Emotion Detection from Human Speech

Venkata Sai Praneeth Kandula, Rohith Movva, Yashwanth Lingareddy

Introduction:

There has been a lot of Research and Development that has been going in recent years especially in the fields of Speech Processing and Facial Recognition. But a large number of Artificially Intelligent systems existing today that utilize these concepts are focused on identities i.e. identifying if a voice or a facial image belongs to a particular person. The future of these systems lies in not just identifying but detecting and recognizing emotional states from sound and images. We are all familiar with Google AI that succeeded in mimicking naturally occurring speech patterns in humans which is a huge leap in building AI systems that are indistinguishable from humans. An important aspect that differentiates AI systems from humans is "Emotion" and the future lies in trying to build AI systems that are emotionally intelligent.

Emotion Recognition and Classification has great potential for application in a variety of industries including robotics, text-to-speech systems to synthesize speech that is more emotionally natural, and other systems and smart devices with interactive user-interaction. Emotion Recognition is also being widely utilized to solve a number of real-world problems. For Example, Companies like Unilever have started incorporating AI systems into their recruitment processes, to automatically screen candidates by emotional intelligence. The AI systems quantify characteristics like honesty, confidence, passion, curiosity, nervousness etc. which help recruiters make better decisions, while simultaneously reducing inherent biases that recruiters might have based on attraction, gender or ethnicity.

Emotion Recognition is also being used in financial fraud detection to screen for insurance claim and loan application fraud. By capturing subliminal behavior of applicants and mapping it to their personality which is then used to asses risk.

Mobile Applications in the Mental Health space have flooded the market in recent times, but most of these are text based and utilize a question and answer format to detect and recognize emotional states, Imagine If these apps could instead record a video of people talking and automatically analyze and detect their emotional states.

Approach:

It is evident that in Human-Machine interaction, it is important that emotional states in human speech are fully and accurately perceived. However, detecting the emotional content of an audio signal is often difficult and has several challenges. The first challenge being able to define what emotion means and how it can be categorized in a precise way.

Another critical challenge is to determine features that influence the recognition of emotion in speech which requires a significant amount of domain knowledge. The existence of different genders, speakers, dialects, speaking styles and accents only makes the problem much worse as it has direct affect on the features such as energy and pitch contours. Existing emotion detection methods make use of feature selection i.e. these algorithms create a set of rich features that are related to speech recognition including fundamental frequency or pitch, energy, speaking rate and spectral coefficients which are then used for classification.

Leveraging advances in computer vision and image recognition, in this project we look at another approach to solve the problem of detecting emotion from an audio signal. By capturing the characteristics of an audio signal and representing it in the form of an image, the entire problem of recognizing emotion is then reduced to an image classification task.

Background and Methodology:

The simplest forms in which an audio signal can be represented is a waveform as seen in fig 1 which measures the change in amplitude on the y-axis, over time measured by the x-axis.

Fig 1 illustrates the waveform of the simplest type of sound which consists of just one tone.

The Zero of the y-axis represents atmospheric pressure and the waveform represents the variation in pressure caused by the sound also known as amplitude over time. In contrast to this waveform containing just a single tone, naturally occurring speech is much more complex and variable reflecting the dynamic nature of speech articulation. Fig 2 illustrates the waveform of a speech.

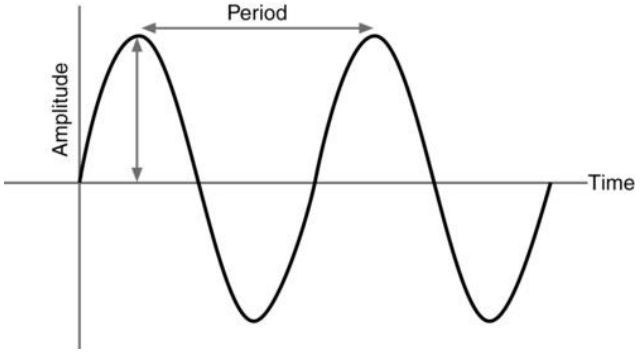

Figure 1 Pure Tone Waveform

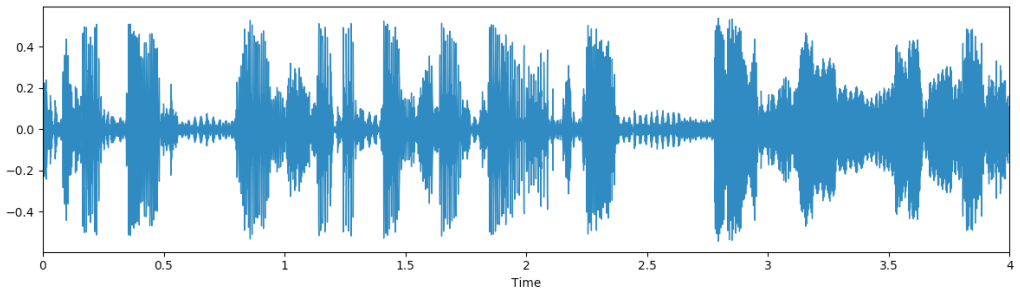

Figure 2 Waveform of Speech

The wave form does not capture complete information regarding the audio signal. An audio signal is typically a convolution or a superimposition of a number of waves which operate at different frequencies. To capture this information, we use the spectrogram which is a frequency domain representation of an audio signal.

As can be seen in fig 3, the signal labelled sum is the waveform that we have seen previously which can be broken down into its component waves with different frequencies.

So, the spectral representation in the frequency domain thus gives us a lot more information than a simple waveform.

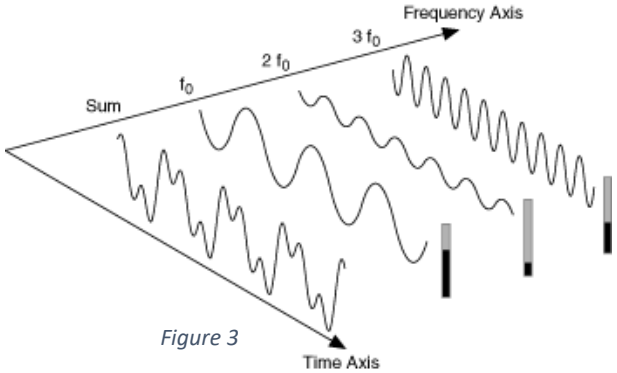

Figure 3

This conversion from time-domain to frequency domain is can be performed using the Fourier Transform. A Mel-spectrogram can be seen in fig 4, the x-axis represents time, the y-axis the frequency, and the intensity or color of at each point represents the magnitude of amplitude or the amount of energy present in each frequency at that particular point in time. The brighter the color, the more energy is present in the sound at that frequency.

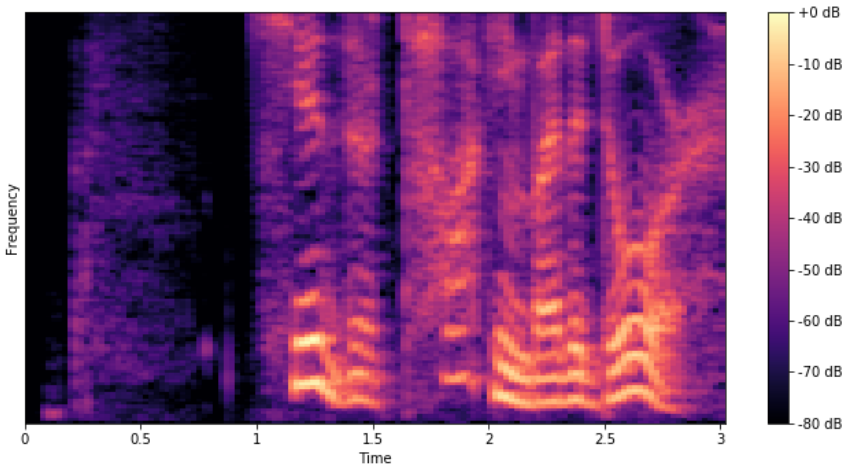

Figure 4 Mel-Spectrogram

A Mel-spectrogram as opposed to a normal spectrogram mimics the frequency mappings of the neurons in the cochlea of the inner ear and thus takes into account human aural perception. This has been proven to be more useful in speech recognition and processing when compared to a simple spectrogram.

So. After obtaining the audio signal, we preprocess it and then using the Fast Fourier Transform, obtain the Mel-Spectrogram for each of the signals. These images are then fed to a Classifier built using a Convolutional Neural Network Architecture.

Dataset:

In order to build the model, we used an openly available and labelled dataset called the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) which was the work of researchers in Ryerson University, Toronto published in May 2018. The dataset includes recording samples of 24 actors, 12 male and 12 female, depicting 8 emotions of neutral, calm, happy, sad, angry, fearful, disgust and surprised.

Each actor performs two repetitions with varying intensities, “Normal” and “Strong” of two different statements, “Kids are talking by the door” and “Dogs are sitting by the door”. The “Neutral” emotion is used as a baseline and hence has only a normal intensity. As depicted in Fig 5, each a ctor has a total of 60 Voice recordings which gives us a total of 1440 voice recordings.

For validating the dataset 247 raters took part who were presented with the recordings and were asked to make three judgments: category of the emotion, strength of the emotion and genuineness of the emotion. The mean proportion that the raters correctly identified for the audio recordings was 67% for strong intensity and 53% for normal intensity, an average of 62%. So it’s not actually an easy task even for humans to ascertain emotion especially when there is no visual stimulus.

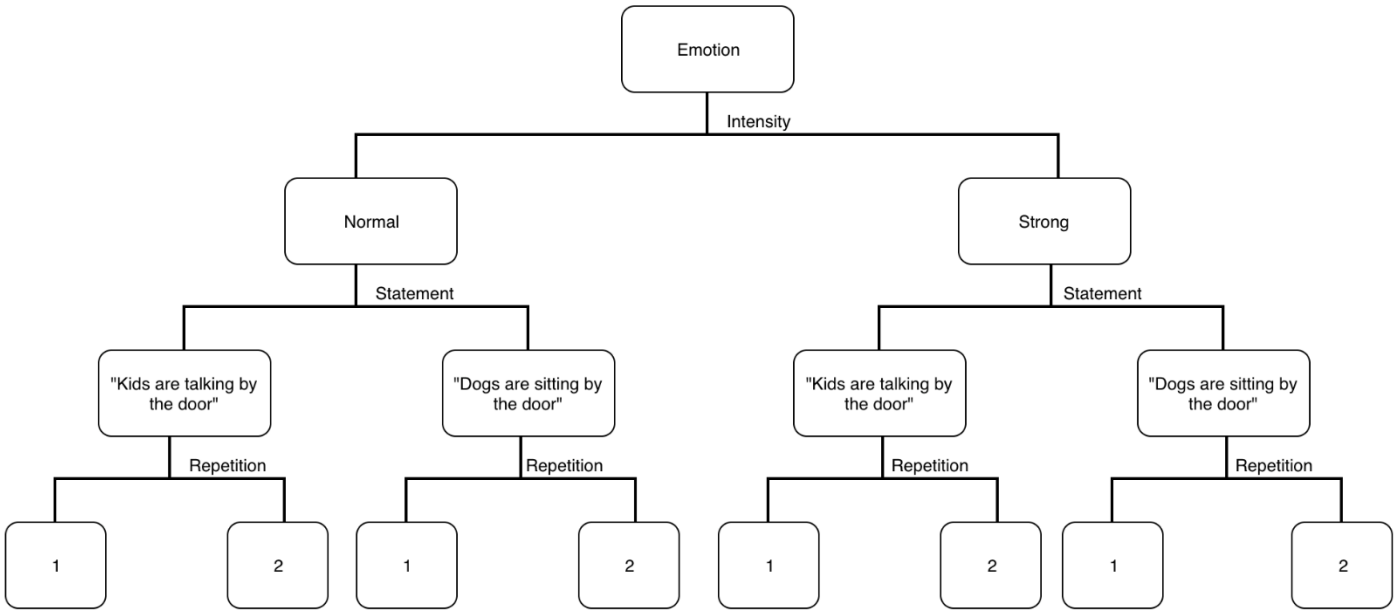

Figure 5 RAVDESS Dataset

Data Processing:

The Librosa library for python was used to process the audio signals. Firstly, all the audio signals had to be aligned so that they had the same length. Most of the recordings were around 3 seconds long and so we clipped all the audio files to 3 seconds. Two recordings were less than the three seconds and hence were removed from our training dataset. Another approach we could have tried was to add silence as padding at the end of these recordings, but since we had only a couple of files we went ahead and eliminated from our dataset.

The default sampling rate of Librosa being 22050Hz, each audio signal was stored as a one-dimensional NumPy array of length 66150. Functions from the Librosa package were then used to transform this signal into a Mel-Spectrogram which resulted in an image with dimensions of 128 X 130 pixels.

The 1438 images were reshaped into a 4-dimensional NumPy array with dimensions (1438,128,130,1) which is the convention for input data when working with a Convolutional Neural Network Architecture in Keras/TensorFlow.

This set of 1438 images were split into training and testing datasets. To make sure that our training and test datasets were totally independent, we used 20 random actors (10 female and 10 male) for training, and 4 actors for the test data.

Results:

Initially we built a traditional Random Forest model with 200 estimators and a maximum tree depth of 7 which gave us an average accuracy of about 30% on the test set and 95% on the training set. Considering that we have 8 different class labels, the accuracy of the initial random forest model was much better than the baseline accuracy of 12.5%.

We used Keras in python to build the neural network models. The initial convolutional neural network architecture that we used was the VGG-16 architecture developed by researchers at Oxford University for large scale image classification. The architecture consists of a series of convolutional and pooling layers with the layer size shrinking successively as shown in fig 6.

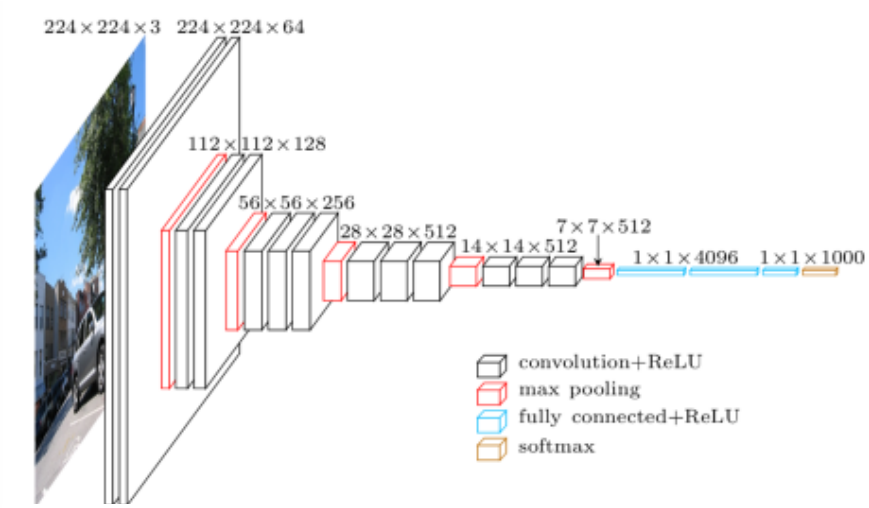

Figure 6 VGG-16 Architecture

Unfortunately, the VGG-16 Architecture took too long to train and did not give us any useful results, so we modified the architecture by simplifying it. The final architecture of the model we built can be seen in fig 7, which consists of a total of 3 convolutional layers, two with dropout, one pooling layer and finally a flat layer densely connected to the output layer. The activation functions used for the convolutional layers was ReLU (Rectified Linear Units) and a SoftMax activation for the output layer in order to get a probability distribution for each class. Three different optimizers were used including "Adam" optimizer, Stochastic Gradient Descent and RMSprop.

The model took about approximately 5 hours to train a 100 epochs. On the training set, the classifier almost reached an accuracy 100%, and 65% on the test set.

The evolution of the classification accuracy with the number of epochs can be seen in fig 8.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 126, 128, 256)	2560
max_pooling2d_1 (MaxPooling2	(None, 63, 64, 256)	0
conv2d_2 (Conv2D)	(None, 61, 62, 128)	295040
dropout_1 (Dropout)	(None, 61, 62, 128)	0
conv2d_3 (Conv2D)	(None, 59, 60, 128)	147584
dropout_2 (Dropout)	(None, 59, 60, 128)	0
flatten_1 (Flatten)	(None, 453120)	0
dense_1 (Dense)	(None, 8)	3624968

Total params: 4,070,152
Trainable params: 4,070,152
Non-trainable params: 0

Figure 7 CNN Architecture

Figure 8 Accuracy vs Number of Epochs

The confusion matrix of the model on the test set can be seen in fig 9 below. The rows in the figure represent the actual labels while the columns correspond to the class labels predicted by our model. As is evident from the confusion matrix, the model was very successful in correctly predicting the "Happy" class while it struggled in the "Neutral" class most probably because the "Neutral" class had just one intensity unlike two for the rest of the emotions.

	Angry	Calm	Disgust	Fearful	Happy	Neutral	Sad	Surprised
Angry	23	4	0	1	1	1	2	0
Calm	2	24	1	3	2	0	0	0
Disgust	1	1	17	4	5	0	1	3
Fearful	3	6	2	18	3	0	0	0
Happy	0	2	0	1	26	0	3	0
Neutral	6	0	0	1	0	7	0	2
Sad	4	0	0	0	5	0	19	4
Surprised	5	0	1	0	0	0	4	22

Figure 9 Confusion Matrix

Conclusions:

In this project we looked at an approach to recognize emotions from Human speech not by using traditional methods of extracting features but by mapping the audio signal to an image which reduces the problem to that of image classification. With advances in computer vision and increasing interest in areas of image classification and recognition, state-of-the-art Convolutional Neural Network architectures are available to solve this problem. We used the Keras/TensorFlow packages in python to

implement Convolutional Neural Networks and the best model resulted in accurately predicting the correct emotion 65% of the times as compared to a random choice of 12.5% which can be considered significant. Even more astonishing is the fact that our model performs better than the accuracy of human raters in the validation task as mentioned in the RAVDESS paper [1] where they were correctly able to identify only 62% of the emotions. In the future, we plan to also use not only audio but also video recordings of actors and attempt a multi modal approach to emotion classification which we believe would give us greater accuracy.

References:

1. Livingstone SR, Russo FA (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(5): e0196391. https://doi.org/10.1371/journal.pone.0196391
2. Librosa Audio and Music Signal Analysis in Python | SciPy 2015 | Brian McFee https://github.com/librosa/librosa
3. Perceptual audio features for emotion detection. Mehmet Cenk Sezgin,Bilge Gunsel and Gunes Karabulut Kurt. EURASIP Journal on Audio, Speech, and Music Processing. https://doi.org/10.1186/1687-4722-2012-16
4. Audio processing in TensorFlow. Dario Cazzani https://towardsdatascience.com/audio-processing-in-tensorflow-208f1a4103aa
5. Understanding Waveforms https://swphonetics.com/praat/tutorials/understanding-waveforms/
6. Mel-Spectrogram. https://en.wikipedia.org/wiki/Mel-frequency_cepstrum
7. VGG-16 Pre-Trained Model in Keras. https://gist.github.com/baraldilorenzo/07d7802847aaad0a35d3
8. CNNs in Python. https://www.datacamp.com/community/tutorials/convolutional-neural-networks-python
9. Keras. https://keras.io/
10. https://wearehuman.io/technology