BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Emotion Tagging In An Audio Signal Using Weakly Supervised learning

Naga Srimouli Borusu

2019AH04075
2019ah04075@wilp.bits-Pilani.ac.in

Plan of Events :

S. No	Task	Expected date of completion	Names of Deliverables	Progress
1	Outline	Oct 25, 2021	Dissertation Outline	Completed
2	Design and Procuring Datasets	Nov 15, 2021	• Project plan • Datasets	Completed
3	Basic Implementation	Dec 7, 2021	• Preparing the CNN models • Preparing the Auto encoder and CNN models	Completed
4	Testing and Fine Tuning the models	Dec 31, 2021	Benchmarking the model	Completed
5	Rework on the Suggestions	Jan 13, 2021	Benchmarking the model on reworked suggestions	Completed
6	Documentation	Jan 18, 2021	Final Project Report	Completed

Evalution from Organization Supervisor :

S. No	Task	Weightage	**Comments**	**Marks Awarded**
1	Dissertation Outline	10%	The abstract looks interesting and is conveying your intentions clearly.	10%
2	Mid-Sem • Progress • Seminar Viva • Work Progress	10% 5% 15%	The datasets gathered and the work done so far tagging the emotions looks impressive.	Seminar - 10% Viva - 5% Work Progress – 15%
3	Final Seminar/Viva	20%	Your work looks impressive.	20%
4	Final Report	40%		40%

Abstract

- Emotions are vital in human-to-human communication and connection because they allow individuals to express themselves in ways beyond the scope of language.

- The computer’s ability to read human emotions is critical in a variety of applications. Compared to other machine/deep learning research domains, such as computer vision, audio analysis has received less attention.

- The majority of existing research in this area is focused on supervised learning algorithms, with limited emphasis on weakly-supervised approaches.

- Obtaining the tagged data sets that subject matter experts have annotated is an expensive and time-consuming process.

- To address this issue, weakly supervised approaches can use labels generated by non-experts to aid the model train with minimal input of annotated tags from subject matter experts.

- The project compares the efficiencies of different deep learning techniques to predict the emotion labels of an input audio signal.

Significance of Emotion Tagging

- Humans communicate through expressive gestures of emotions and sentiments identified through experience and knowledge. These emotions might be expressed verbally or via body language.

- The varieties of traits that may contain additional details well about the emotional significance of each utterance are examined in this research. The characteristics that contribute to emotions may differ between spoken languages

- An effective human emotion detection algorithm will aid in making human-computer interaction more natural and pleasant.

- Recognizing emotions is a challenging problem because human emotions lack temporal constraints and are diverse

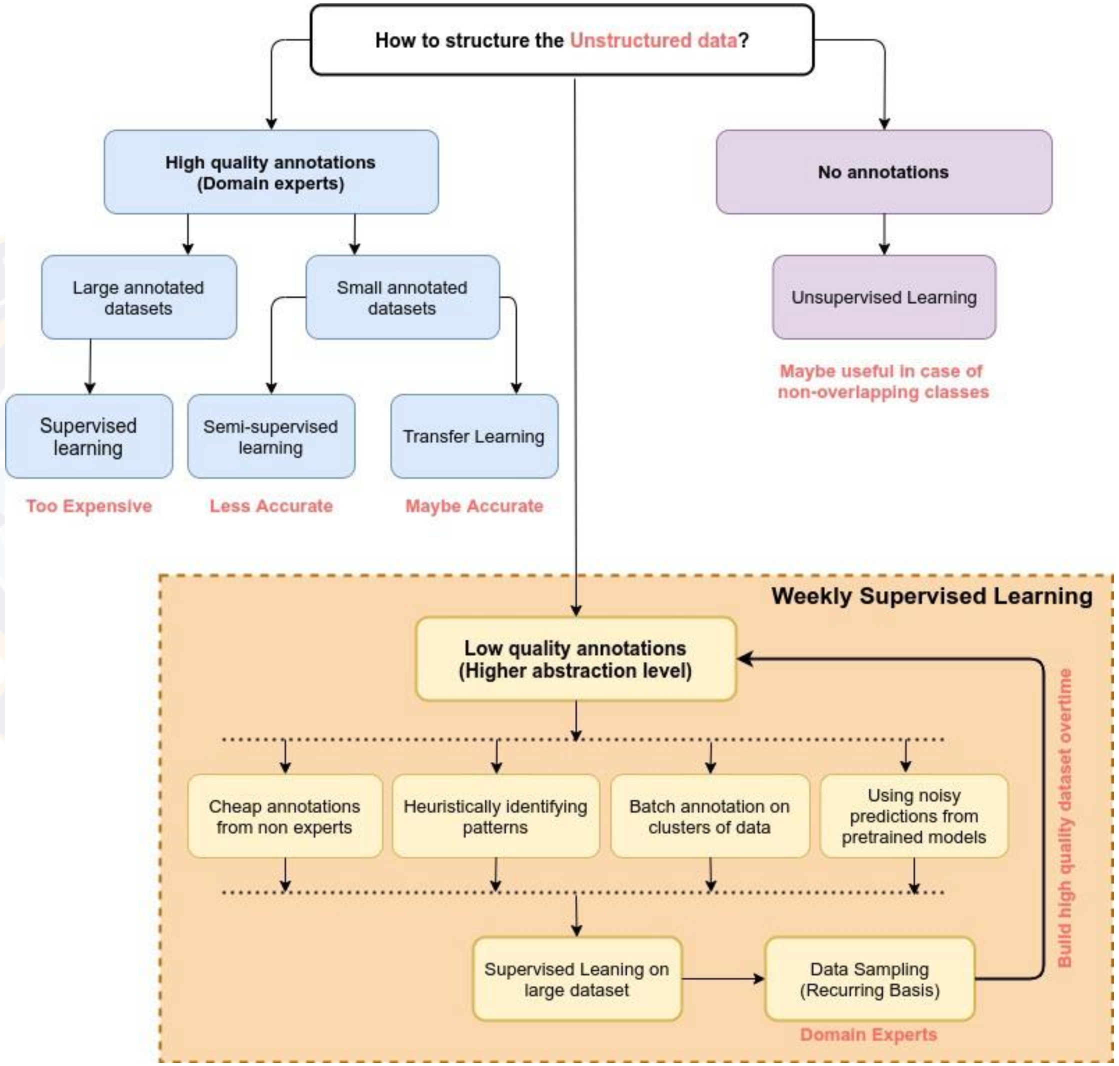
- The automatic recognition of spontaneous emotions from the speech is a challenging task. On the one hand, acoustic features need to be robust enough to capture the emotional content for various speaking styles. On the other, machine learning algorithms need to be insensitive to outliers while modelling the context

Weakly Supervised Learning

- “Weakly Supervised Learning” is a branch of machine learning in which noisy, restricted, or inaccurate sources are employed to give supervision signals for categorizing vast volumes of training data in a supervised learning scenario.

- This methodology ameliorates the burden of acquiring hand-labelled data sets, which can be expensive or onerous.

- Instead, low-cost weak labels are applied with the awareness that they are imprecise but can still be used to build a powerful prediction model.


How to structure the Unstructured data?
High quality annotations (Domain experts)
No annotations
Large annotated datasets
Small annotated datasets
Unsupervised Learning
Maybe useful in case of non-overlapping classes
Supervised learning
Semi-supervised learning
Transfer Learning
Too Expensive
Less Accurate
Maybe Accurate
Weekly Supervised Learning
Low quality annotations (Higher abstraction level)
Cheap annotations from non experts
Heuristically identifying patterns
Batch annotation on clusters of data
Using noisy predictions from pretrained models
Supervised Leaning on large dataset
Data Sampling (Recurring Basis)
Domain Experts
Build high quality dataset overtime

Datasets Used

Surrey Audio-Visual Expressed Emotion [SAVEE]

- This dataset is provided by researchers at the University of Surrey, Guildford, England. The speakers are aged from 27 to 31 years. Emotion has been described psychologically in discrete categories: anger, disgust, fear, happiness, sadness and surprise

Ryerson AV Database of Emotional Speech and Song [RAVDESS]

- The database contains 24 professional actors (12 female, 12 male), vocalizing two lexically matched statements in a neutral North American accent. Speech 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

Toronto emotional speech set [TESS]

- These stimuli were modeled on the Northwestern University Auditory Test No. 6 (NU-6; Tillman and Carhart, 1966). A set of 200 target words were spoken in the carrier phrase ”Say the word ” by two actresses (aged 26 and 64 years) and recordings were made of the set portraying each of seven emotions (anger, disgust, fear, happiness, pleasant surprise, sadness, and neutral). There are 2800 stimuli in total. Two actresses were recruited from the Toronto area. Both actresses speak English as their first language, are university educated, and have musical training. Audio metric testing indicated that both actresses have thresholds within the normal range.

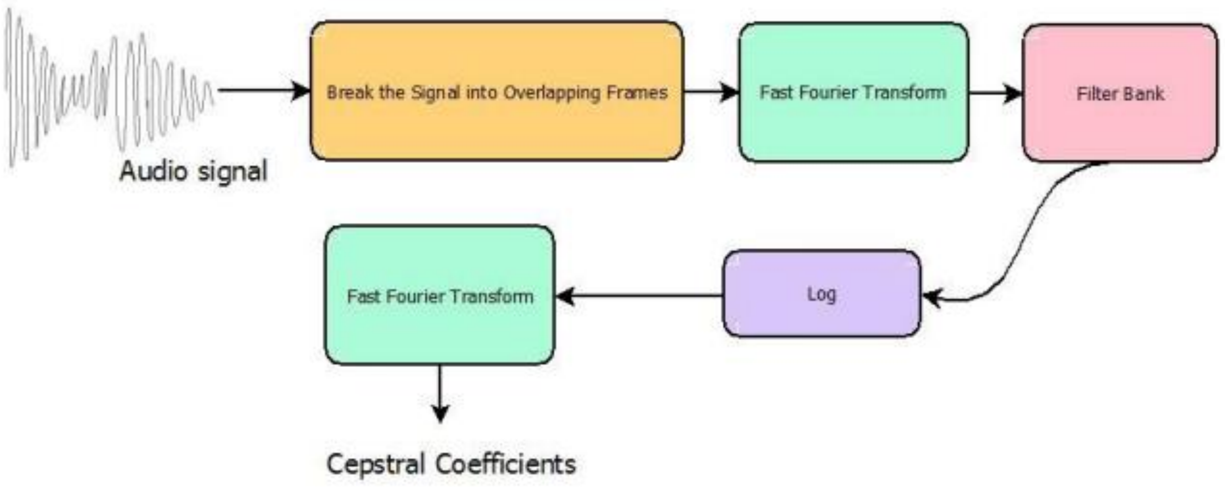
Crowd-sourced Emotional Multimodal Actors Dataset [CREMA-D]

- CREMA-D is a data set of 7,442 original clips from 91 actors. These clips were from 48 male and 43 female actors between the ages of 20 and 74 coming from a variety of races and ethnicities (African America, Asian, Caucasian, Hispanic, and Unspecified). Actors spoke from a selection of 12 sentences. The sentences were presented using one of six different emotions (Anger, Disgust, Fear, Happy, Neutral and Sad)

Feature Selection

Mel-frequency cepstral coefficient

- Cepstrum is the rate of change information in spectral bands. In conventional temporal signal analysis, any periodic component (for example, echoes) appears as sharp peaks in the associated frequency spectrum. This is achieved by performing a Fourier transform on the time signal.



Zero Crossing Rate

- The zero-crossing rate (ZCR) is the rate at which a signal changes from positive to zero to negative or from negative to zero to positive.

Feature Selection

Chroma Short-time Fourier transform(STFT)

- An audio's Chroma value essentially represents the strength of the twelve various pitch classes used to study music. They may be used to differentiate the pitch class profiles of audio streams. STFT encodes information concerning pitch categorization and signal structure. It represents the spike as a series of high and low numbers.

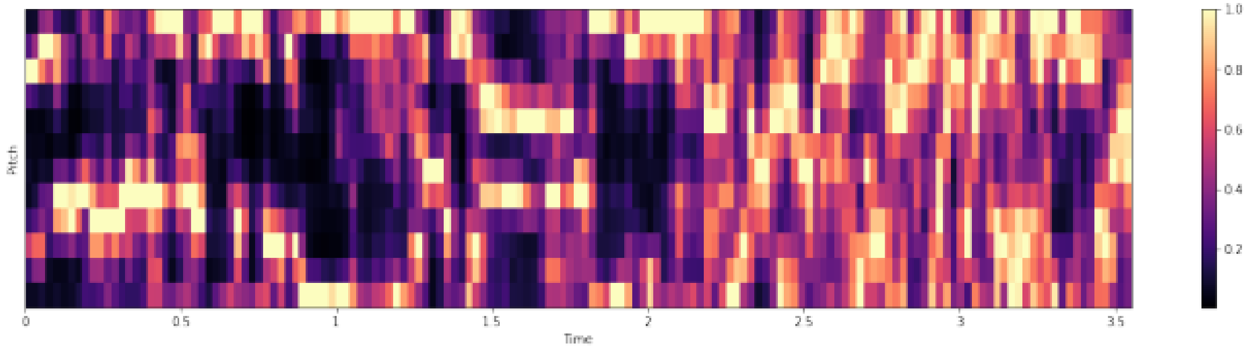


FIGURE 3.4: ChromaSTFT of an Angry emotion from CREMA−D dataset

Data Augmentation

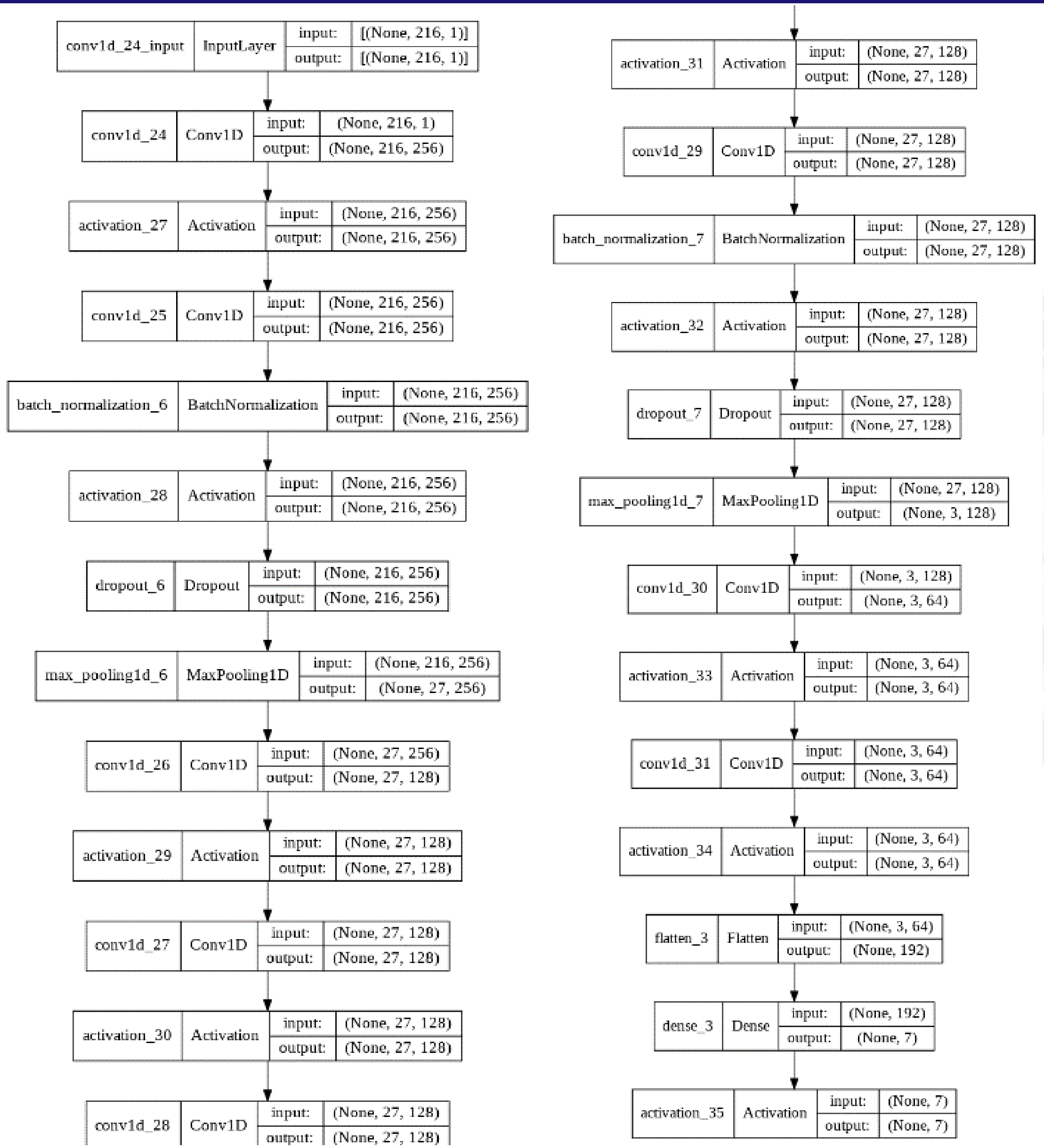
The techniques used to increase the amount of data by adding slightly modified copies of already existing data or newly created synthetic data from existing data. It acts as a regularizer and helps reduce overfitting when training a machine learning model. It is closely related to oversampling in data analysis. In order for the model to be resilient to unknown the following augmentation techniques are applied to the data a static noise in the background, phase shift, signal stretch, pitch, dynamic change are added to the data.

Experiments

&

Results

Model-1 | Test Accuracy : 50%



conv1d_24_input InputLayer input: [(None, 216, 1)] output: [(None, 216, 1)]

conv1d_24 Conv1D input: (None, 216, 1) output: (None, 216, 256)

activation_27 Activation input: (None, 216, 256) output: (None, 216, 256)

conv1d_25 Conv1D input: (None, 216, 256) output: (None, 216, 256)

batch_normalization_6 BatchNormalization input: (None, 216, 256) output: (None, 216, 256)

activation_28 Activation input: (None, 216, 256) output: (None, 216, 256)

dropout_6 Dropout input: (None, 216, 256) output: (None, 216, 256)

max_pooling1d_6 MaxPooling1D input: (None, 216, 256) output: (None, 27, 256)

conv1d_26 Conv1D input: (None, 27, 256) output: (None, 27, 128)

activation_29 Activation input: (None, 27, 128) output: (None, 27, 128)

conv1d_27 Conv1D input: (None, 27, 128) output: (None, 27, 128)

activation_30 Activation input: (None, 27, 128) output: (None, 27, 128)

conv1d_28 Conv1D input: (None, 27, 128) output: (None, 27, 128)

activation_31 Activation input: (None, 27, 128) output: (None, 27, 128)

conv1d_29 Conv1D input: (None, 27, 128) output: (None, 27, 128)

batch_normalization_7 BatchNormalization input: (None, 27, 128) output: (None, 27, 128)

activation_32 Activation input: (None, 27, 128) output: (None, 27, 128)

dropout_7 Dropout input: (None, 27, 128) output: (None, 27, 128)

max_pooling1d_7 MaxPooling1D input: (None, 27, 128) output: (None, 3, 128)

conv1d_30 Conv1D input: (None, 3, 128) output: (None, 3, 64)

activation_33 Activation input: (None, 3, 64) output: (None, 3, 64)

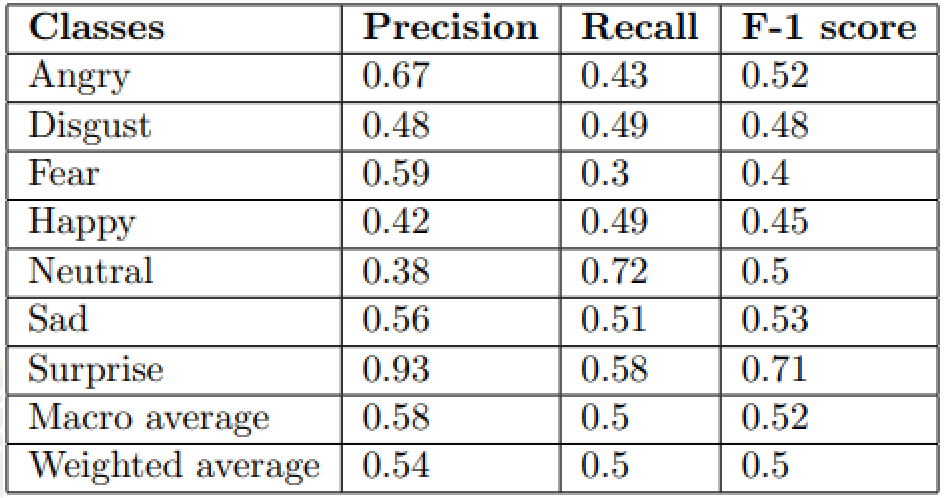
conv1d_31 Conv1D input: (None, 3, 64) output: (None, 3, 64)

activation_34 Activation input: (None, 3, 64) output: (None, 3, 64)

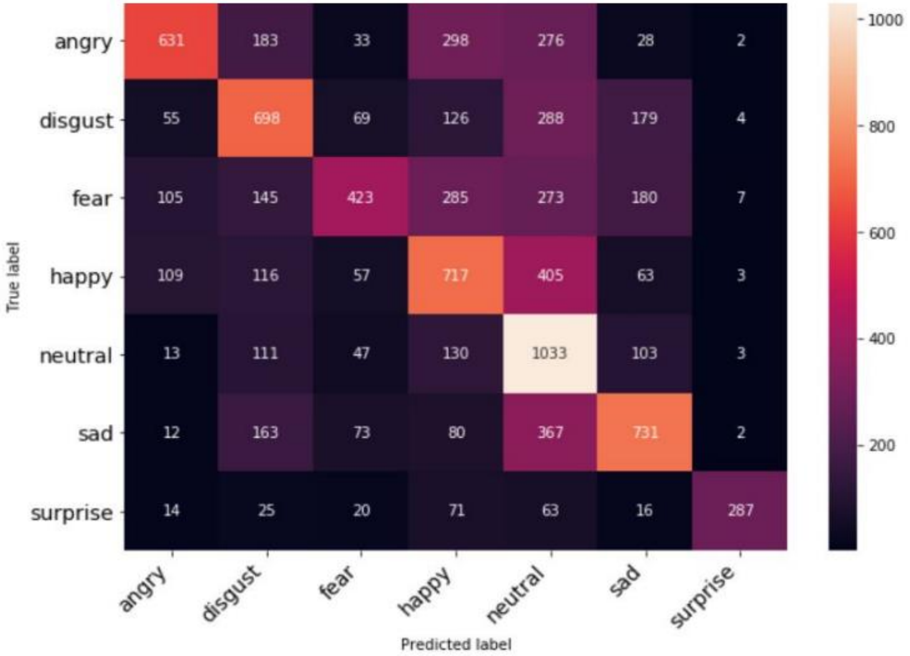
flatten_3 Flatten input: (None, 3, 64) output: (None, 192)

dense_3 Dense input: (None, 192) output: (None, 7)

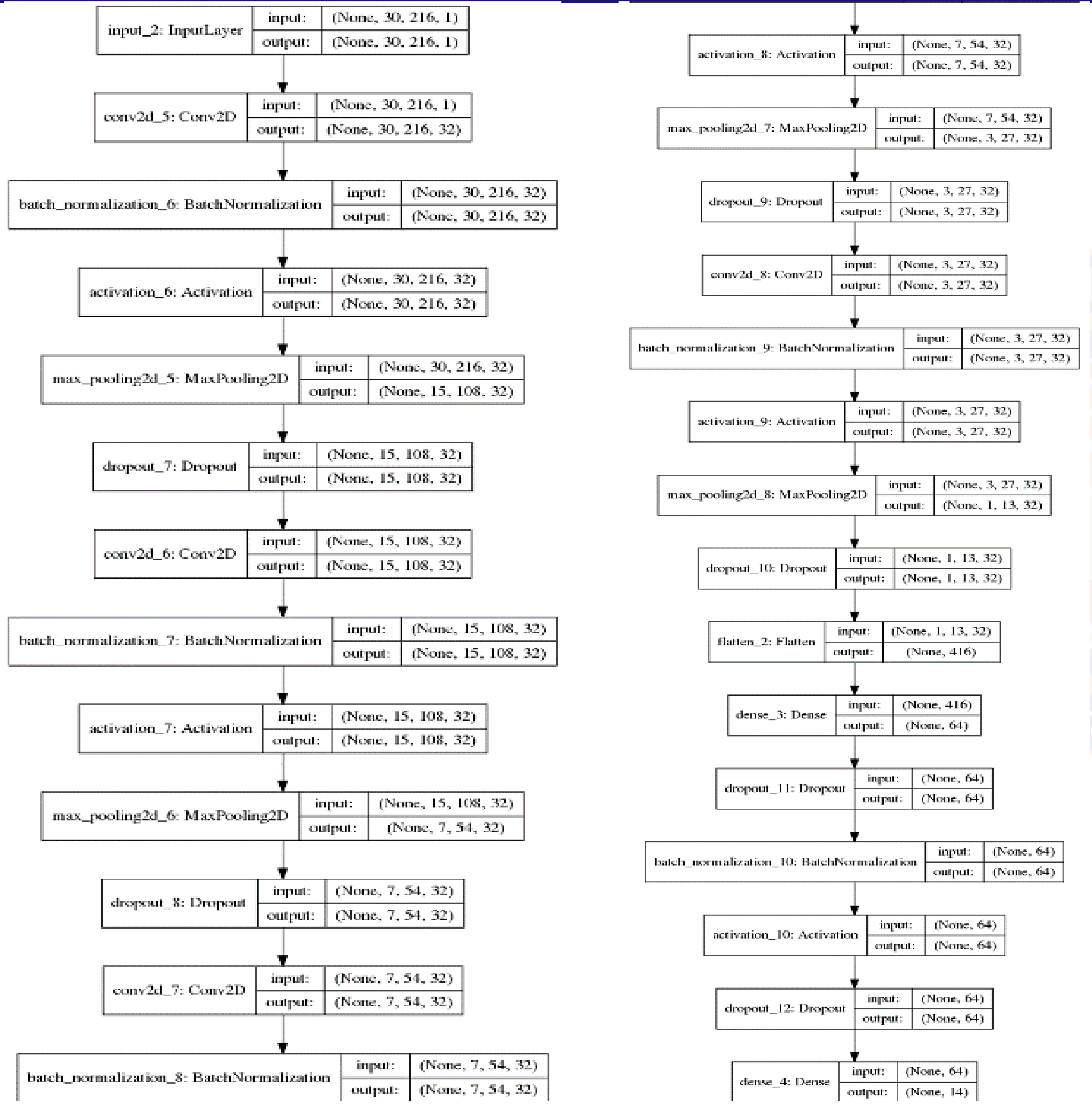
activation_35 Activation input: (None, 7) output: (None, 7)

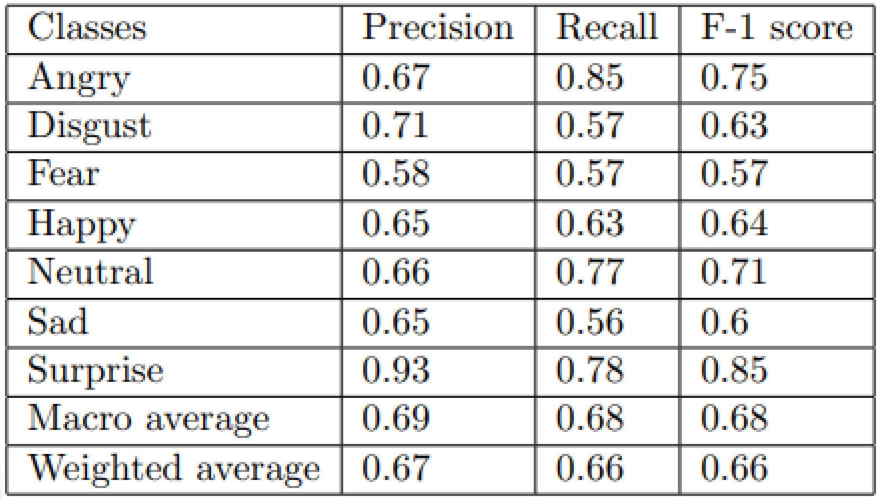


Classes	Precision	Recall	F-1 score
Angry	0.67	0.43	0.52
Disgust	0.48	0.49	0.48
Fear	0.59	0.3	0.4
Happy	0.42	0.49	0.45
Neutral	0.38	0.72	0.5
Sad	0.56	0.51	0.53
Surprise	0.93	0.58	0.71
Macro average	0.58	0.5	0.52
Weighted average	0.54	0.5	0.5

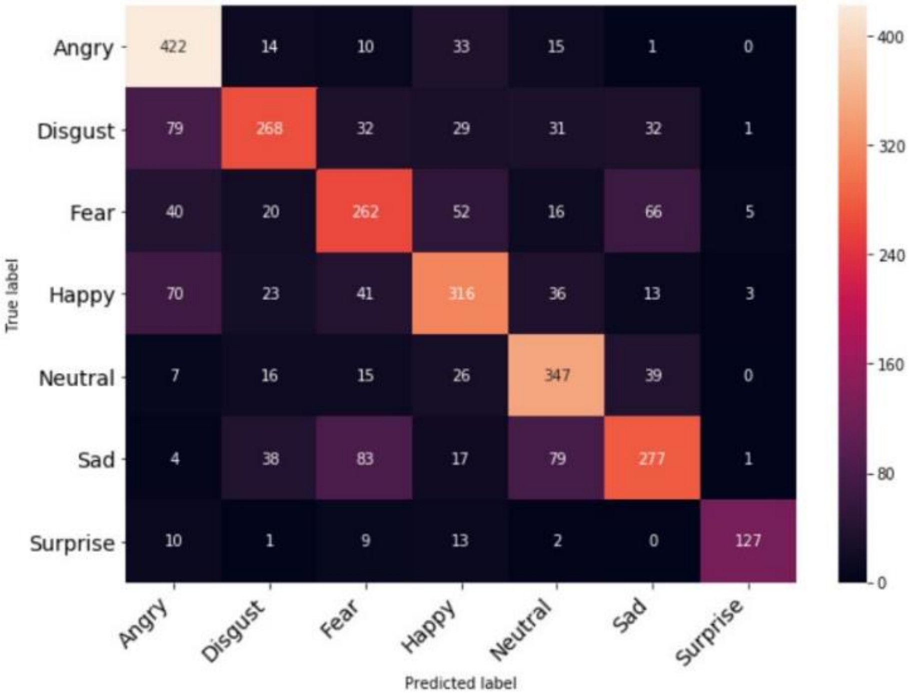


Model-2 | Test Accuracy : 66%

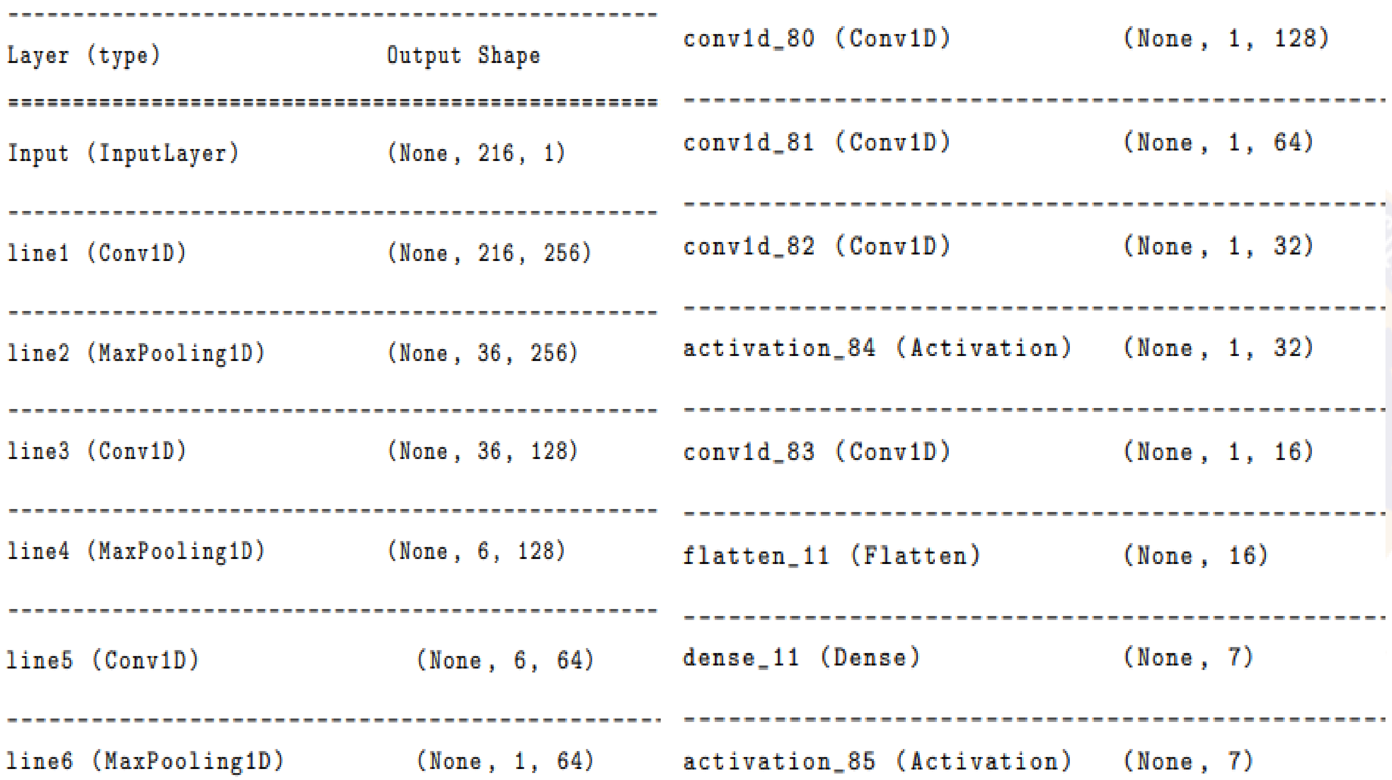

input_2: InputLayer | input: (None, 30, 216, 1) | output: (None, 30, 216, 1)
conv2d_5: Conv2D | input: (None, 30, 216, 1) | output: (None, 30, 216, 32)
batch_normalization_6: BatchNormalization | input: (None, 30, 216, 32) | output: (None, 30, 216, 32)
activation_6: Activation | input: (None, 30, 216, 32) | output: (None, 30, 216, 32)
max_pooling2d_5: MaxPooling2D | input: (None, 30, 216, 32) | output: (None, 15, 108, 32)
dropout_7: Dropout | input: (None, 15, 108, 32) | output: (None, 15, 108, 32)
conv2d_6: Conv2D | input: (None, 15, 108, 32) | output: (None, 15, 108, 32)
batch_normalization_7: BatchNormalization | input: (None, 15, 108, 32) | output: (None, 15, 108, 32)
activation_7: Activation | input: (None, 15, 108, 32) | output: (None, 15, 108, 32)
max_pooling2d_6: MaxPooling2D | input: (None, 15, 108, 32) | output: (None, 7, 54, 32)
dropout_8: Dropout | input: (None, 7, 54, 32) | output: (None, 7, 54, 32)
conv2d_7: Conv2D | input: (None, 7, 54, 32) | output: (None, 7, 54, 32)
batch_normalization_8: BatchNormalization | input: (None, 7, 54, 32) | output: (None, 7, 54, 32)
activation_8: Activation | input: (None, 7, 54, 32) | output: (None, 7, 54, 32)
max_pooling2d_7: MaxPooling2D | input: (None, 7, 54, 32) | output: (None, 3, 27, 32)
dropout_9: Dropout | input: (None, 3, 27, 32) | output: (None, 3, 27, 32)
conv2d_8: Conv2D | input: (None, 3, 27, 32) | output: (None, 3, 27, 32)
batch_normalization_9: BatchNormalization | input: (None, 3, 27, 32) | output: (None, 3, 27, 32)
activation_9: Activation | input: (None, 3, 27, 32) | output: (None, 3, 27, 32)
max_pooling2d_8: MaxPooling2D | input: (None, 3, 27, 32) | output: (None, 1, 13, 32)
dropout_10: Dropout | input: (None, 1, 13, 32) | output: (None, 1, 13, 32)
flatten_2: Flatten | input: (None, 1, 13, 32) | output: (None, 416)
dense_3: Dense | input: (None, 416) | output: (None, 64)
dropout_11: Dropout | input: (None, 64) | output: (None, 64)
batch_normalization_10: BatchNormalization | input: (None, 64) | output: (None, 64)
activation_10: Activation | input: (None, 64) | output: (None, 64)
dropout_12: Dropout | input: (None, 64) | output: (None, 64)
dense_4: Dense | input: (None, 64) | output: (None, 14)



Classes	Precision	Recall	F-1 score
Angry	0.67	0.85	0.75
Disgust	0.71	0.57	0.63
Fear	0.58	0.57	0.57
Happy	0.65	0.63	0.64
Neutral	0.66	0.77	0.71
Sad	0.65	0.56	0.6
Surprise	0.93	0.78	0.85
Macro average	0.69	0.68	0.68
Weighted average	0.67	0.66	0.66

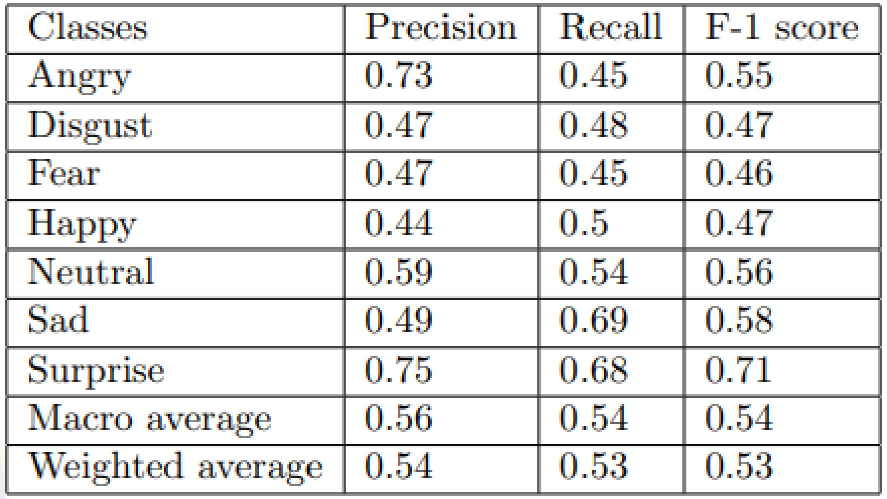


Model-3 | Test Accuracy : 53 %

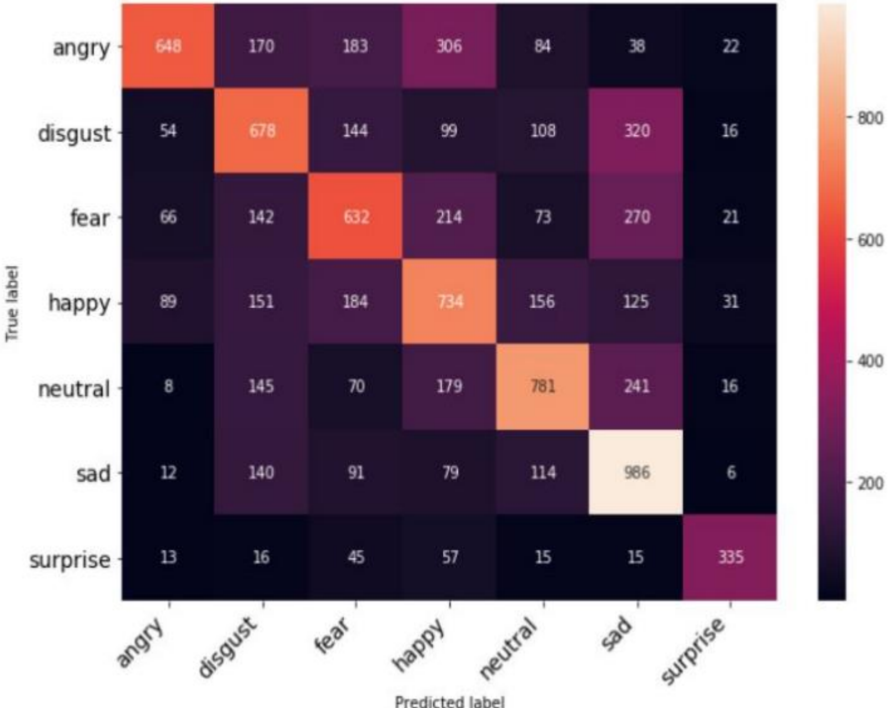


```
____________________________________________________
Layer (type)                 Output Shape
====================================================
Input (InputLayer)           (None, 216, 1)
____________________________________________________
line1 (Conv1D)               (None, 216, 256)
____________________________________________________
line2 (MaxPooling1D)         (None, 36, 256)
____________________________________________________
line3 (Conv1D)               (None, 36, 128)
____________________________________________________
line4 (MaxPooling1D)         (None, 6, 128)
____________________________________________________
line5 (Conv1D)                 (None, 6, 64)
____________________________________________________
line6 (MaxPooling1D)           (None, 1, 64)

conv1d_80 (Conv1D)             (None, 1, 128)
________________________________________________________
conv1d_81 (Conv1D)             (None, 1, 64)
________________________________________________________
conv1d_82 (Conv1D)             (None, 1, 32)
________________________________________________________
activation_84 (Activation)     (None, 1, 32)
________________________________________________________
conv1d_83 (Conv1D)             (None, 1, 16)
________________________________________________________
flatten_11 (Flatten)           (None, 16)
________________________________________________________
dense_11 (Dense)               (None, 7)
________________________________________________________
activation_85 (Activation)     (None, 7)
```



Classes	Precision	Recall	F-1 score
Angry	0.73	0.45	0.55
Disgust	0.47	0.48	0.47
Fear	0.47	0.45	0.46
Happy	0.44	0.5	0.47
Neutral	0.59	0.54	0.56
Sad	0.49	0.69	0.58
Surprise	0.75	0.68	0.71
Macro average	0.56	0.54	0.54
Weighted average	0.54	0.53	0.53



Conclusion :

- From the experiments use of 2D Convolution has yielded better accuracies compared to other architectures
- The weakly supervised methods can be of great help in similar problem statements. The use of Crema-D dataset which is a crowd sourced dataset has helped to add many training examples and has helped to demonstrate the use weakly supervised learning.
- From the results its evident that when the Speech emotion recognition models have a bias towards the speaker voice characteristics and the model tends to over fit when deeper architectures are used with speakers from different nationalities.
- The model performs well with shallow architectures. Even the results using dimensionality reduction techniques as Auto encoders provide higher accuracies, the encoders could be used to clear out unneeded noise in the data.
- The paradigm of weak supervision has been fruitful and has helped in yielding decent results in classifying the emotions Neutral, Happy, Fear, Surprise, Sad, Disgust and Angry. But however there has been an overlap when recognizing the emotions Happy - Surprise, Disgust - Angry, Sad-Fear.
- The accuracy of the models improve when they are used to predict genders but when the audio data set is analyzed only for emotions the overlap percentage increases.

Future Work:

- The results from the work show promising results in using CNN and Auto encoder architectures with weak supervision, hence these could be explored deeper.
- Architectures like LSTM, RNN also could be explored.
- The decrease in accuracy is because the data set is divided into two genders, hence more complex architectures which aren’t biased towards gender could be explored.
- Different encoders, even in combination, such as de-noising AE, are likely to enhance the results when used along with weak supervision.
- Datasets with more emotions could be used along with weak supervision to improve the model performance.

Thank You!