END TO END SPEECH EMOTION RECOGNITION

A report submitted in partial fulfillment of the

requirements Of

Mini-Project (IS65)

In

Sixth Semester

By

1MS20IS108	**Shishir N**
1MS20IS116	**Srivatsa V**
1MS20IS131	**Vashista A N**
1MS20IS135	**Vivek B V**

Under the guidance of

Dr. Sumana M
Associate Professor
Dept. of ISE, RIT

RAMAIAH

Institute of Technology

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

RAMAIAH INSTITUTE OF TECHNOLOGY

(AUTONOMOUS INSTITUTE AFFILIATED TO VTU)

M. S. RAMAIAH NAGAR, M. S. R. I. T. POST, BANGALORE – 560054

2022-2023

RAMAIAH INSTITUTE OF TECHNOLOGY
(Autonomous Institute Affiliated to VTU)
M. S. Ramaiah Nagar, M. S. R. I. T. Post, Bangalore – 560054

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

RAMAIAH INSTITUTE OF TECHNOLOGY

CERTIFICATE

This is to certify that the project work entitled **"END TO END SPEECH EMOTION RECOGNITION"** is a bonafide work carried out by **Shishir N** bearing **USN: 1MS20IS108, Srivatsa V** bearing **USN: 1MS20IS116, Vashista A N** bearing **USN: 1MS20IS131, Vivek B V** bearing **USN: 1MS20IS135** in partial fulfillment of requirements of Mini-Project (ISL65) of Sixth Semester B.E. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report. The project has been approved as it satisfies the academic requirements in respect of project work prescribed by the above said course.

_____	_____
Signature of the Guide	Signature of the HOD
Dr. Sumana M	**Dr. Sanjay H A**
Associate Professor	Professor and Head,
Dept. of ISE, RIT,	Dept. of ISE, RIT
Bangalore-54	Bangalore-54

Other Examiners

Name of the Examiners:	**Signature**
1.	
2.	

Abstract

Speech emotion recognition (SER) is a rapidly evolving field that aims to automatically identify and classify human emotions from speech signals. The ability to recognize emotions in speech is crucial for improving the performance and user experience of various applications, such as virtual assistants, call center analytics, and affective computing systems. In this project, we present an end-to-end speech emotion recognition system that combines machine learning, artificial neural networks (ANN), and natural language processing (NLP) techniques for a comprehensive and robust analysis of emotions in speech.

The proposed system starts by extracting relevant features from the audio signal, such as Mel-frequency cepstral coefficients (MFCC), which capture the spectral characteristics of the speech signal. These features are then used to train a feedforward artificial neural network (ANN) to classify the emotions in the speech. The neural network model is optimized using techniques like dropout and adaptive learning rates to prevent overfitting and improve generalization performance.

In addition to the audio features, the model also incorporates natural language processing (NLP) techniques to analyze the textual content of the speech. It also uses speech-to-text conversion to obtain the transcript of the speech, and then apply sentiment analysis using TextBlob to calculate the sentiment polarity of the text. This information is combined with the emotion predicted by the ANN model to create a more effective emotion classification, taking into account both the acoustic and linguistic aspects of the speech.

The end-to-end SER system is evaluated on a publicly available RAVDESS dataset, achieving high accuracy in emotion classification. The inclusion of NLP techniques to analyze the sentiment polarity of the text contributes to the overall performance improvement of the system, demonstrating the potential of combining audio and text-based features for robust speech emotion recognition.

INDEX

I. INTRODUCTION

Human emotions play a significant role in communication, as they convey essential information that goes beyond the mere content of spoken words. With the rapid advancements in artificial intelligence and natural language processing, there is an increasing need for systems capable of recognizing and interpreting emotions in speech. Speech Emotion Recognition (SER) is a growing interdisciplinary field that aims to develop algorithms and techniques to automatically identify and classify human emotions from spoken language. The potential applications of SER systems are vast, spanning various domains such as virtual assistants, customer service, mental health monitoring, and affective computing.

In this project, an end-to-end speech emotion recognition system that harnesses the power of machine learning, artificial neural networks (ANN), and natural language processing (NLP) to provide a comprehensive and robust analysis of emotions in speech. The primary objective of this project is to design and implement a system that effectively combines acoustic and linguistic features to improve the accuracy and reliability of emotion classification in speech.

To achieve this goal, the model begins by extracting relevant acoustic features from the audio signals, such as Mel-frequency cepstral coefficients (MFCC), which effectively capture the spectral characteristics of speech. These features serve as the input for a feedforward artificial neural network (ANN) trained to classify emotions in speech. To enhance the model's performance, employment of optimization techniques such as dropout and adaptive learning rates to mitigate overfitting and promote generalization.

In addition to acoustic features, our SER system incorporates natural language processing (NLP) techniques to analyze the textual content of speech. It utilizes speech-to-text conversion to obtain the transcripts of spoken words, followed by sentiment analysis using TextBlob to compute sentiment polarity scores. This textual information is integrated with the ANN model's predictions to derive a more effective emotion classification that accounts for both acoustic and linguistic aspects of speech.

The end-to-end speech emotion recognition system is thoroughly evaluated on a well-known benchmark dataset, the RAVDESS dataset, to assess its performance in classifying emotions accurately. The incorporation of NLP techniques for sentiment analysis demonstrates a notable improvement in the overall performance of the system, underscoring the benefits of combining audio and text-based features for robust and accurate speech emotion recognition.

In conclusion, the end-to-end speech emotion recognition system leverages machine learning, artificial neural networks, and natural language processing for accurate emotion classification in speech. By combining acoustic and linguistic features, the system achieves improved performance. With applications in virtual assistants, customer service, mental health monitoring, and affective computing, this versatile system enables more empathetic human-machine interactions. Future work may explore additional features and techniques to further enhance performance and investigate new applications.

II. OBJECTIVES

1. The ANN model is trained from the labeled data using the RAVDESS audio dataset.
2. Emotion of the unlabeled data is predicted using extracted features from the mfcc.
3. The speech in the audio file is converted to text and the polarity of the text is calculated.
4. The combined effective emotion of the entire audio file is estimated and displayed.

DATASET DESCRIPTION:

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) is a popular dataset for speech emotion recognition. It contains audio and video recordings of actors performing scripted speech and songs with different emotions. Here is a detailed description of the RAVDESS dataset**:**

Audio-visual samples:

- 24 professional actors (12 male, 12 female) participated in the recording sessions.
- Each actor performed two scripted statements (a neutral statement and a statement with emotional content) with a total of 8 emotions: neutral, calm, happy, sad, angry, fearful, disgust, and surprised.
- Actors also performed the same emotional expressions with a strong intensity level.
- Speech and song modalities are available, with a total of 7356 recordings.
- All audio files are in WAV format, with a 48 kHz sampling rate and 24-bit depth.
- Video files are in MP4 format, with a 30 fps frame rate and 720p resolution.
- The duration of each audio sample varies between 3 and 5 seconds.

File naming convention:

- The file names follow a specific pattern: Modality-Emotion-Intensity-Actor-Statement-Take
- Modality: 01 for speech, 02 for song
- Emotion: 01 = neutral, 02 = calm, 03 = happy, 04 = sad, 05 = angry, 06 = fearful, 07 = disgust, 08 = surprised
- Intensity: 01 = normal, 02 = strong
- Actor: 01 to 24, with even numbers representing male actors and odd numbers representing female actors
- Statement: 01 = statement 1, 02 = statement 2
- Take: 01 or 02 (each statement was performed twice)

III. METHODOLOGY

- Data Collection and Preprocessing:

  Collect a large dataset of speech samples with labeled emotions (RAVDESS, or TESS datasets). Convert the audio files to a suitable format (e.g., WAV) and Perform voice activity detection (VAD) to remove non-speech segments and reduce computational complexity.

- Feature Extraction:

  Extract acoustic features from the speech signals using open-source libraries like librosa. Common features include Mel-frequency cepstral coefficients (MFCCs), chroma features, spectral contrast, and pitch. For NLP-based analysis, use an automatic speech recognition (ASR) system like Google Speech-to-Text to convert the speech signals into text transcripts.

- Text Preprocessing and Sentiment Analysis:

  Use an NLP library like NLTK to perform sentiment analysis on the text data and compute polarity scores.

- Model Development:

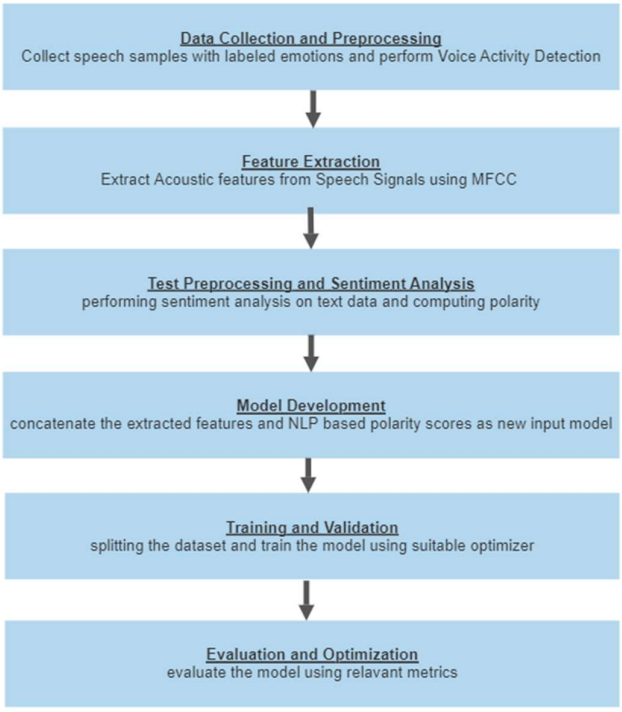
  For the acoustic model, use a deep learning architecture like feed forward neural networks.
  Concatenate the extracted acoustic features and NLP-based polarity scores as input to a new model, which can be another deep learning architecture.

- Training and Validation:

  Split the dataset into training and testing sets (our case 80% and 20%).
  Train the models on the training set using a suitable optimizer (e.g., Adam).

- Evaluation and Optimization:

  Evaluate the trained models on the test set using relevant metrics such as accuracy, F1-score, and confusion matrix.



III. LITERATURE SURVEY

In [1], This project aims to classify speech into four emotions: sadness, anger, fear, and happiness. Speech samples from LDC and UGA databases were used, and important characteristics such as energy, pitch, MFCC and LPCC coefficients, and speaker rate were extracted. Support Vector Machine (SVM) was used as the classifier with two strategies: One against All (OAA) and Gender Dependent Classification. The study also conducted a comparative analysis between the two strategies and two algorithms, LPCC and MFCC

In [2], The paper "End-to-end speech emotion recognition using a novel context-stacking dilated convolution neural network" proposes a novel approach for speech emotion recognition based on an end-to-end architecture that employs a context-stacking dilated convolutional neural network (CS-DCNN). The proposed model is capable of learning the emotional features directly from the speech signal without any intermediate feature extraction steps.

In [3], The paper "Attention and Feature Selection for Automatic Speech Emotion Recognition Using Utterance and Syllable-Level Prosodic Features" by Starlet Ben Alex, Leena Mary, and Ben P. Babu proposes a method for automatic speech emotion recognition using a combination of utterance and syllable-level prosodic features, attention mechanisms, and feature selection techniques.

In [4], It proposes a new end-to-end approach for speech emotion recognition that incorporates an acoustic-to-word automatic speech recognition (ASR) model. It consists of two stages: a feature extraction stage and a classification stage. In the feature extraction stage, the speech signal is transformed into a sequence of word-level embeddings using the acoustic-to-word ASR model. The word-level embeddings are then used as input to a convolutional neural network (CNN) to extract high-level features that are relevant for emotion recognition.

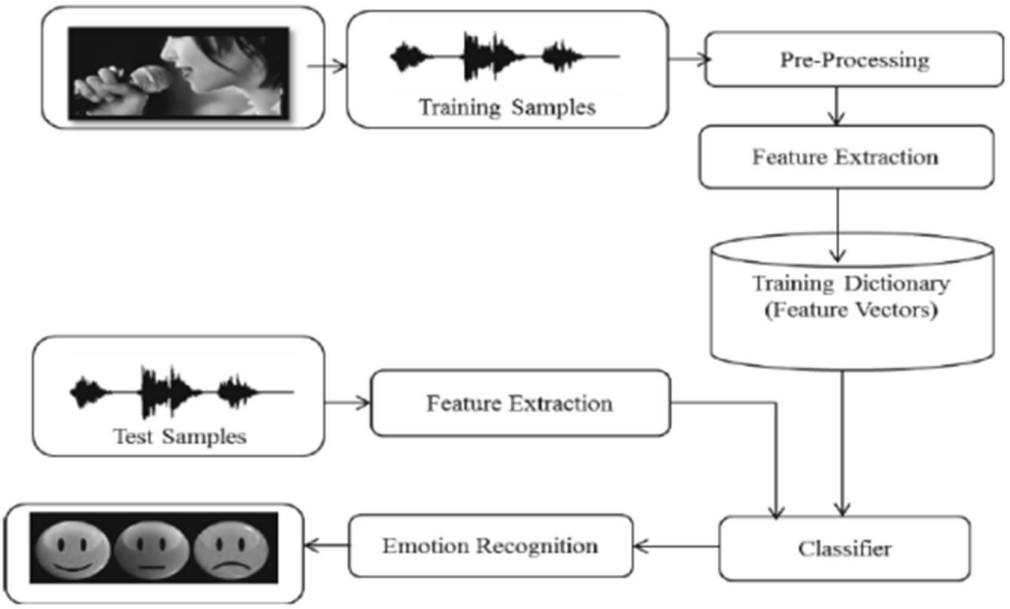
In [5], The paper proposes a speech emotion recognition system that considers both verbal and nonverbal speech sounds. The system is based on a deep neural network that takes as input both speech signals and facial expressions, which are extracted using a deep convolutional neural network.

In [6], It is an improved end-to-end speech emotion recognition system that uses a self-attention mechanism and multitask learning to enhance the accuracy and robustness of the model. The system consists of a feature extraction module that extracts both spectral and prosodic features from speech signals, a self-attention module that learns the importance of each feature for emotion recognition, and a multitask learning module that jointly learns to predict emotion labels and speaker identities

In [7], The paper provides a detailed explanation of deep learning techniques such as convolutional neural networks (CNN), recurrent neural networks (RNN), long short-term memory (LSTM), and gated recurrent units (GRU) and their applications in speech emotion recognition. The authors also discuss attention mechanisms, which allow the models to focus on the most relevant parts of the input signal.

In [8], The paper "Speech emotion recognition using recurrent neural networks with directional self-attention" proposes a novel approach for speech emotion recognition using a directional self-attention mechanism incorporated into a recurrent neural network (RNN). The directional self-attention mechanism helps the model to focus on the most relevant parts of the speech signal while ignoring irrelevant information.

SYSTEM ARCHITECTURE:



Data Preprocessing:

a. Audio Preprocessing:
Load the dataset audio files (e.g., RAVDESS dataset in WAV format).
Perform noise reduction or filtering, if necessary.

b. Feature Extraction:
Extract relevant features from the audio samples, such as Mel-frequency cepstral coefficients, chroma features, spectral contrast, etc.

Data Splitting:

Divide the dataset into training, validation, and testing sets.

Model Development:

Design an appropriate ANN architecture, such as a combination of convolutional layers,recurrent layers (LSTM or GRU), and dense layers, or use a pre-trained model with transfer learning.

Model Training:

Train the ANN model using the training set and adjust hyperparameters based on the validation set performance.

Model Evaluation:

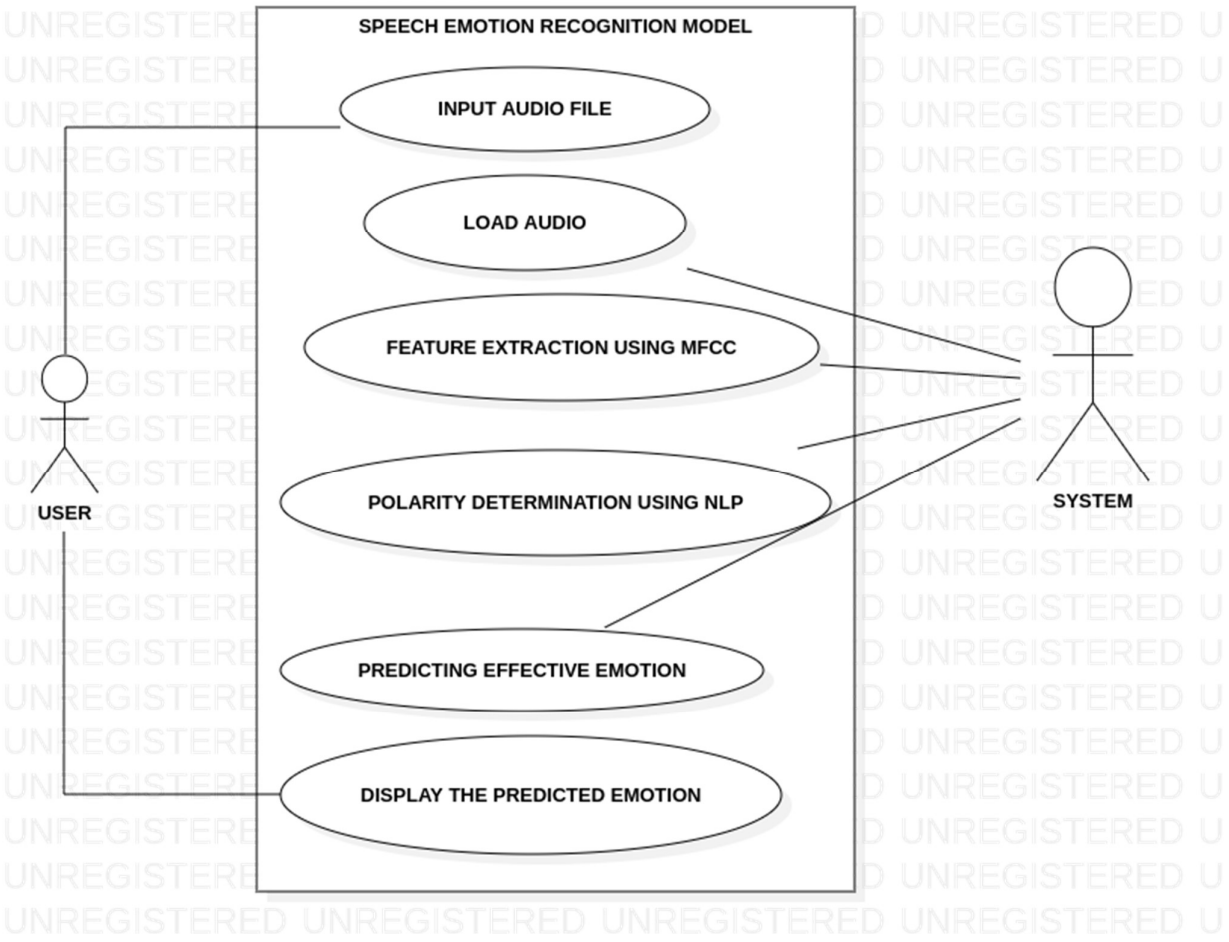
Test the model's performance using the testing set, and analyze metrics like accuracy, F1-score, and confusion matrix.

NLP for Context:

a. Speech-to-Text:
Convert the audio samples into text transcripts using a speech-to-text service or library.

b. Sentiment Analysis:
Perform sentiment analysis on the transcripts using NLP techniques, such as TextBlob, BERT.

USE CASE DIAGRAM:



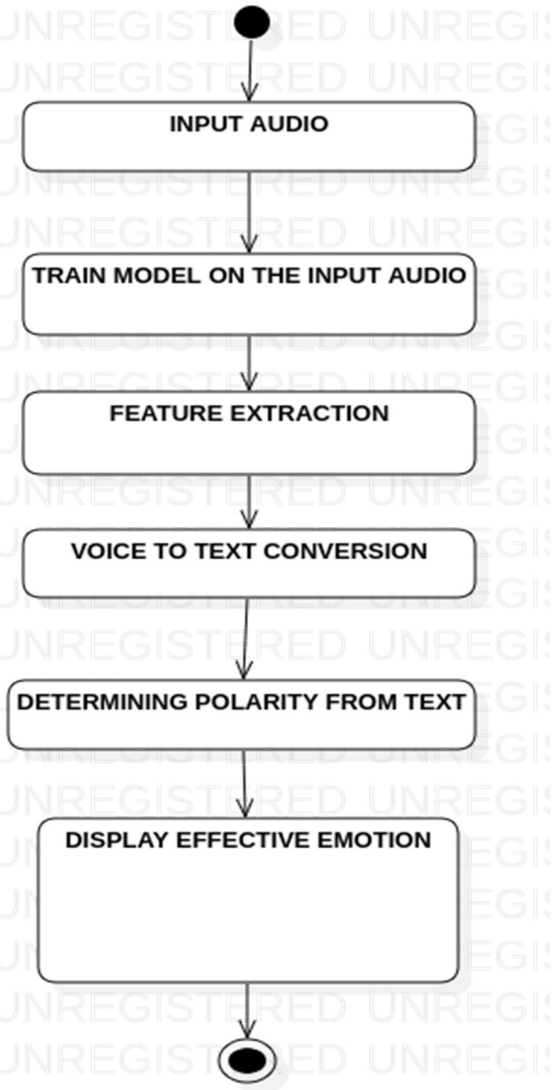
Actors:

- User: A person or another system that interacts with the speech emotion recognition system to analyze emotions in speech.
- System: The speech emotion recognition system.

Use Cases:

- Input Audio File: The user input audio data, including noise.
- Load Audio: The system loads the audio from the user and filters out the noise during preprocessing.
- Feature Extraction Using MFCC: The system extracts the 39 main features of the audio using MFCC (13 of energy features, 13 delta features, 13 double delta features).
- Polarity determination using NLP: The system converts audio samples to text using a speech-to-text API or a pre-trained model and calculates the polarity.
- Predict Effective Emotion: The system performs sentiment analysis on the transcriptions using NLP techniques to get context and the ANN model determines the emotion according to the training and the combined effective emotion is calculated by the system.
- Display The Predicted Emotion: The system displays the predicted effective emotion to the user.

ACTIVITY DIAGRAM:



For the speech emotion recognition system, a sequence diagram can be created to illustrate the order of processes and interactions between components. Here's a description of a sequence diagram for analyzing emotions in a given speech input:

- End-User sends an audio input to the Speech Emotion Recognition System.
- The system preprocesses the audio input, including noise reduction, filtering, and feature extraction.
- The preprocessed audio is passed to the trained ANN model for emotion recognition.
- The ANN model predicts the emotion based on the acoustic features.
- The audio input is converted to text using a speech-to-text API or a pre-trained model.
- The text is passed to the sentiment analysis component, which analyzes the sentiment using NLP techniques.
- Sentiment analysis results are returned.
- The emotion recognition results from the ANN model and the sentiment analysis are integrated to generate a final emotion prediction.
- The final emotion prediction is returned to the End-User.

IV. PROPOSED SOLUTION

The proposed solution for the end-to-end speech emotion recognition project using ML, ANN, and NLP consists of several stages, including preprocessing, feature extraction, feature fusion, and classification. Here's an outline of the model:

1. Preprocessing:
   Load the speech data and convert it into a suitable format. Perform noise reduction and normalization to improve the quality of the audio signal. Apply windowing and frame segmentation to divide the audio signal into overlapping frames.

2. Feature Extraction:
   - Acoustic Features: Extract relevant acoustic features, such as Mel-frequency cepstral coefficients (MFCCs), pitch, and energy, which are known to be effective for emotion recognition.
   - Linguistic Features: Convert speech to text using speech recognition techniques (e.g., Google Speech-to-Text API). Then, apply NLP techniques like sentiment analysis (e.g., using TextBlob) to obtain sentiment polarity scores and other linguistic features.

3. Feature Fusion:
   Combine the extracted acoustic and linguistic features into a single feature vector, capturing complementary information from both sources.

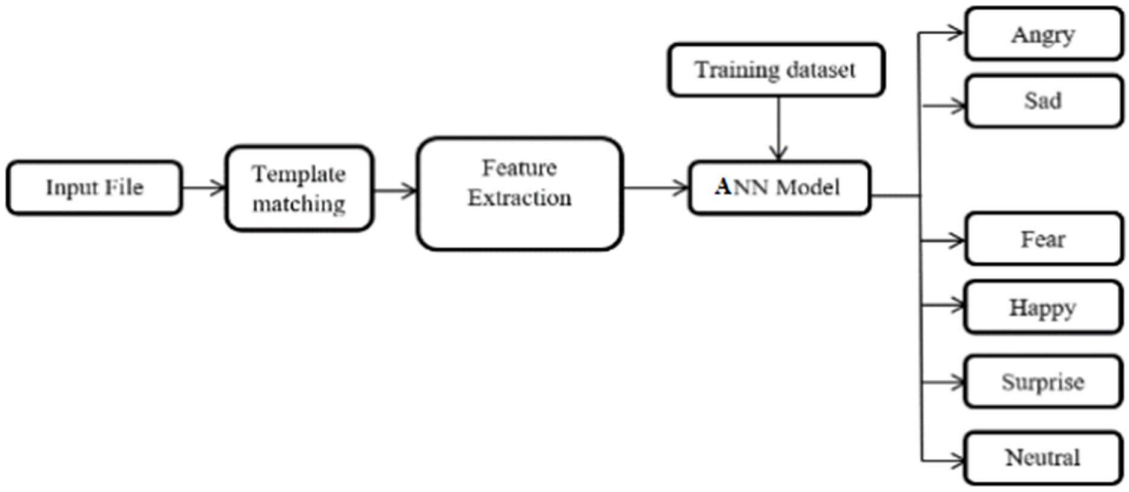
4. Classification:
   Train an ANN model (e.g., a feed-forward neural network) using the fused feature vectors and corresponding emotion labels. Optimize the model by fine-tuning hyperparameters, such as the number of hidden layers, neurons, and learning rate.

5. Evaluation:
   Evaluate the model's performance on a test dataset by calculating accuracy, precision, recall, and F1 score. Analyze the results and compare them with existing approaches to assess the effectiveness of the proposed model.

6. Post-processing (optional):
   Implement a decision-making mechanism that considers both the ANN's output and the sentiment polarity to determine the final emotion label, potentially improving the overall performance.

The proposed solution aims to overcome the challenges associated with speech emotion recognition by integrating acoustic and linguistic features and leveraging the power of ANN for classification. By incorporating NLP techniques and feature fusion, the model has the potential to achieve high accuracy and robustness in recognizing emotions across various speakers and real-world conditions.

ALGORITHM:



Feature Extraction:
Extract relevant acoustic features from the audio samples, such as Mel-frequency cepstral coefficients (MFCCs).

ANN Model Development:
Design a feed-forward neural network with multiple layers, including input, hidden, and output layers. Determine the number of neurons in each layer and the activation functions (ReLU, sigmoid, softmax). Define the loss function (categorical cross-entropy) and optimization algorithm (Adam).

Model Training and Evaluation:
Train the ANN model using the training set, adjusting hyperparameters based on the validation set performance. Implement techniques like dropout and early stopping to prevent overfitting and improve generalization.

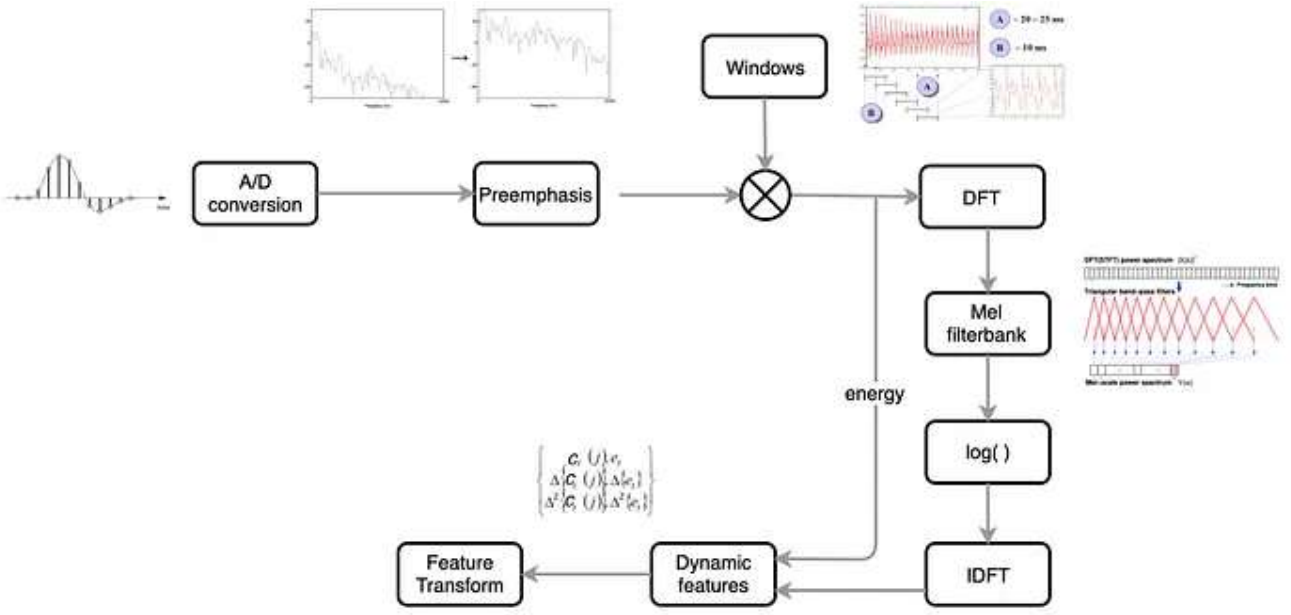
Model Evaluation:
Test the model's performance using the testing set, and analyze metrics like accuracy, F1-score, and confusion matrix.

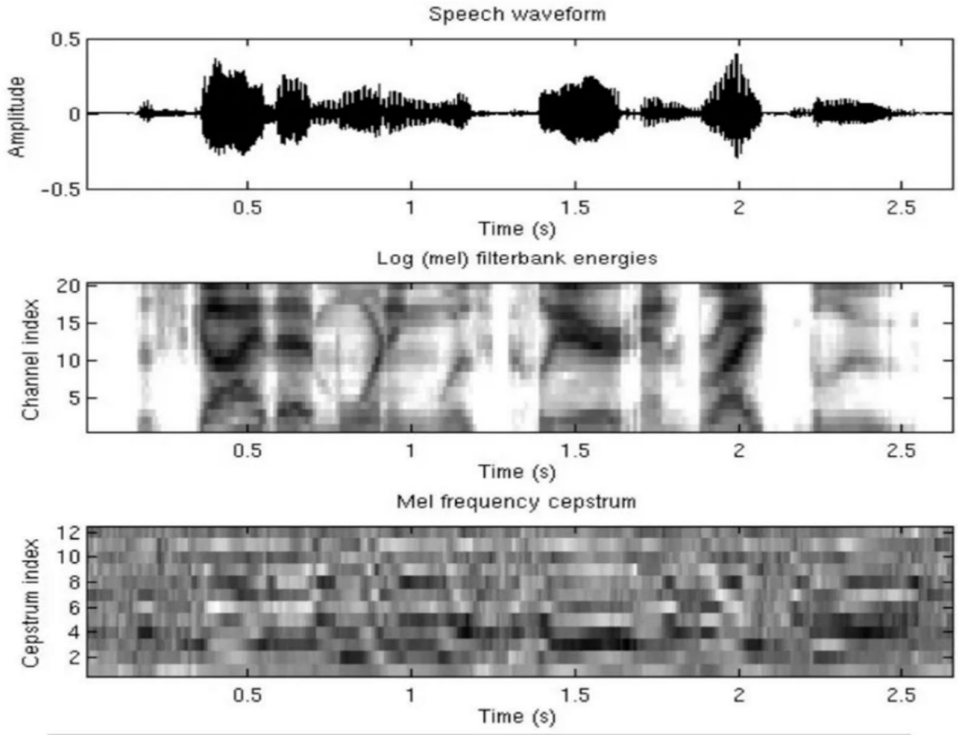
Speech-to-Text Conversion:
Convert the audio samples to text using a speech-to-text API or a pre-trained model (Google Speech-to-Text API).

Sentiment Analysis:
Perform sentiment analysis on the transcriptions using NLP techniques, such as TextBlob or pre-trained sentiment analysis models.

Working of MFCC:

1. Pre-emphasis: This filter increases the amplitude of high-frequency components and reduces the amplitude of low-frequency components, which helps emphasize the high-frequency information and balance the frequency range.
2. Framing: The pre-emphasized signal is divided into short overlapping frames, typically around 20-30 ms in length, with an overlap of 50% or more.
3. Windowing: Each frame is multiplied by a window function (e.g., Hamming window) to reduce the discontinuity at the edges of the frame.
4. Fast Fourier Transform (FFT): The windowed frame is transformed into the frequency domain using FFT, which allows the analysis of the frequency content of each frame.
5. Mel Filter Bank: A Mel filter bank consisting of a set of triangular filters is applied to the FFT spectrum to obtain the Mel-scaled spectrum. This step helps to focus on the most relevant frequency components for speech analysis.
6. Logarithm: The log of the Mel-scaled spectrum is calculated to mimic the human perception of loudness, which is logarithmic in nature.
7. Discrete Cosine Transform (DCT): The log Mel-scaled spectrum is transformed using DCT to obtain the MFCC features.
8. Feature Selection: Typically, the first few MFCCs are selected, as they carry the most relevant information about the spectral shape. Additionally, delta and delta-delta coefficients (first and second-order derivatives) can be computed and appended to the MFCCs to capture the dynamic information in the speech signal.





Source

Windowing involves the slicing of the audio waveform into sliding frames. But it cannot just chop it off at the edge of the frame. The suddenly fallen in amplitude will create a lot of noise that shows up in the high-frequency. To slice the audio, the amplitude should gradually drop off near the edge of a frame.

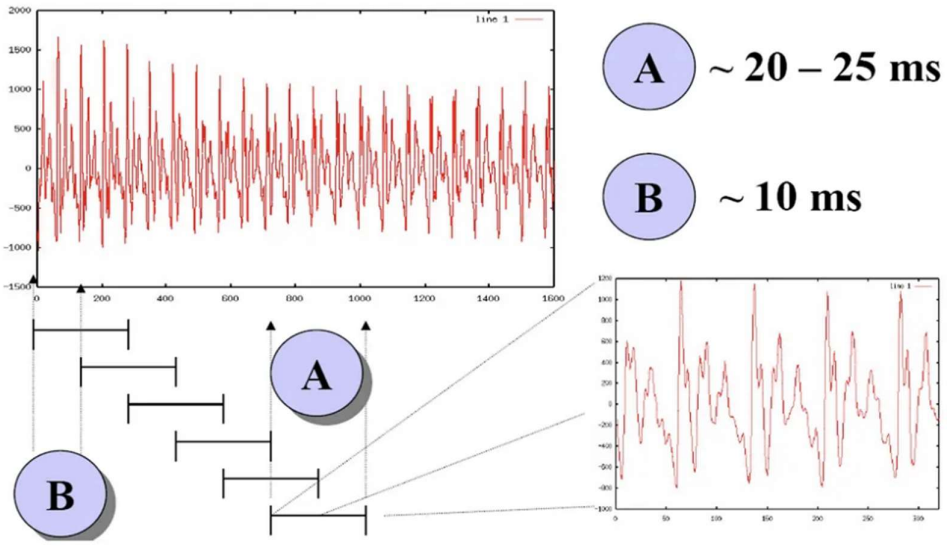
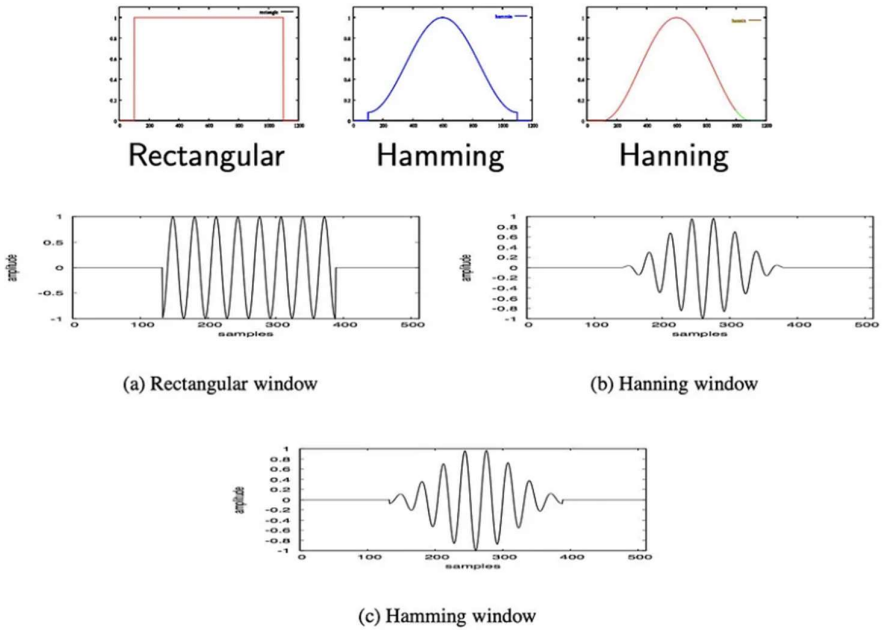


Image from Bryan Pellom

A few alternatives for w are the Hamming window and the Hanning window. The following diagram indicates how a sinusoidal waveform will be chopped off using these windows. As shown, for Hamming and Hanning window, the amplitude drops off near the edge. (The Hamming window has a slight sudden drop at the edge while the Hanning window does not.)



(a) Rectangular window

(b) Hanning window

(c) Hamming window

SOME OF THE MAJOR EQUATIONS USED ARE:

$$x[n] = w[n]\,s[n]$$

sliced frame — original audio clip

Hamming ($\alpha = 0.46164$) or *Hanning* ($\alpha = 0.5$) window

$$w[n] = (1-\alpha) - \alpha\cos\left(\frac{2\pi n}{L-1}\right) \qquad L: \text{window width}$$

DCT (DISCRETE COSINE TRANSFORM):

$$X[k] = \sum_{n=0}^{N-1} x[n]\exp\left(-j\frac{2\pi}{N}kn\right)$$

IDFT (INVERSE DISCRETE FOURIER TRANSFROM):

$$y_t[n] = \sum_{m=0}^{M-1} \log(Y_t[m])\cos\left(n(m+0.5)\frac{\pi}{M}\right)$$

V. RESULTS AND DISCUSSION

```python
# Set the path to the RAVDESS dataset
data_path = "C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_features"

# Define function to extract emotion, gender, and file path for each audio file
def load_data(data_path):
    emotion_labels = {
        '01': 'neutral',
        '02': 'calm',
        '03': 'happy',
        '04': 'sad',
        '05': 'angry',
        '06': 'fearful',
        '07': 'disgust',
        '08': 'surprised'
    }

    data = []
    for subdir, dirs, files in os.walk(data_path):
        for file in files:
            if file.endswith(".wav"):
                file_path = os.path.join(subdir, file)
                file_name = os.path.basename(file_path).split(".")[0]
                emotion = emotion_labels[file_name.split("-")[2]]
                data.append((file_path, emotion))
    return data

data = load_data(data_path)
```

Python

The above snap shows the labeling and mapping of emotions, the path of the dataset is described and it being loaded to the model.

```python
model = Sequential()
model.add(Dense(256, activation='relu', input_shape=(X_train.shape[1],)))
model.add(Dropout(0.5))
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(8, activation='softmax'))

model.compile(loss='categorical_crossentropy', optimizer=Adam(lr=0.001), metrics=['accuracy'])

history = model.fit(X_train, y_train, batch_size=32, epochs=100, validation_split=0.1)
```

Python

Here the training of the ANN model using Feed Forward Neural Networks starting with 256 layers and ReLU activation function and a dropout to prevent overfitting, down to 8 layers because there are 8 emotion labels.

```python
y_pred = model.predict(X_test)
y_pred_classes = np.argmax(y_pred, axis=1)
y_test_classes = np.argmax(y_test_encoded, axis=1)

# Calculate accuracy
accuracy = accuracy_score(y_test_classes, y_pred_classes)
print(f"Accuracy: {accuracy}")

# Calculate confusion matrix
conf_matrix = confusion_matrix(y_test_classes, y_pred_classes)
print(f"Confusion Matrix:\n{conf_matrix}")

# Create a function to predict the emotion from an audio file
def predict_emotion(file_path, model, scaler):
    features = librosa.feature.mfcc(y=librosa.load(file_path)[0], n_mfcc=40)
    features_scaled = scaler.transform(np.mean(features, axis=1).reshape(1, -1))
    prediction = model.predict(features_scaled)
    emotion_labels = ['neutral', 'calm', 'happy', 'sad', 'angry', 'fearful', 'disgust', 'surprised']
    return emotion_labels[np.argmax(prediction)]

# Test the prediction function
file_path = "C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_features/Actor_11/0
print(f"Predicted emotion: {predict_emotion(file_path, model, scaler)}")
```

Python

```
51/51 [==============================] - 0s 760us/step
Accuracy: 0.91495965238982
Confusion Matrix:
[[227   1   2   1   1   2   0   4]
 [  0  67   0   1   2   1   0   0]
 [  3   4 175   0   0   0   1   5]
 [  6   0   2 209   3   0  13   2]
 [  3   8   1   6 220   0   2   3]
 [  0  20   0   0   0 176   1   2]
 [  0  11   0   7   0   1 209   4]
 [  2   1   5   1   3   1   1 191]]
1/1 [==============================] - 0s 17ms/step
Predicted emotion: calm
```

The model in use works best for the metric ‘Accuracy_Score’ as the Feed Forward Neural Network model depends on the emotion prediction more than the polarity calculation. But the polarity calculation gives a bit more information regarding the overall context of the conversation.

```
Accuracy: 0.91495965238982
Precision: 0.9242079475915034
Recall: 0.91495965238982
F1 Score: 0.917382055056903
```

```
Matthews Correlation Coefficient: 0.9025941523812553
```

The MCC value is the overall optimal performance metric as it gives the best results.

```python
def speech_to_text(file_path):
    recognizer = sr.Recognizer()
    with sr.AudioFile(file_path) as source:
        audio_data = recognizer.record(source)
        try:
            text = recognizer.recognize_google(audio_data)
            return text
        except sr.UnknownValueError:
            return "Unrecognized speech"
        except sr.RequestError as e:
            return f"Could not request results: {e}"

def sentiment_polarity(text):
    sentiment = TextBlob(text).sentiment
    return sentiment.polarity

def effective_emotion(emotion, polarity):
    if polarity > 0.2:
        return "positive"
    elif polarity < -0.2:
        return "negative"
    else:
        return emotion

data = load_data('C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_features')

for file_path, emotion in data:
    text = speech_to_text(file_path)
    polarity = sentiment_polarity(text)
    eff_emotion = effective_emotion(emotion, polarity)
    print(f"Audio file: {file_path}\nEmotion: {emotion}\nText: {text}\nPolarity: {polarity}\nEffective emotion: {eff
```

Python

```
ile: C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_featu
: neutral
Shadow bird live
y: 0.13636363636363635
ive emotion: neutral
```

```
Audio file: C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_featu
Emotion: neutral
Text: Tonk first sitting by the door
Polarity: 0.25
Effective emotion: positive
```

```
Audio file: C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_featu
Emotion: happy
Text: casual talking by the door
Polarity: -0.5000000000000001
Effective emotion: negative
```

```
Audio file: C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_featur
Emotion: neutral
Text: say the word hate
Polarity: -0.8
Effective emotion: negative
```

```
Audio file: C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_featu
Emotion: neutral
Text: say the word large
Polarity: 0.21428571428571427
Effective emotion: positive
```

```
Audio file: C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_featur
Emotion: fearful
Text: the word tight
Polarity: -0.17857142857142858
Effective emotion: fearful
```

```
Audio file: C:/Users/Vivek/Desktop/6th_sem_miniproj_full_code/full_code_speech_emotion/dataset_featur
Emotion: sad
Text: the word hate
Polarity: -0.8
Effective emotion: negative
```

VI. CONCLUSION AND FUTURE SCOPE

Conclusion:

In conclusion, end-to-end speech emotion recognition systems that synergistically leverage machine learning, artificial neural networks, and natural language processing have made remarkable strides in the domains of human-computer interaction and affective computing. The seamless integration of these cutting-edge technologies has resulted in a more accurate and reliable identification of emotions from speech signals, paving the way for a myriad of practical applications.

Such applications encompass mental health monitoring, where these systems can aid in detecting early signs of emotional distress, thus facilitating timely intervention and support. In the customer service industry, these systems can be employed to gauge customer sentiment and automatically route calls to appropriate agents or escalate issues, leading to improved customer satisfaction. Furthermore, the enhancement of human-computer interfaces through emotion-aware systems can foster more natural and empathetic interactions, contributing to a more immersive user experience.

Overall, the ongoing advancements in end-to-end speech emotion recognition systems have the potential to revolutionize various industries and lead to innovative solutions that address complex challenges. As these systems continue to evolve, they will unlock new opportunities for fostering more effective communication and empathetic understanding between humans and machines, ultimately transforming the way users interact with technology.

Future Scope:

Multimodal emotion recognition: Integrating other modalities, such as facial expressions, body language, and physiological signals, could significantly improve the performance of emotion recognition systems. Combining these modalities would provide a more comprehensive understanding of human emotions and lead to more robust recognition systems.

Cross-cultural and cross-linguistic studies: Extensive research should be conducted to understand the variations in emotional expressions across different cultures and languages. This will enable the development of culturally-sensitive emotion recognition systems that can cater to a diverse global population.

Real-world applications: With the improvement of emotion recognition systems, more practical applications will emerge in various fields, including healthcare, education, marketing, entertainment, and social robotics. This will open up new avenues for research and development, as well as create opportunities for technology transfer and commercialization.

VII. REFERENCES

1. Manas Jain, Shruthi Narayan, Pratibha Balaji, Bharath K P, Abhijit Bhowmick, Karthik R, Rajesh Kumar Muthu (2020). Speech Emotion Recognition using Support Vector Machine. Journal of Imaging, 7(9), 165.

2. Duowei Tang,Peter Kuppens,Luc Geurts and Toon van Waterschoot. (2021).End-to-end speech emotion recognition using a novel context-stacking dilated convolution neural network. Intelligent Automation & Soft Computing, 36(2).

3. Starlet Ben Alex, Leena Mary, Ben P. Babu (2020, May).Attention and Feature Selection for Automatic Speech Emotion Recognition Using Utterance and Syllable-Level Prosodic Features. In 2018 IEEE Canadian conference on electrical & computer engineering (CCECE) (pp. 1-4). IEEE.

4. Han Feng, Sei Ueno, Tatsuya Kawahara, 2020, End-to-End Speech Emotion Recognition Combined with Acoustic-to-Word ASR-Model. INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 09, Issue 06 (June 2020).

5. Kun-Yi Huang, Chung-Hsien Wu, Qian-Bei Hong, Ming-Hsiang Su and Yi-Hsuan Chen(2019, June). Speech emotion recognition using deep neural networks considering verbal and non-verbal speech sounds. In 2019 IEEE international advance computing conference (IACC) (pp. 617-622). IEEE.

6. Yuanchao Li, Tianyu Zhao, Tatsuya Kawahara. (2019).Improved End-to-End Speech Emotion Recognition Using Self Attention Mechanism and Multitask Learning. In Real-Time Image Processing and Deep Learning 2020 (Vol. 11401, pp. 13-22). SPIE.

7. Eva Lieskovská , Maroš Jakubec, Roman Jarina and Michal Chmulík. (2021). A Review on Speech Emotion Recognition Using Deep Learning and Attention Mechanism. In 2016 2nd international conference on next generation computing technologies (NGCT) (pp. 347-351). IEEE.

8. Dongdong Li, Jinlin Liu, Zhuo Yang, Linyu Sun, Zhe Wang. (2021).Speech emotion recognition using recurrent neural networks with directional self-attention. ICTACT Journal on soft computing, 3(4), 563-575.