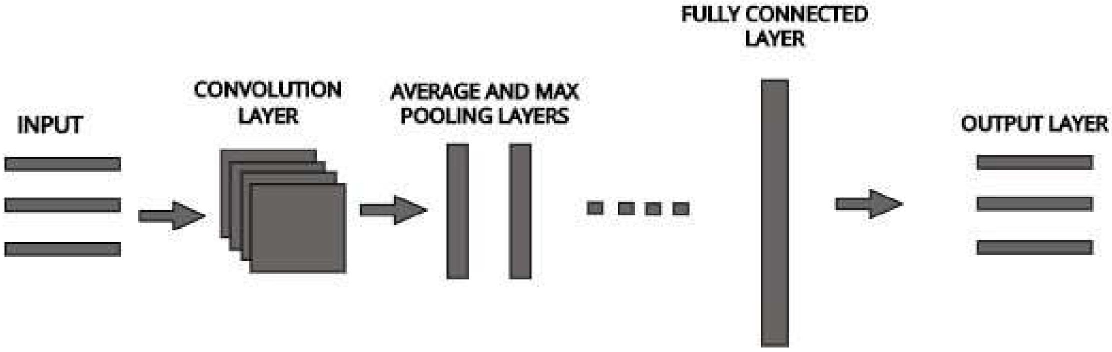

Figure 2: Process of Emotion recognition

The process is as detailed in figure 2 above, which shows the execution pipeline of the emotion recognition part of this project.

The CNN is trained separately on spectrograms from clean audio, clean audio with white noise overlaid, clean audio with randomly selected urban sounds overlaid on the audio and clean audio with both urban sounds and white noise overlaid. We then compare the results of the model for each group to understand the effects of noise on the robustness of the model.

As seen in figure (y), the output of the neural network is used as the input to a MRS in order to choose the song to be played.

5 Implementation

There are six (6) stages of implementation – Audio Loading, Audio trimming, Noise overlay, Spectrogram generation, Model training, Song Selection.

1. **Audio Loading:**
   The audio files were loaded using the Librosa library using the load method with a sampling rate of 22.05KHz, an offset of 0.0 and the mono channel option set to true.
2. **Audio Trimming:**
   The audio files were trimmed to remove frames of silence at a Signal-to-Noise ratio threshold of 45dB.
3. **Noise Overlay:**
   Each of the trimmed audio files was combined with two types of noise, randomly generated AWGN and urban noise files from the UrbanSound8k dataset. The resulting audio file combinations are the:
   - Clean audio
   - Clean audio overlaid with AWGN
   - Clean audio overlaid with urban noise
   - Clean audio overlaid with AWGN and urban noise

   To merge clean audio with noise files of unequal duration, the shorter audio file was padded with 0s to make the audio and noise files of equal length before audio overlay. The noise and audio files were merged with an SNR of a randomly selected number between 10 and 20. This was done to take the model as far away from the ideal scenario as possible and allow for the robustness of the model in the presence of disturbance to be tested. We take this approach with the rationale that the loudness of noise in the real world is random.

Noise overlay was carried out at a randomly selected SNR between 10dB and 20dB.

4. **Spectrogram Generation:**
   The merged audio files described above are converted to mel frequency spectrograms using the Librosa Library according to the audio file vocal channel.
   - Speech Vocal channel settings (n_mels=128, fmax=8000, n_fft = 512)
   - Song Vocal Channel settings (n_mels=128, fmax=8000, n_fft = 2048)
5. **Model Training**
   All the generated spectrograms are then split into train, test, and validation data in the ratio 60:20:20. The training and validation splits are used to train the convolutional neural network described in section 4.
6. **Song selection:**
   The emotion identified by the neural network is used select the song to be played according to the scoring metric as presented in section 4.

6 Evaluation

The results of the audio-based emotion recognition are evaluated on the accuracy, precision, recall, area under the curve (AUC) and F1 score metrics. Experiments were carried out on all the groups of spectrogram data collected for training to ascertain the scores for our evaluation metrics, as well as investigate the effects of noise on the performance of our model. The trends of loss, accuracy, precision, and accuracy were measured and graphed for the training, validation, and testing datasets in the subsection 6.1, 6.2, 6.3 and 6.4 below. In section 6.5, we discuss the results of these experiments and their implications.

6.1 EMOTION CLASSIFICATION OF CLEAN AUDIO FILES

In this subsection, we discuss the results of the classification of the clean audio files and obtained the following results.

We obtained a training accuracy of 93.09%, a training precision of 81.02%, a training recall of 58.43%, a training AUC of 93.90%, a training F1 score of 0.6759 with a training loss of 0.9096. Also, validation results give us a validation accuracy of 93.64%, a validation precision of 83.59%, a validation recall of 61.11%, a validation AUC of 95.10%, a validation F1 score of 0.7067 with a validation loss of 0.8134.

For our test results, we obtain a loss of 0.8536 with a test accuracy of 93.86%, a test precision of 83.73%, a recall of 63.17%, an AUC of 94.63% and an F1 score of 0.7163.

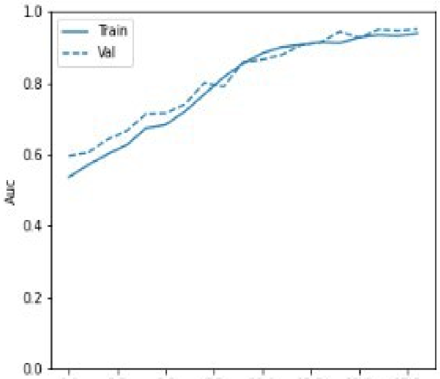

Figure 3(a): Trend of Classification results for AUC for clean audio

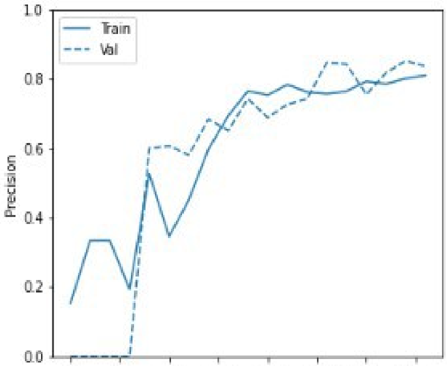

Figure 3(b): Trend of Classification results for Precision for clean audio

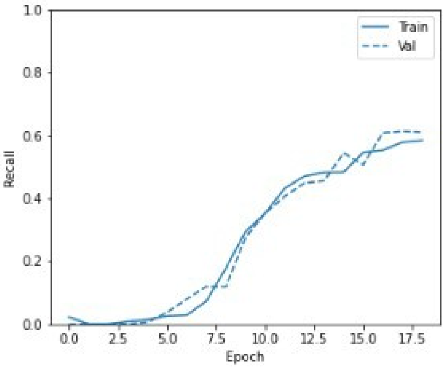

Figure 3(c): Trend of Classification results for Recall for clean audio spectrograms

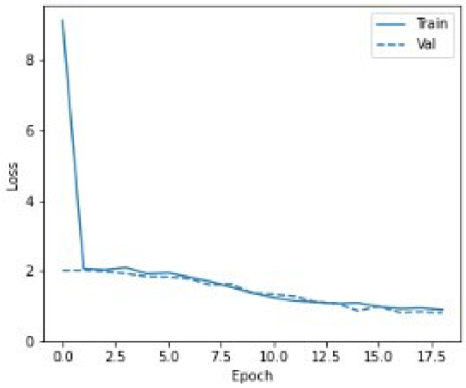

Figure 3(d): Trend of Classification results for Loss for clean audio spectrograms

Figures 3(a-d) show a minimal difference in the loss, recall, prediction, and the AUC. This suggests convergence of the model.

6.2 EMOTION CLASSIFICATION OF WHITE NOISE OVERLAID ON AUDIO FILES

In this subsection, we discuss the results of the classification of the clean audio files and obtained the following results.

We obtained a training accuracy of 93.97%, a training precision of 84.81%, a training recall of 63.02%, a training AUC of 95.02%, a training F1 score of 0.7229 with a training loss of 0.8186. Also, validation results give us a validation accuracy of 93.75%, a validation precision of 82.23%, a validation recall of 63.78%, a validation AUC of 94.91%, a validation F1 score of 0.7184 with a validation loss of 0.8361.

For our test results, we obtain a loss of 0.8457 with a test accuracy of 93.91%, a test precision of 82.24%, a recall of 65.19%, AUC of 95.03% and an F1 score of 0.7272.

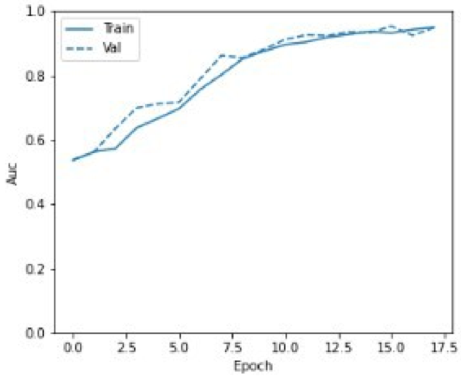

Figure 4(a): Trend of Classification results for AUC for clean audio files with white noise overlay

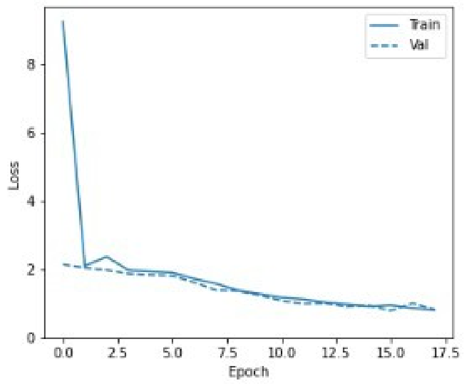

Figure 4(b): Trend of Classification results for Loss for clean audio files with white noise overlay

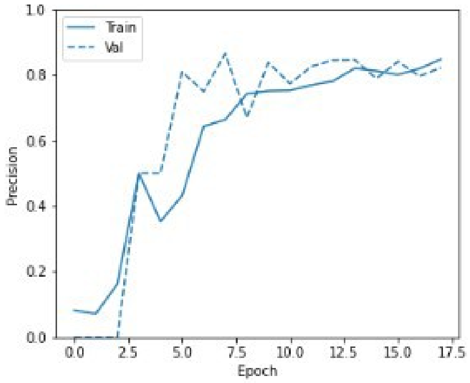

Figure 4(c): Trend of Classification results for Precision for clean audio files with white noise overlay

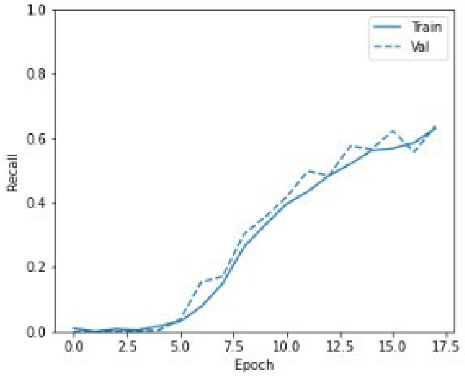

Figure 4(d): Trend of Classification results for Recall for clean audio files with white noise overlay

Figures 4(a-d) show a minimal difference in the loss, recall, prediction, and the AUC. This suggests convergence of the model.

6.3 EMOTION CLASSIFICATION OF URBAN NOISE OVERLAID ON AUDIO FILES

In this subsection, we discuss the results of the classification of the clean audio files and obtained the following results.

We obtained a training accuracy of 92.69%, a training precision of 80.37%, a training recall of 54.98%, a training AUC of 92.50%, a training F1 score of 0.6529 with a training loss of 1.0021. Also, validation results give us a validation accuracy of 93.11%, a validation precision of 83.67%, a validation recall of 55.78%, a validation AUC of 93.04%, a validation F1 score of 0.6693 with a validation loss of 0.9679.

For our test results, we obtain a loss of 0.9493 with a test accuracy of 92.72%, a test precision of 80.35%, a recall of 55.35%, AUC of 93.37% and an F1 score of 0.6554.

Below, we show the trend for our training and validation the effect white noise on the data and its impact on the results of the model.

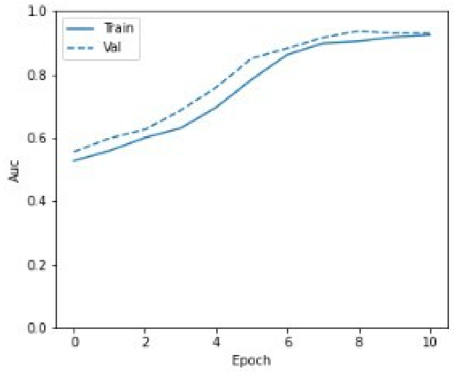

Figure 5a: Trend of Classification results for AUC for clean audio files with urban noise overlay

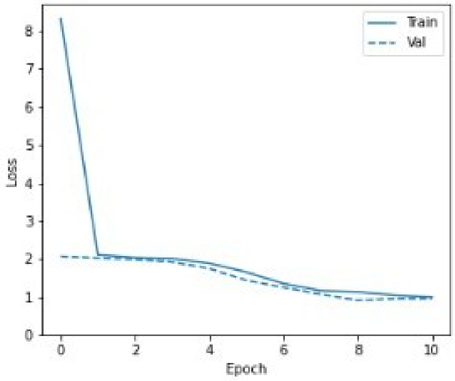

Figure 5b: Trend of Classification results for Loss for clean audio files with urban noise overlay

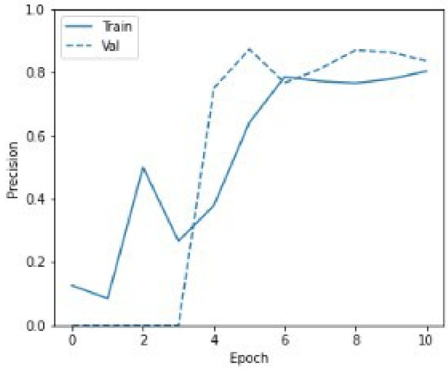

Figure 5c: Trend of Classification results for Precision for clean audio files with urban noise overlay

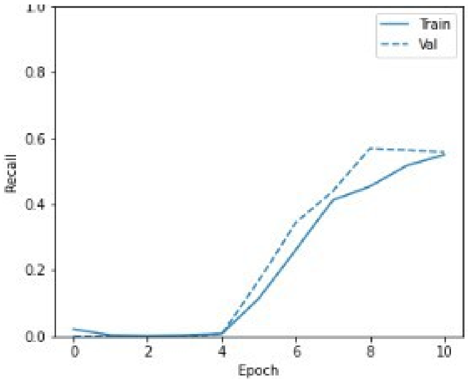

Figure 5d: Trend of Classification results for Recall for clean audio files with urban noise overlay

Figures 5(a-d) show a minimal difference in the loss, recall, prediction, and the AUC. This suggests convergence of the model.

6.4 EMOTION CLASSIFICATION OF WHITE AND URBAN NOISE OVERLAID ON AUDIO FILES

In this subsection, we discuss the results of the classification of the clean audio files and obtained the following results.
We obtained a training accuracy of 94.22%, a training precision of 84.88%, a training recall of 65.40%, a training AUC of 95.24%, a training F1 score of 0.7387 with a training loss of 0.7974. Also, validation results give us a validation accuracy of 94.25%, a validation precision of 84.42%, a validation recall of 66.22%, a validation AUC of 95.71%, a validation F1 score of 0.7422 with a validation loss of 0.7579.
For our test results, we obtain a loss of 0.7489 with a test accuracy of 94.70%, a test precision of 87.43%, a recall of 67.33%, AUC of 95.76% and an F1 score of 0.7607.
Below, we show the trend for our training and validation the effect white noise on the data and its impact on the results of the model.

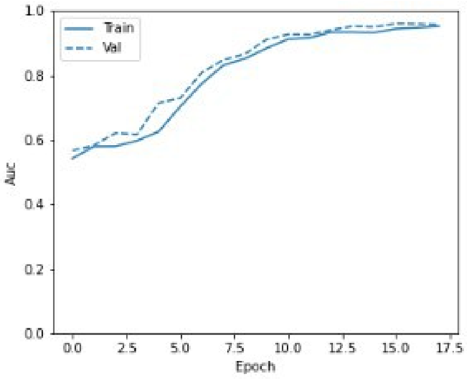

Figure 6(a): Trend of Classification results for AUC for clean audio files with white and urban noise overlay

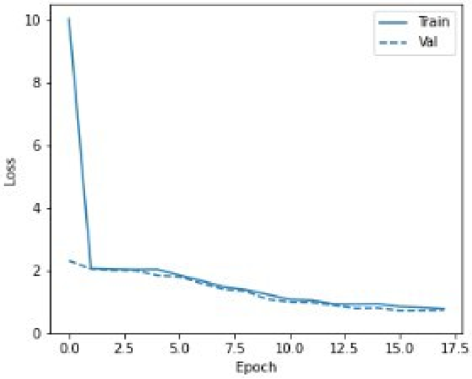

Figure 6(b): Trend of Classification results for Loss for clean audio files with white and urban noise overlay

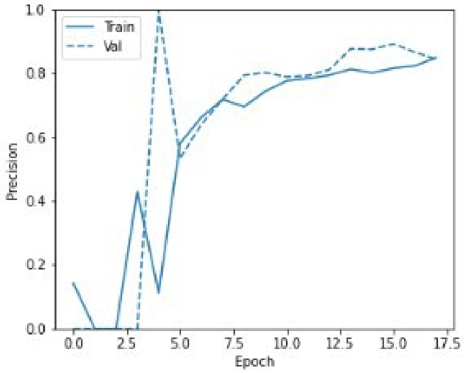

Figure 6(c): Trend of Classification results for Precision for clean audio files with white and urban noise

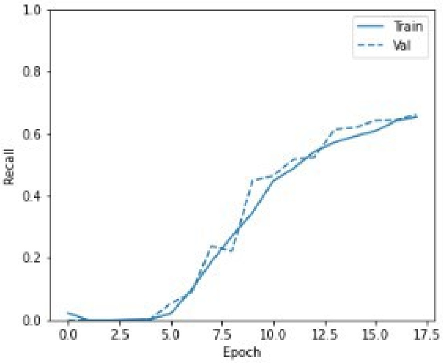

Figure 6(d): Trend of Classification results for Recall for clean audio files with white and urban noise overlay

Figures 6(a-d) show a minimal difference in the loss, recall, prediction, and the AUC. This suggests convergence of the model.

6.5 EMOTION CLASSIFICATION OF ALL SPECTROGRAMS WITH AND WITHOUT NOISE OVERLAID ON AUDIO FILES

In this subsection, we discuss the results of the classification of the clean audio files and obtained the following results.

We obtained a training accuracy of 95.60%, a training precision of 88.28%, a training recall of 74.70%, a training AUC of 97.04%, a training F1 score of 0.8092 with a training loss of 0.6187. Also, validation results give us a validation accuracy of 95.78%, a validation precision of 90.54%, a validation recall of 73.93%, a validation AUC of 97.42%, a validation F1 score of 0.8139 with a validation loss of 0.5856.

For our test results, we obtain a loss of 0.6556 with a test accuracy of 95.25%, a test precision of 87.42%, a recall of 72.44%, AUC of 96.73% and an F1 score of 0.7922.

Below, we show the trend for our training and validation the effect white noise on the data and its impact on the results of the model.

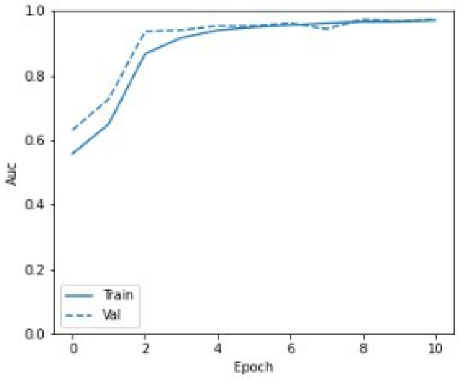

Figure 7(a): Trend of Classification results for AUC for all audio files

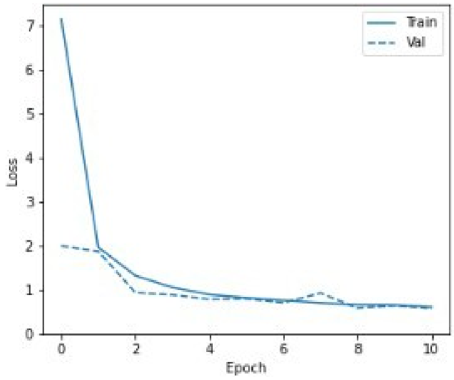

Figure 7(b): Trend of Classification results for Loss for all audio files

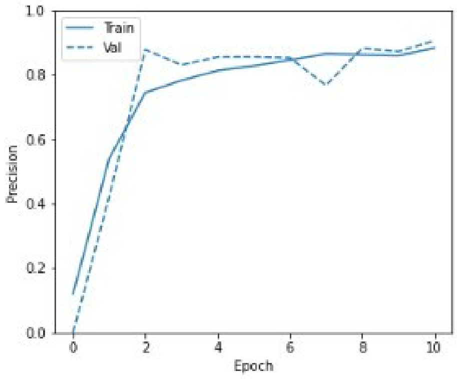

Figure 7(c): Trend of Classification results for Precision for all audio files

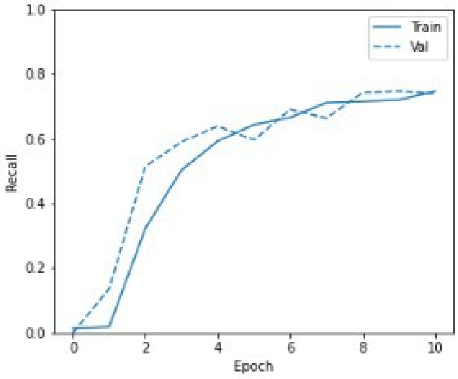

Figure 7(d): Trend of Classification results for Recall for all audio files

Figures 7(a-d) show a minimal difference in the loss, recall, prediction, and the AUC. This suggests convergence of the model.

The loss shown in each of the experiments shows a slight difference in the final values for the loss for all models suggesting we have a valid model.

Table 6: Table of Aggregated Performance Results

DATA GROUP	ACCURACY(%)	PRECISION(%)	RECALL(%)	F1 SCORE
Clean Audio	93.86	83.73	63.17	0.7163
Clean Audio with White Noise	93.91	82.24	65.19	0.7272
Clean Audio with Urban Noise	92.72	80.35	55.35	0.6554
Clean Audio with White and Urban Noise	94.76	87.43	67.33	0.7607
Clean Audio, Clean audio with White noise, clean audio with Urban Noise	95.25	87.42	72.44	0.7922

6.6 DISCUSSION

The results experiments performed as shown in subsections 6.2 – 6.5 show a consistency in the results obtained for the training, evaluation, and testing. The small margin in the difference between the results of the training and validation show the model is not overfitting. We obtain the best results of the experiments in subsection 6.5 as expected. This is due to the availability of more data for training the model. We obtained a test accuracy 95.25%, a test precision of 87.42%, a recall of 72.44%, AUC of 96.73% and an F1 score of 79.22%.

We observed the worst performance of the model in subsection 6.3, with the addition of urban noise causing a reduction in the precision and recall. We obtained a test accuracy of 92.72%, a test precision of 80.35%, a recall of 55.35%, AUC of 93.37% and an F1 score of 65.54%. We attribute this to the randomness introduced into the audio by introducing urban noise. When compared to the performance of introducing white noise which is more stable as it is constant over the entire audio, where we obtained performance of a test accuracy of 93.91%, a test precision of 82.24%, a recall of 65.19%, AUC of 95.03% and an F1 score of 72.72%. This shows that the type of noise introduced affects the ability of the model to generalize. The model however returns good results for all the experiments carried out, with F1 scores of greater than 65% for all the experiments.

Due to the size of the dataset, there is no justification for adding more layers and building a more complex model. However, reducing the minimum delta in the stopping function of the model could further improve the results obtained from the experiments. Further diversification of the data to include audio from other nationalities could aid in building a more generalized model. Also, the inclusion of stuttered speech, and varying intensity levels of speech into the data as is common with dementia patients will further improve the model. Our model is not easily compared to other models due to the introduction of artificial noise directly into the audio files in the dataset. However, comparing the results from the clean audio files to results obtained from similar studies in the table below,

Table 7: Table of comparisons of results of evaluation metrics for similar projects

REFERENCE	CLASSIFIER	EVALUATION METRIC	SCORE OF REFERENCE	SCORE OF PROPOSED MODEL
(Atsavasirilert et al., 2019)	CNN	Recall	85.54%	63.17%
(Cao et al., 2019)	CNN	Accuracy	84.3±4%	93.86%

Our model was found to outperform other models in accuracy, but its recall can be improved.

7 Conclusion and Future Work

In conclusion, we have investigated the usability of a CNN in classifying audio signals by emotion in a noisy environment. We used a CNN model to classify eight emotions with the introduction of different types of noise (white noise and urban noise). We investigated the effect of noise on the performance of a CNN model and concluded that urban noise such as other speech, dog barking and other random sounds pose a greater hinderance to the performance of the model. The noise was overlaid on clean audio files containing speech of individuals with an SNR between 10dB and 20dB.

Our findings are consistent with expectations. However, there is room to improve the precision and recall of the model. Due to the purpose of this project, there is not much room for error, hence we require a better model to achieve our objectives.

Also, we only performed our experiments at an SNR of between 10dB and 20dB. Reducing the SNR further and hence increasing the presence of noise will have an adverse effect on the model. As this would be the case in a real-world scenario with different noise levels, it would be of interest to investigate the performance of the model at different noise levels.

The data used to build this model was only gathered from English speaking North American countries, hence making the model very localized. For a truly inclusive model, investigating the performance of the model with data from other regions of the world, taking into consideration differences in dialect, intonation, and other peculiarities could be seen to influence the importance of the model.

To further improve the model, the incorporation of Mel Frequency Cepstral Coefficients with the noise levels taken into consideration in a form of ensemble using a classifier such as a support vector machine (SVM) should be investigated. This is as opposed to performing noise reduction or noise removal using audio gating techniques, hence preparing a model to classify emotion and recommend music despite the noise.

Also, we used an arbitrary formula to generate a scoring metric. More research is required to properly implement a scoring metric for music recommendation.

References

Achakulvisut, T., Acuna, D.E., Ruangrong, T., Kording, K., 2016. Science Concierge: A Fast Content-Based Recommendation System for Scientific Publications. PLOS ONE 11, e0158423. https://doi.org/10.1371/journal.pone.0158423

Aljanaki, A., Wiering, F., Veltkamp, R.C., 2016. Studying emotion induced by music through a crowdsourcing game. Inf. Process. Manag., Emotion and Sentiment in Social and Expressive Media 52, 115–128. https://doi.org/10.1016/j.ipm.2015.03.004

Atsavasirilert, K., Theeramunkong, T., Usanavasin, S., Rugchatjaroen, A., Boonkla, S., Karnjana, J., Keerativittayanun, S., Okumura, M., 2019. A Light-Weight Deep Convolutional Neural Network for Speech Emotion Recognition using Mel-Spectrograms, in: 2019 14th International Joint Symposium on Artificial Intelligence and Natural Language Processing (ISAI-NLP). Presented at the 2019 14th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP), pp. 1–4. https://doi.org/10.1109/iSAI-NLP48611.2019.9045511

Cao, G., Ma, Y., Meng, X., Gao, Y., Meng, M., 2019. Emotion Recognition Based On CNN, in: 2019 Chinese Control Conference (CCC). Presented at the 2019 Chinese Control Conference (CCC), pp. 8627–8630. https://doi.org/10.23919/ChiCC.2019.8866540

Chowdhury, A., Ross, A., 2020. Fusing MFCC and LPC Features Using 1D Triplet CNN for Speaker Recognition in Severely Degraded Audio Signals. IEEE Trans. Inf. Forensics Secur. 15, 1616–1629. https://doi.org/10.1109/TIFS.2019.2941773

de Pinto, M.G., Polignano, M., Lops, P., Semeraro, G., 2020. Emotions Understanding Model from Spoken Language using Deep Neural Networks and Mel-Frequency Cepstral Coefficients, in: 2020 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS). Presented at the 2020 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS), pp. 1–5. https://doi.org/10.1109/EAIS48028.2020.9122698

Deebika, S., Indira, K.A., Jesline, 2019. A Machine Learning Based Music Player by Detecting Emotions, in: 2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM). Presented at the 2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), pp. 196–200. https://doi.org/10.1109/ICONSTEM.2019.8918890

Domínguez-Jiménez, J.A., Campo-Landines, K.C., Martínez-Santos, J.C., Delahoz, E.J., Contreras-Ortiz, S.H., 2020. A machine learning model for emotion recognition from physiological signals. Biomed. Signal Process. Control 55, 101646. https://doi.org/10.1016/j.bspc.2019.101646

Elbir, A., Aydin, N., 2020. Music genre classification and music recommendation by using deep learning. Electron. Lett. 56, 627–629. https://doi.org/10.1049/el.2019.4202

Goudbeek, M., Scherer, K., 2010. Beyond arousal: valence and potency/control cues in the vocal expression of emotion. J. Acoust. Soc. Am. 128, 1322–1336. https://doi.org/10.1121/1.3466853

Jingzhou, S., Yongbin, W., Xiaosen, C., 2019. Audio Segmentation and Classification Approach Based on Adaptive CNN in Broadcast Domain, in: 2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS). Presented at the 2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS), pp. 1–6. https://doi.org/10.1109/ICIS46139.2019.8940257

Khan, S.H., Xu, C., Purpura, R., Durrani, S., Lindroth, H., Wang, S., Gao, S., Heiderscheit, A., Chlan, L., Boustani, M., Khan, B.A., 2020. Decreasing Delirium Through Music: A Randomized Pilot Trial. Am. J. Crit. Care 29, e31–e38. https://doi.org/10.4037/ajcc2020175

Lee, C.-C., Mower, E., Busso, C., Lee, S., Narayanan, S., 2011. Emotion recognition using a hierarchical binary decision tree approach. Speech Commun. 53, 1162–1171. https://doi.org/10.1016/j.specom.2011.06.004

Livingstone, S.R., Russo, F.A., 2018. The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. PLOS ONE 13, e0196391. https://doi.org/10.1371/journal.pone.0196391

Lu, K., Jia, Y., 2012. Audio-visual emotion recognition using Boltzmann Zippers, in: 2012 19th IEEE International Conference on Image Processing. Presented at the 2012 19th IEEE International Conference on Image Processing (ICIP 2012), IEEE, Orlando, FL, USA, pp. 2589–2592. https://doi.org/10.1109/ICIP.2012.6467428

Medhat, F., Chesmore, D., Robinson, J., 2017. Masked Conditional Neural Networks for Environmental Sound Classification. ArXiv180510004 Cs Eess Stat 10630, 21–33. https://doi.org/10.1007/978-3-319-71078-5_2

Mehta, R., Rana, K., 2017. A review on matrix factorization techniques in recommender systems, in: 2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA). Presented at the 2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA), IEEE, Mumbai, India, pp. 269–274. https://doi.org/10.1109/CSCITA.2017.8066567

Muljono, Prasetya, M.R., Harjoko, A., Supriyanto, C., 2019. Speech Emotion Recognition of Indonesian Movie Audio Tracks based on MFCC and SVM, in: 2019 International Conference on Contemporary Computing and Informatics (IC3I). Presented at the 2019 International Conference on contemporary Computing and Informatics (IC3I), pp. 22–25. https://doi.org/10.1109/IC3I46837.2019.9055509

Park, D.H., Kim, H.K., Choi, I.Y., Kim, J.K., 2012. A literature review and classification of recommender systems research. Expert Syst. Appl. 39, 10059–10072. https://doi.org/10.1016/j.eswa.2012.02.038

Salamon, J., Jacoby, C., Juan Pablo Bello, 2014. A Dataset and Taxonomy for Urban Sound Research. http://dx.doi.org/10.1145/2647868.2655045

Sangapillai, T., Hegde, S.M., 2019. Emotion Extrication and Analysis on Videos, in: 2019 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM). Presented at the 2019 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), pp. 92–96. https://doi.org/10.1109/CCEM48484.2019.00019

Shin, S.-H., Yun, H.-W., Jang, W.-J., Park, H., 2019. Extraction of acoustic features based on auditory spike code and its application to music genre classification. IET Signal Process. 13, 230–234. https://doi.org/10.1049/iet-spr.2018.5158

Su, X., Khoshgoftaar, T.M., 2009. A Survey of Collaborative Filtering Techniques [WWW Document]. Adv. Artif. Intell. https://doi.org/10.1155/2009/421425

Suganya, S., Charles, E.Y.A., 2019. Speech Emotion Recognition Using Deep Learning on audio recordings, in: 2019 19th International Conference on Advances in ICT for Emerging Regions (ICTer). Presented at the 2019 19th International Conference on Advances in ICT for Emerging Regions (ICTer), pp. 1–6. https://doi.org/10.1109/ICTer48817.2019.9023737

Wang, H., Zhao, M., Xie, X., Li, W., Guo, M., 2019. Knowledge Graph Convolutional Networks for Recommender Systems. World Wide Web Conf. - WWW 19 3307–3313. https://doi.org/10.1145/3308558.3313417

Zhang, Shiqing, Zhang, Shiliang, Huang, T., Gao, W., 2018. Speech Emotion Recognition Using Deep Convolutional Neural Network and Discriminant Temporal Pyramid Matching. IEEE Trans. Multimed. 20, 1576–1590. https://doi.org/10.1109/TMM.2017.2766843

Table of Figures