Speech Emotion Recognition

Ishan Kumar Anand
isanand@ucsd.edu

1 Introduction

Speech Emotion Recognition (SER) refers to the process of identifying emotions in individuals based on the tone and nuances of their speech and dialogue. With the rise of voice assistants like Apple Siri, Amazon Alexa, and Google Assistant, these technologies have become commonplace in daily life. However, while these assistants are highly efficient at recognizing and responding to user commands, they lack the ability to detect the emotional state of a person based on the tone and inflection in their voice. This project aims to develop a module that can be integrated with existing voice assistants, enabling them to recognize the user's emotional state. The assistant's responses would then be adjusted to better align with the user's emotional tone, creating a more empathetic and context-aware interaction.

The goal of this project is to build an efficient model capable of detecting speech emotion. Initially, the project aimed to develop models based on lower-level architectures, with the potential to train the dataset with Vision Transformers. However, due to the challenges posed by insufficient and incomplete data, progress toward using Transformers was not feasible. Nevertheless, a simpler architecture was successfully implemented, yielding high accuracy in emotion detection. The model is designed to learn key features and attributes from speech data, enabling it to recognize emotions effectively.

Speech data can be represented in various forms, such as waveform representations, bit representations, spectrograms, and histograms. To train our model, we preprocess these representations in a way that allows for efficient feature extraction while minimizing the loss of important information. Most existing research on speech emotion recognition focuses on extracting the most relevant features from speech waveforms and feeding them into the model for learning. For this project, six emotion classes were considered: Happy, Angry, Fear, Sad, Disgust, and Neutral. Unfortunately, the available data for the emotions of Boredom and Surprise was insufficient, so these classes were excluded from the analysis. By concentrating on these six core emotions, we aim to build a model that is both accurate and robust in identifying the emotional state of a speaker based on their voice.

2 Related work

There has been considerable research in the field of Speech Emotion Recognition (SER), with most studies focusing on identifying patterns in the waveform of speech to classify emotions. Figure 1 shows an example of waveform of the speech.

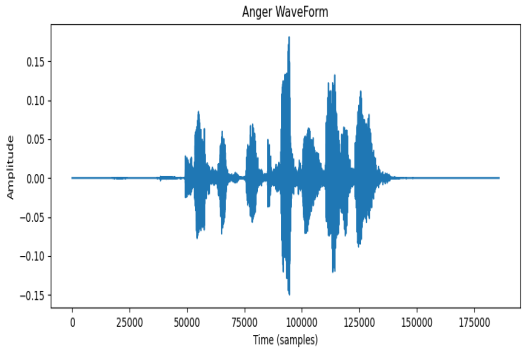

Figure 1: Anger Waveform

The initial research in this area, as discussed in (Ingale and Chaudhari, 2012)[1], employed classifiers such as Gaussian Mixture Models (GMM), K-Nearest Neighbors (K-NN), Support Vector Machines (SVM), Artificial Neural Networks (ANN), and Hidden Markov Models (HMM). These models were used after extracting speech features like Mel Frequency Cepstral Coefficients

(MFCC) and Linear Predictive Cepstral Coefficients (LPCC), achieving accuracy levels of up to 80

In contrast, (Devi et al., 2014) proposes a different approach by focusing on extracting energy from the speech waveform. The features are then used for training traditional machine learning models such as Logistic Regression, Unimodal Gaussian, and K-Nearest Neighbor. One key takeaway from this study is that extracting and leveraging energy-based features can be a promising strategy for emotion recognition in speech.

(Bankar et al., 2021) takes a different approach by using Mel Spectrograms as the feature extraction method. A Mel Spectrogram is a visual representation of the energy distribution of speech, similar to how humans perceive sound. In this paper, a transfer learning model based on VGG19, along with a custom Convolutional Neural Network (CNN) architecture, was used to learn speech vectors. This approach demonstrates the effectiveness of deep learning models in extracting features from spectrograms for emotion recognition.

(Swain et al., 2018) focuses on the essential features that can be extracted from speech for emotion recognition. This paper provides valuable insights into the speech features that are critical for SER, many of which are used in the current study. It also offers useful resources for speech databases that can be leveraged to train models for emotion recognition.

Finally, (Khalil et al., 2019) compares several deep learning models for emotion recognition, including Deep Boltzmann Machines (DBM), Deep Belief Networks (DBN), and Auto-encoders. This paper discusses how different algorithms can be applied to SER, elaborating on the features extracted by each model and their impact on emotion classification.

3 Dataset

CREMAD is a dataset consisting of 7,442 audio clips from 91 actors, including 43 female voices and 48 male voices, aged between 20 and 74 years. To ensure diversity and avoid bias, the dataset includes actors of various ethnicities, including African American, Asian, Caucasian, Hispanic, and some unspecified ethnic groups. The dataset features 6 distinct emotions: Fear, Happiness, Anger, Disgust, Sadness, and Neutral. These emotions are expressed across 12 different sentences, each spoken in 4 different tones (low, medium, high, and an unspecified mid-range tone). The variety of voices and emotional expressions makes this dataset suitable for training complex emotion recognition models.

EMODB, the German Speech Database, was created by the Institute of Communication Science at the Technical University of Berlin, Germany. It contains 535 audio utterances recorded by 10 students (5 male and 5 female). The dataset includes emotions such as Anger, Boredom, Anxiety, Happiness, Sadness, Disgust, and Neutral. The recordings also vary in terms of frequency, ranging from 48 kHz to 16 kHz. However, since the dataset is based on German speech, models trained exclusively on EMODB may perform best on German language data.

The RAVDESS (Ryerson Audio-Visual Database of Emotional Speech and Song) dataset includes voice recordings from 24 professional speakers with a central North American accent—12 male and 12 female. The recordings were made with varying intensity levels, ranging from Normal to Neutral to Strong. The emotions expressed include Fear, Calm, Happy, Surprise, Anger, Disgust, and Sadness. In total, the dataset includes 1,440 voice recordings, with each speaker providing 60 different speech trials.

The SAVEE dataset, recorded by postgraduate students at the University of Surrey, features recordings from 15 speakers (aged 27 to 31) and is based on a European accent. The dataset contains 7 emotion categories: Anger, Fear, Disgust, Sadness, Surprise, Happiness, and Neutral. The recordings include 15 utterances per speaker, with 3 common phrases, 2 emotion-specific sentences, and 10 generic sentences. In total, the dataset consists of 120 recordings per speaker.

Finally, the TESS dataset consists of 2,800 audio files recorded by two voice actresses aged 26 and 64. The dataset includes 8 different emotions: Anger, Fear, Happiness, Pleasant Surprise, Sadness, Disgust, and Fear. The recordings capture a variety of emotional expressions, contributing to the diversity of speech emotion data. We have used this dataset combine in this Project.

For our project, the audio format chosen is .wav rather than other formats like .mp3, .awb, etc. This decision was made because .wav files retain full audio content without any compression or data

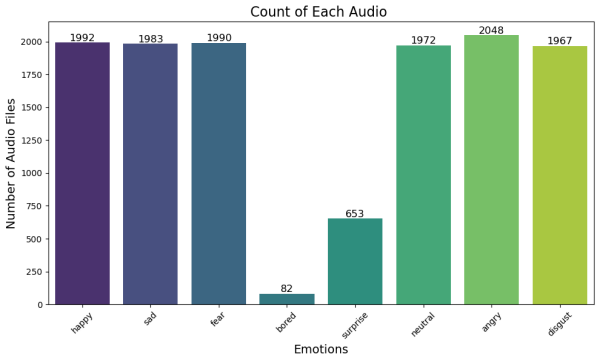

Figure 2: Count Plot of all Emotion Dataset

loss, unlike other formats where the audio data may be deduced or compressed, potentially affecting the quality and accuracy of emotion recognition.

The emotion categories included in our dataset are Angry, Fear, Disgust, Happy, Neutral, and Sad. Figure 2 illustrates the frequency distribution of each emotion in the dataset, which will be used for training and testing the model. We have excluded Surprise and Boredom from the analysis, as the number of data points for these two emotions is insufficient compared to the rest of the emotion categories, which would lead to an imbalance in the dataset.

3.1 Data preprocessing

Since my dataset consists of five different sets of data, the first step in preprocessing is to combine all of them into a single dataframe that includes both the emotions and the audio data. As this is a multi-class classification problem, I have assigned numerical labels to the emotions: 0 for 'Happy', 1 for 'Angry', 2 for 'Fear', 3 for 'Sad', 4 for 'Disgust', and 5 for 'Neutral'.

Initially, our plan was to use a Vision Transformer (ViT), and one of the first considerations was the number of data points required to train such a model effectively. Transformers, by design, have vast architectures and a high number of parameters, which means they tend to require large amounts of data for optimal performance. To address this, I decided to augment the voice data, specifically by adding noise to the recordings. Other augmentation techniques such as stretching or amplification were avoided, as they could alter the original nature of the data. Adding noise, however, seemed like a realistic approach, as background noise is common in real-world audio recordings, and such an augmentation would help the model generalize better.

Following this, I proceeded to extract features from the audio data. The initial plan was to extract the Mel-Spectrogram, a representation of the audio that reflects human hearing perception, as described in (Bankar et al., 2021). However, I quickly realized that relying on the Mel-Spectrogram alone did not provide enough information for the model to learn the associations between the data points effectively. As a result, I expanded the feature extraction process to include a variety of audio features: Mel-frequency Cepstral Coefficients (MFCC), Root Mean Square (RMS), Zero Crossing Rate (ZCR), Spectral Centroid, Spectral Roll-Off, and Chroma Features.

MFCCs are crucial for capturing the timbral properties of the audio, which essentially help differentiate emotional states in speech, such as excitement, anger, or sadness. These coefficients represent the frequencies in a way that aligns with human auditory perception, making them highly useful for speech and emotion recognition. The logarithmic transformation of the MFCCs produces the Mel-Spectrogram, which is a perceptible energy representation of the audio, as shown in Figure 3. This transformation effectively reduces the dimensionality of the audio data while preserving important acoustic features.

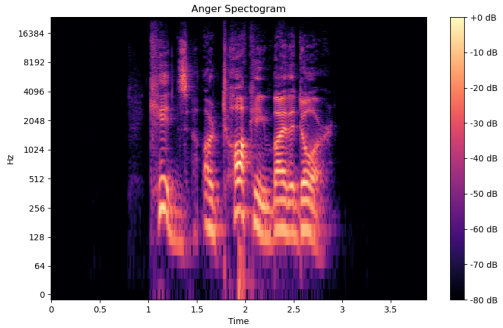

Figure 3: Mel-Spectogram of a Anger Speech

Root Mean Square (RMS) is another important feature that measures the energy distribution of the audio signal. It is strongly correlated with the emotional intensity of the speech. For instance, deeper, more fearful voices tend to have higher RMS values, while calm or sad voices typically exhibit lower RMS levels. RMS is a simple yet effective way to capture speech intensity, which plays a significant role in emotional arousal and

recognition.

Zero Crossing Rate (ZCR) measures how frequently the audio signal crosses the zero axis within a given time frame. This feature helps capture rhythmic aspects of speech and can provide insights into the intensity and energy of the speaker. It is particularly useful for distinguishing between emotional states that involve high energy (such as excitement or anger) and those that are more subdued (such as sadness or calmness).

Spectral Centroid and Spectral Roll-Off are both features that help capture the sharpness and harmonic content of the voice. The Spectral Centroid represents the ”center of mass” of the spectrum, which can indicate the brightness or sharpness of the sound. Spectral Roll-Off measures the frequency below which a certain percentage of the total spectral energy is contained, providing insight into the tonal quality and harmonic density of the speech. These features are helpful for identifying the overall texture of the speech, which can be linked to various emotional states.

Lastly, Chroma Features focus on the harmonic distribution within the audio. They capture how pitch classes are distributed across the signal, helping identify the underlying harmonic structure of the speech. This can be particularly useful in emotion recognition, as emotional expressions often have distinct harmonic patterns that differentiate them from neutral or calm speech.

In conclusion, combining these diverse features provides a rich and comprehensive set of descriptors that can be used as input for machine learning models. They offer a nuanced representation of the speech data, capturing both temporal and spectral information that is essential for effective emotion recognition. Together, these features provide the necessary data for training models that can accurately predict the emotional state of the speaker based on their speech.

3.2 Data annotation

Since all the raw data for this project was sourced from well-established and reputable websites specializing in Speech Emotion Recognition, there was no need for additional annotations from my side. These datasets already came with pre-labeled emotion categories. For the purpose of this project and to align with our learning objectives, I have labeled the various emotions as follows Happy(0), Angry(1), Fear(2), Sad(3), Disgust(4), Neutral(5)

Given that the data already contained the necessary emotion labels, no further manual annotations were required from my end for this particular project. The provided labels allowed me to focus on the code and the technical aspects of the Speech Emotion Recognition task.

4 Baselines

The initial approach of this project was to train the dataset features extracted from the audio using Vision Transformers (ViT). The idea was to treat the feature vector as a large 2D image, which would then be divided into blocks (patches) and fed to each node of the transformer model. However, there were a few drawbacks with this approach. First, a larger dataset was required to train the ViT properly, as insufficient data led to underfitting. Second, as referenced in the original ViT paper (Dosovitskiy, 2020), breaking an image into 16 blocks, for example, could lead to around 16 billion parameters in the transformer model. This made the computation cost enormous, rendering the approach impractical for our project. Due to these issues, we decided to abandon the Vision Transformer architecture for this task.

According to fellow researchers at the University of California, San Diego (Zhang et al., 2019), a Convolutional Neural Network (CNN) approach was tested for audio recognition, specifically for sounds such as bass drum, gurgling, finger snapping, harmonica, and hi-hat. They explored various methods for audio recognition, including weighted and unweighted transfer learning, and deep CNN architectures. Their model worked well for recognizing some sounds but failed on others. Based on this analogy, we designed two different CNN architectures for our own audio classification task.

The first architecture was a more complex structure, with four convolutional layers stacked sequentially, followed by fully connected linear layers. While this architecture achieved high accuracy in both the training and testing datasets, it tended to overfit, as it memorized the parameters rather than generalizing well to unseen data. As a result, we turned to the second architecture, which used only one convolutional layer. However, this approach did not improve the results and produced similar performance to the first, indicating that the model was still not able to generalize effectively.

Hence my next approach was to test with Long

Short Term Memory, which worked out well.

5 Approach

My next approach was to perform model training using Long Short Term Memory (LSTM), which yielded promising results. Unlike the earlier convolutional architectures, which struggled to generalize and thus didn't perform well, the LSTM architecture showed better capacity for generalization. Notably, the LSTM model didn't overfit the data, which indicates that the model was learning the relevant patterns rather than memorizing specific parameter values.

As we observed earlier, the architectures began to generalize more effectively over time, so I opted for a more simplified design with just a single LSTM layer, followed by only three fully connected layers of a Convolutional Neural Network. Additionally, I made another key modification by removing noisy data from the training set. The rationale behind this was that noisy data could lead the model to learn irrelevant features, which could hinder its ability to focus on the true underlying patterns in the data. To ensure robust evaluation, the entire dataset was split into 90 percentage for training and 10 percentage for testing. The performance of the LSTM model was quite encouraging, with the training and testing accuracies following similar trends, confirming that the model was not overfitting the data.

This balanced setup not only improved the model's ability to generalize but also demonstrated the importance of noise reduction in enhancing model performance. The LSTM model's efficiency in learning the relevant temporal dependencies without overfitting further validated its effectiveness for this task.

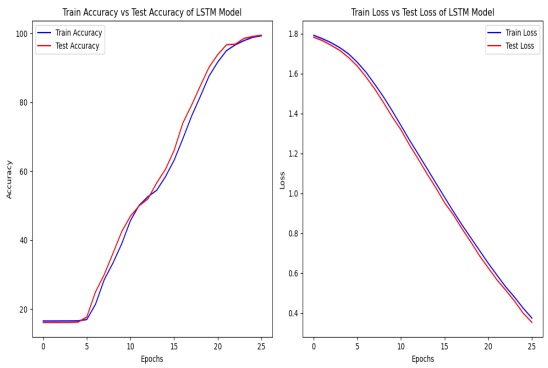

Figure 4: LSTM Model Performance

For this project, we utilized the Metal GPU provided by Apple, which offered efficient computational power for the task at hand. Since the training data points are 2D vectors and not 3D, the computational requirements were relatively low, and heavy computing resources were not necessary. The entire processing took approximately 35 minutes to complete. The training and testing accuracies reached around 99 percentage, and when the model was tested on an unknown dataset, the results were quite promising.

While the current model performed well, further optimization could be achieved by experimenting with additional data manipulation techniques and testing on a wider variety of models. These adjustments could potentially enhance the model's performance, making it more robust and capable of handling more complex or varied datasets. In future work, fine-tuning the data and exploring alternative model architectures could lead to even better results and greater generalization.

6 Error analysis

As the training dataset primarily consists of European and American voices, the model's performance is suboptimal when tested on South Asian speeches. Additionally, there is a possibility that the model might confuse emotions with language-specific features, as it could inadvertently learn to categorize based on language rather than the intended emotional cues. Consequently, the model's performance diminishes when applied to out-of-the-box speech data.

To address this issue, it would be beneficial to incorporate a more diverse and representative dataset, including voices from various regions and languages. By feeding the model with a broader range of speech data that spans different accents, dialects, and linguistic variations, we can help the model better generalize and accurately capture emotional cues across different cultural contexts. Expanding the dataset in this way would likely improve the model's robustness and its ability to handle speech data from a variety of sources, leading to more reliable performance on diverse inputs.

7 Conclusion

There were two key observations I made during the project: the importance of feature extraction using various energy density methods and the model selection process. It became evident that the CNN model performed relatively well, and

there is potential for further improvement, especially by exploring the use of Vision Transformers. Looking ahead, there are several future approaches to consider. One such approach would be to generalize the dataset by incorporating a wider variety of regional languages, which could make the training more robust and inclusive.

Another critical aspect to address is understanding exactly what the model is learning. Since the task involves speech, it's essential to ensure that the model is focusing on the emotional content rather than other aspects such as language, semantics, or other non-relevant characteristics. This understanding would be key in refining the model selection approach and ensuring that the model is correctly trained to recognize the intended features—emotion in this case—rather than being influenced by language-specific patterns.

References

Bankar, A., Gandhi, A., and Baviskar, D. (2021). Image and signal processing of mel-spectrograms in isolated speech recognition. *International Journal of Computer Applications*, 183:11–17.

Devi, J. S., Yarramalle, S., and Nandyala, S. P. (2014). Speaker emotion recognition based on speech features and classification techniques. *International Journal of Image, Graphics and Signal Processing*, 6(7):61.

Dosovitskiy, A. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, pages 1–7.

Ingale, A. B. and Chaudhari, D. S. (2012). Speech emotion recognition. *International Journal of Soft Computing and Engineering (IJSCE)*, 2(1):235–237.

Khalil, R. A., Jones, E., Babar, M. I., Jan, T., Zafar, M. H., and Alhussain, T. (2019). Speech emotion recognition using deep learning techniques: A review. *IEEE access*, 7:117327–117345.

Swain, M., Routray, A., and Kabisatpathy, P. (2018). Databases, features and classifiers for speech emotion recognition: a review. *International Journal of Speech Technology*, 21:93–120.

Zhang, B., Leitner, J., and Thornton, S. (2019). Audio recognition using mel spectrograms and convolution neural networks. *Noiselab University of California: San Diego, CA, USA*.