Enhancing Emotional Intelligence in LLM Responses Using Speech Emotion Recognition

Juanita Cathy J
Department of Computer Science and Engineering
Jeppiaar Engineering College
Chennai, India
juanitacathy486@gmail.com

Suraj Pradeepkumar
Department of Information Technology
Jeppiaar Engineering College
Chennai, India
suraj12141509@gmail.com

Abstract

This research presents a hybrid deep learning approach for Speech Emotion Recognition integrated with a LLM to enhance AI-generated responses with emotional awareness. The SER system employs a CNN and BiLSTM network for emotion classification. Mel-Frequency Cepstral Coefficients are extracted from speech signals, and the model is trained on four benchmark datasets—CREMA-D, RAVDESS, SAVEE, and TESS—comprising a total of 12,162 labeled speech samples across seven emotion categories: anger, happiness, sadness, fear, disgust, surprise, and neutral. The model achieved a training accuracy of 99.7% with validation accuracy of 84.7%. Compared to traditional machine learning classifiers like SVM and Random Forest, which achieve average accuracies of 78.57% and 71%, respectively, our model demonstrates a significant improvement. By incorporating the detected emotions into an LLM, the system enables dynamic response modulation based on user sentiment. The system processes speech inputs with an average inference time of 48ms per sample making it suitable for real-time applications. The results highlight the effectiveness of combining deep learning-based SER with LLMs to enhance emotionally intelligent human-computer interactions.

Introduction

Human communication is deeply influenced by emotions which shapes the way we interact and express thoughts. Traditional text-based emotion detection often fails to capture the full depth of emotions, as it lacks vocal cues such as pitch, tone, and rhythm. The nuances of human communication which are important for understanding emotions are not captured resulting in AI responses that lack empathy and contextual awareness. Speech Emotion Recognition addresses this gap by analyzing these acoustic features to determine the speaker's emotional state. SER captures the subtle emotional cues embedded in speech.

With the growing adoption of AI-driven conversational agents, integrating SER into Large Language Models can significantly enhance user interactions. While LLMs generate contextually relevant responses, they typically lack emotional awareness,

often leading to neutral or inappropriate replies. This could be an issue especially when dealing with sensitive applications where emotional nuances are important. SER also finds its use in various domains like healthcare and customer service and increases workflow efficiency. By incorporating SER into LLMs, AI systems can adapt and tune the responses to the user's emotional state, making conversations more natural.

This research focuses on developing a hybrid deep learning model that combines Convolutional Neural Networks and Bidirectional Long Short-Term Memory networks for emotion classification. The detected emotions are used to dynamically modify LLM responses to ensure empathetic and context-aware responses. We also use an adaptive conversational framework to track emotional progression over multiple interactions.

Related Work

2.1 Traditional and Deep Learning Approaches to Speech Emotion Recognition

Early SER models relied on handcrafted feature extraction methods such as Mel-Frequency Cepstral Coefficients, prosodic features, and spectral features, followed by machine learning classifiers like Support Vector Machines and Hidden Markov Models [1]. However, these approaches struggled with generalization due to speaker variability and dataset differences.

Deep learning methods, particularly CNNs and RNNs, have significantly improved SER performance by automatically learning discriminative features from raw audio signals [2]. Hybrid architectures, such as CNN-BiLSTM models, further enhance SER by capturing both spatial and temporal dependencies in speech signals [3]. More recently, transformer-based models like wav2vec 2.0 have demonstrated superior performance in SER by leveraging self-supervised learning on large speech datasets [4].

Cross-corpus SER remains a challenge due to variations in emotional expression across datasets. Huijuan et al. [5] proposed a deep local domain adaptation technique to improve SER performance across multiple datasets. Additionally, multimodal approaches that integrate speech with text and visual data have been explored to enhance emotion recognition accuracy [6].

2.2 Text-Based Emotion Recognition with LLMs

Recent advances in NLP have enabled LLMs to detect emotions from textual data, enhancing their response generation capabilities. Studies such as Ravi et al. [7] have demonstrated how LLMs can infer emotions based on syntactic and semantic

contexts. However, these methods are inherently limited as they do not capture paralinguistic cues such as tone, pitch, and pace—key indicators of emotional state in speech.

Our work builds upon these text-based approaches by integrating SER, allowing LLMs to consider both textual and speech-based emotional cues. This hybrid approach enhances the emotional intelligence of LLM-generated responses, making them more nuanced and contextually aware.

2.3 Multimodal SER for Speech-Text Emotion Recognition

Multimodal emotion recognition, which combines speech and text inputs, has gained traction in recent years. Davis et al. [8] introduced a speech-to-text emotion detection system that utilized both acoustic features and textual transcriptions for emotion analysis. While effective in improving emotion detection, these models did not explore how detected emotions could enhance AI-generated responses.

Similarly, Vasilenko et al. [9] focused on improving multimodal emotion detection but did not address response generation. Our work advances this field by not only detecting emotions from speech but also dynamically incorporating them into LLM responses. This enables a more empathetic and emotionally responsive AI system.

2.4 Enhancing Conversational AI with Emotion-Aware LLMs

Several studies have investigated the integration of SER with conversational AI. Xu et al. [10] explored the use of acoustic features to fine-tune LLM responses, enabling virtual assistants to respond to users with improved emotional awareness. However, their approach relied on additional feature extraction models, increasing computational complexity. In contrast, our system directly translates speech emotion features into natural language descriptors that guide LLM responses. This simplifies the architecture while maintaining high adaptability across various conversational AI applications.

2.5 Existing Text-Based Emotion Detection Systems

Traditional text-based emotion recognition systems in LLMs, such as those discussed by Brown et al. [11], have seen significant improvements but fail to incorporate paralinguistic speech information. Gong et al. [12] demonstrated that contextual emotional cues from speech could enhance sentiment analysis and conversation modeling. However, these systems primarily focus on classifying emotions rather than improving the emotional intelligence of AI-generated responses.

Our work builds on these advancements by integrating CNN-BiLSTM-based SER with LLM to generate emotion-aware responses that enhance the LLM's emotional intelligence. Unlike previous studies that focus solely on classifying emotions, our approach dynamically adjusts conversational responses based on detected emotions. This makes the model's responses more human-like and emotionally sensitive, distinguishing it from traditional systems that only focus on text-based cues.

Methodology

Speech-based emotion detection captures vocal nuances, allowing for a more accurate interpretation of the speaker's emotional state. Our methodology consists of three core components: speech emotion recognition, emotion-aware LLM response generation, and an adaptive conversational framework.

Speech Emotion Recognition Pipeline

SER enables machines to recognize and interpret human emotions by analyzing acoustic features extracted from raw audio. We begin by capturing speech input, followed by feature extraction, where key characteristics such as pitch, energy, and spectral properties are identified. These extracted features are then processed using a deep learning model to classify the emotional state of the speaker. The detected emotion is then incorporated into LLM responses, allowing the system to adapt its tone, word choice, and response structure accordingly.

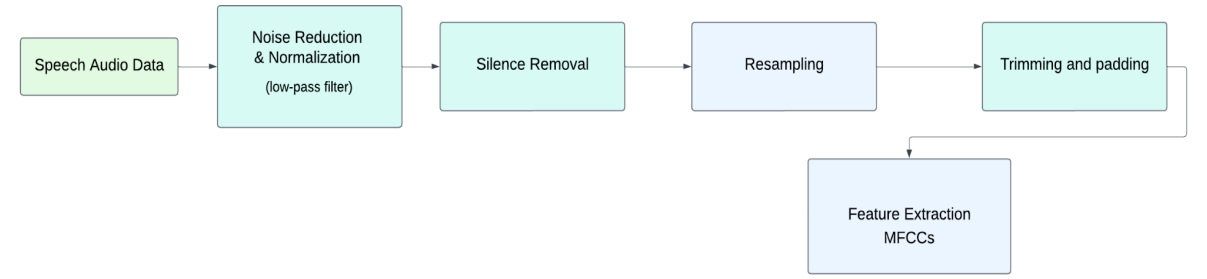
1.1 Dataset and Preprocessing

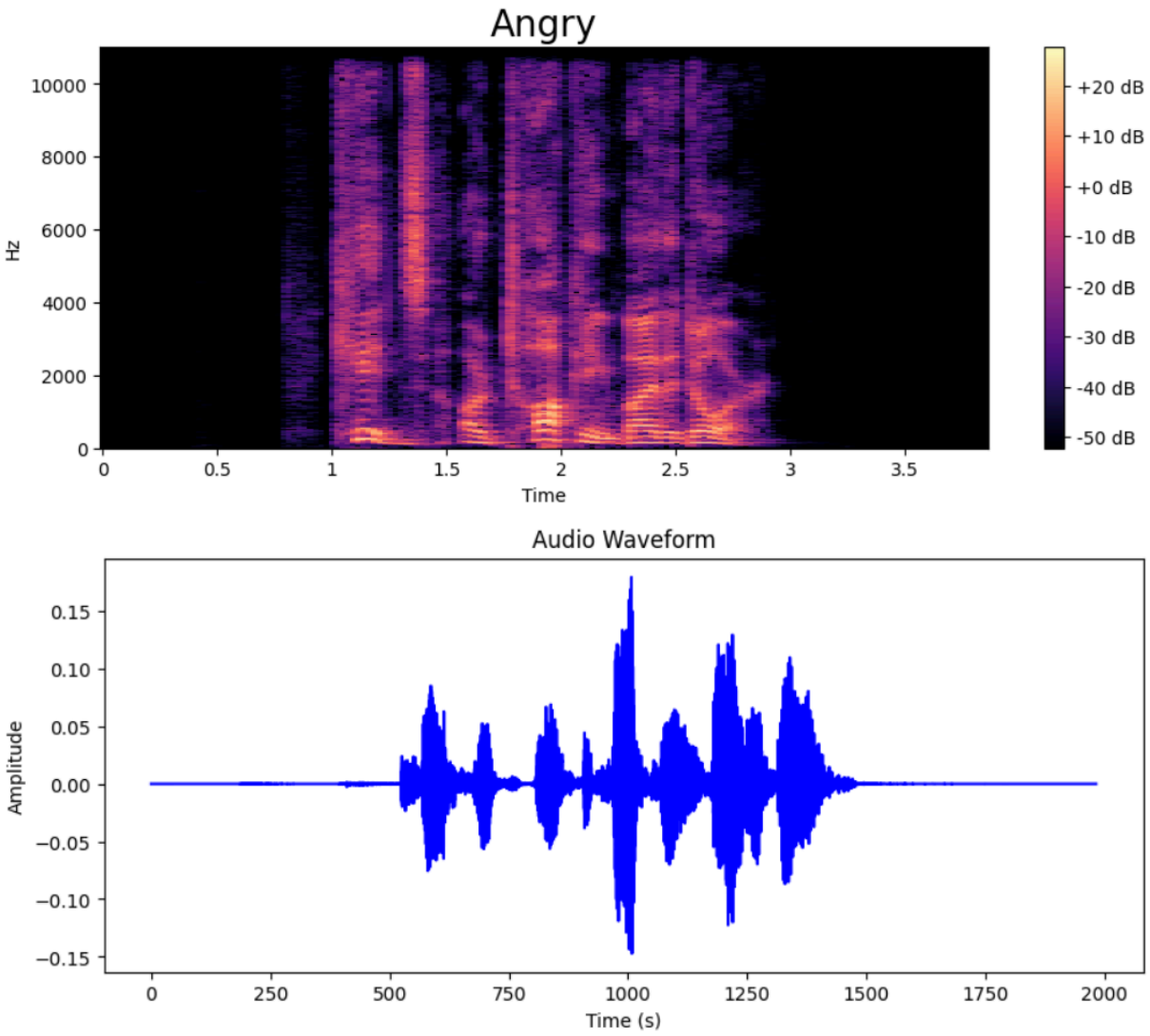
We use four speech emotion datasets: CREMA-D , RAVDESS , SAVEE, and TESS. These datasets provide a diverse collection of labeled speech samples representing various emotional states that includes anger, fear, sadness, happiness, disgust, surprise, and neutrality.

Before feeding the audio data into the model, we apply a series of preprocessing steps to enhance feature extraction and improve classification accuracy. Noise reduction is performed using a low-pass filter to eliminate background disturbances. Resampling is done to standardize all audio samples to 16kHz to ensure consistency across the different datasets. Trimming and padding are applied to maintain a uniform length across all speech samples to prevent any inconsistencies due to varying durations.

We also use waveplot and spectrogram visualizations to analyze the temporal and frequency characteristics of different emotions, providing deeper insights into speech variations across emotional states. These preprocessing techniques are applied on the dataset for effective model training and improve the accuracy of emotion recognition.

PREPROCESSING FLOW DIAGRAM:





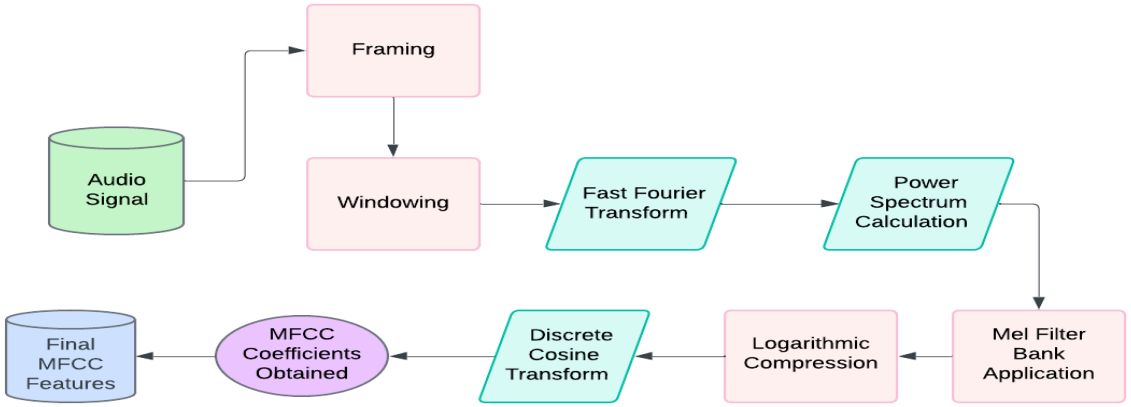
1.2 Feature Extraction using Mel-Frequency Cepstral Coefficients

To transform raw speech signals into a suitable format, we extract Mel-Frequency Cepstral Coefficients. MFCCs are widely used in speech and emotion recognition tasks as they effectively capture the spectral characteristics of speech while filtering

out irrelevant variations. They provide a compact representation of speech signals which makes them highly useful for ML models.

The extraction process involves several key steps. First, pre-emphasis filtering is applied to amplify high-frequency components, ensuring that crucial speech features are preserved. The speech signal is then segmented into short frames (20-40ms) using framing and windowing techniques, allowing the model to capture short-term spectral properties. After segmentation, the Fourier Transform is performed to convert the signal from the time domain to the frequency domain. Next, Mel filterbank processing is applied, where a set of triangular filters are used to scale the frequencies based on human auditory perception, emphasizing perceptually important frequency bands. Finally, the logarithm and Discrete Cosine Transform (DCT) are used to compress the spectral information into a compact set of coefficients, reducing dimensionality while retaining key speech patterns.

MFCC Extraction Flow:



The extracted MFCCs for each audio sample forms a numerical representation of speech features that is then used as input for the deep learning model. The dataset contains 12,162 processed audio samples, each represented by a feature vector of 60 MFCC coefficients.

```
def extract_mfcc(filename):
    y, sr = librosa.load(filename, duration=3, offset=0.5)
    mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n_mfcc=60).T, axis=0)
    return mfcc
```

```
x = data2['paths'].apply(lambda x: extract_mfcc(x))
```

```
0        [-287.03732, 129.61015, 9.607376, 26.902523, -...
1        [-231.46657, 75.31091, -21.464455, 38.420166, ...
2        [-343.2415, 118.26945, 7.298577, 34.67476, -17...
3        [-326.90222, 123.055435, -0.23093681, 46.83661...
4        [-242.48978, 111.04384, -24.815582, 45.390205,...
                                ...
12157    [-237.88043, 95.220985, 0.9635511, 28.659952, ...
12158    [-197.90538, 113.72018, -4.9860163, 29.73043, ...
12159    [-334.3724, 60.880363, -14.755251, -2.3327641,...
12160    [-342.9861, 94.82616, -24.713985, -26.03091, -...
12161    [-426.46286, 106.69511, 4.007008, -20.765448, ...
Name: paths, Length: 12162, dtype: object
```

1.3 Deep Learning Model for Emotion Classification

After extraction, we work on classifying emotions from the speech. We implement a **hybrid deep learning model** that integrates **CNNs** and **BiLSTM networks**. This architecture uses the strengths of both CNNs, which excel at extracting spatial features, and BiLSTMs that can capture sequential dependencies in speech signals, developing a strong classification system.

The model architecture is structured as follows:

- Convolutional Layers: The first stage of the model applies 1D convolutional layers to extract local patterns from the MFCC spectrogram. We identify key acoustic features relevant to emotion detection using these layers.
- Batch Normalization: Batch normalization layers are added after convolutional layers, normalizing activations and reducing internal covariate shifts to improve training stability and convergence.
- Bidirectional LSTM Layers: Since speech is inherently sequential, BiLSTM layers are incorporated to capture temporal dependencies in both forward and backward directions. This enables the model to better understand the context of speech signals.
- MaxPooling and Dropout: MaxPooling layers help downsample feature representations, reducing computational complexity while retaining crucial information. Dropout layers prevent overfitting by randomly deactivating neurons during training.
- Fully Connected Layers: The final layers map the extracted features to emotion labels. A softmax activation function is applied in the output layer to classify speech into seven distinct emotions: anger, disgust, fear, happiness, neutral, sadness, and surprise.

We trained the model using the categorical cross-entropy loss function, which is suitable for multi-class classification. The performance is evaluated based on validation accuracy, ensuring that the model generalizes well to unseen speech data.

```
from keras.models import Sequential
from keras.layers import Dense, LSTM, Dropout
from tensorflow.keras.callbacks import ModelCheckpoint
from keras.models import Sequential
from keras.layers import Dense, LSTM, Dropout
from keras.models import Model
from keras.models import Sequential
from keras.layers import Conv1D, MaxPooling1D #, AveragePooling1D
from keras.layers import Flatten, Dropout, Activation # Input,
from keras.layers import Dense , Embedding
from keras import layers, models

model = Sequential()

model.add(layers.Conv1D(256, 5, input_shape=(60, 1), activation='relu'))
model.add(layers.BatchNormalization())
model.add(layers.Conv1D(128, kernel_size=3, activation='relu'))
model.add(layers.MaxPooling1D(pool_size=2))
model.add(layers.BatchNormalization())
model.add(layers.Bidirectional(layers.LSTM(128, return_sequences=True)))
model.add(layers.Bidirectional(layers.LSTM(128, return_sequences=True)))
model.add(layers.GlobalMaxPooling1D())
model.add(layers.Dense(128, activation='relu'))
model.add(layers.BatchNormalization())
model.add(layers.Dropout(0.2))
model.add(layers.Dense(7, activation='softmax'))

model.compile(loss='categorical_crossentropy', optimizer='rmsprop', metrics=['accuracy'])
model.summary()
```

Once the speaker's emotion is detected, it is integrated into the LLM to dynamically adjust its response tone. Instead of providing generic replies, the LLM modifies its language style based on the detected emotional state which is our goal- to enhance conversational engagement by making interactions more empathetic.

1.4 LLM Integration Implementation

The detected emotional state is linked to a predefined set of response adjustments. For example, if frustration is detected, the LLM generates a more supportive and understanding response rather than a neutral one. The LLM input prompt is modified to include emotion metadata, ensuring that responses align with the user's emotional state. This step helps in contextual adaptation, allowing the model to fine-tune its language dynamically. The LLM generates responses with adjusted tone, vocabulary, and sentence structure to match the user's emotion. This ensures that the conversation feels natural and emotionally intelligent.

```python
def generate_emotion_aware_response(text, emotion):
    prompt = f"User is feeling {emotion}. Respond accordingly: {text}"

    response = llm.generate_response(prompt)
    return response
```

```python
user_text = "I don't think this is working for me."
detected_emotion = "frustration"
llm_response = generate_emotion_aware_response(user_text, detected_emotion)

print(llm_response)
```

Comparison of Text-Only vs. SER-Enhanced LLM Responses-

User Input	Text-Only LLM Response	SER-Enhanced LLM Response
"I just feel so exhausted... nothing is going right." *(Sad tone)*	"I'm sorry to hear that. What happened?"	"I hear you. It sounds like you're feeling really drained. Want to talk about what's been going on?"
"Ugh, I can't believe this happened!" *(Angry tone)*	"That sounds frustrating. What happened?"	"I understand why you're upset. Do you want to vent about it?"
"Wow! That's amazing!" *(Excited tone)*	"That's great to hear!"	"That's awesome! Tell me more!"

1.5 Adaptive Conversational Framework

Traditional AI responses often feel disjointed because they fail to retain and adapt to shifts in the user's emotional state. To address this, our system incorporates an adaptive conversational framework that ensures responses are not only contextually relevant but also emotionally coherent. This is achieved through a stateful emotion memory that tracks the user's emotional progression over multiple interactions, preventing abrupt shifts in tone.

The system dynamically adjusts its responses by analyzing the trajectory of detected emotions. For instance, if a user moves from frustration to relief, a typical response model might abruptly switch from an apologetic tone to a neutral one, making the conversation feel robotic. In contrast, our framework ensures a smooth transition by gradually shifting from an empathetic or solution-oriented tone to a more reassuring and appreciative one. This nuanced adaptation fosters a more human-like interaction, making the user feel truly understood rather than merely acknowledged.

To achieve this, we implement a sliding window mechanism that retains emotional context from previous exchanges, allowing the AI to recognize whether an emotion is persistent or momentary. Additionally, a weighted emotion averaging approach is used to prioritize recent emotions while still considering past states. This prevents the system from overreacting to temporary mood fluctuations, ensuring that responses feel natural and consistent. Emotion tracking is embedded directly into the LLM’s prompt, allowing the model to generate responses that align with the ongoing emotional flow rather than reacting in isolation to a single detected emotion.

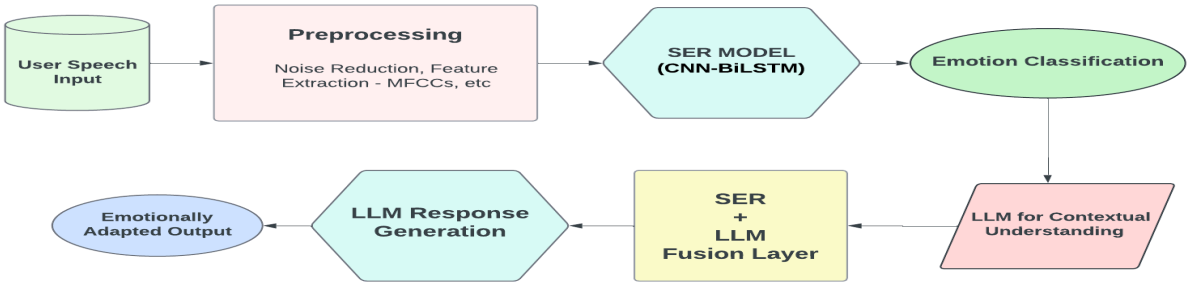
User's Emotional Shift	Naïve Response	Adaptive Response
User (frustrated): "This app never works!	"I understand your issue."	"I can see this is frustrating. Let me fix it for you right away."
User (calming down): "Thanks, I just need it to work."	"Okay, here’s how to fix it."	"I appreciate your patience. Let’s go step by step and get this sorted."

The goal of adaptive framework is to transform standard AI interactions into emotionally intelligent dialogues that maintain coherence across multiple exchanges. This ensures that the system does not simply recognize emotions but responds in a way that reflects an evolving and contextual understanding of the user's state of mind.

1.6 System Architecture

The system is designed as a multi-stage pipeline that integrates speech processing, emotion recognition, and LLM-based response generation. It consists of three main modules:

- Speech Processing Module that extracts relevant features from raw speech and classifies emotions using the CNN-BiLSTM model. The detected emotion is then passed as metadata to the next stage.
- Emotion-Aware Response Generation module where the recognized emotion is mapped to an appropriate LLM response modification, ensuring that the generated text aligns with the user's mood.
- Conversational Adaptation module that maintains an emotional memory to ensure responses evolve naturally over multiple interactions.



Results

The model was trained for 400 epochs using a batch size of 64, achieving a training accuracy of 99.7%. However, the validation accuracy varied due to the complexity of real-world speech variations, with the best validation accuracy reaching 84.7%.

The model exhibited strong performance under clean audio conditions, achieving high recognition accuracy. However, when tested with noisy or low-quality recordings, accuracy dropped by approximately 8.5%, indicating the impact of background noise and variations in speaker tone. This shows the challenge of

adapting the model for real-world audio scenarios with inconsistencies in recording quality.

Despite some misclassification in neutral and sad emotions, our model performed well in detecting high-energy emotions such as anger with a precision of 83%. The model's ability to generalize was further validated through emotion classification on unseen audio samples, where it successfully identified emotions with high confidence. When integrated with a LLM, the SER system significantly enhanced AI-generated responses by incorporating detected emotions. Compared to text-only emotion recognition, which often failed to capture vocal nuances, our approach led to more emotionally appropriate and empathetic responses.

This was evident in user interactions, where responses adjusted dynamically based on detected emotions, improving conversational fluidity and engagement. Our future work will focus on increasing cross-corpus adaptability and reducing misclassification rates for low-energy emotions like sad and neutral.

Conclusion

This study demonstrates that combining SER with LLMs enhances AI-generated responses by making them more emotionally intelligent. The results confirm that speech-based emotion detection is more effective than text-only methods, leading to better sentiment understanding and more natural conversations. However, we face challenges when it comes to dealing with noisy environments and capturing more complex emotional states.

We aim to focus on expanding the range of emotions detected and improving noise-handling techniques in the future. Multimodal approaches that combine speech with facial expressions and gestures can further refine emotional understanding. Enhancing real-time efficiency will also be crucial for applications requiring immediate AI-generated responses, such as virtual assistants and customer service chatbots.

While Speech Emotion Recognition significantly improves the emotional intelligence of LLMs over text-based detection, the combination of speech with text-based emotion detection, known as multimodal emotion recognition goes a bit further to enhance the system's ability to comprehend and respond to human emotions better. Multi-Modal Emotional Recognition could represent the next big step in building emotionally intelligent AI systems that better meet the emotional needs of humans.

References

[1] P. Ekman and W. V. Friesen, *Unmasking the Face: A Guide to Recognizing Emotions from Facial Clues*, Englewood Cliffs, NJ, USA: Prentice-Hall, 1975.

[2] S. Schuller, B. Schmitt, D. Arsic, F. Wallhoff, and G. Rigoll, "Feature extraction methods for emotion recognition and classification," *IEEE International Conference on Multimedia and Expo (ICME)*, pp. 151-154, 2005.

[3] Z. Aldeneh and E. M. Provost, "Using CNNs to Classify Emotion in Speech: Challenges and Applications," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 12, pp. 2066-2078, Dec. 2018.

[4] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 12449-12460, 2020.

[5] H. Huijuan, Y. Jiahua, and L. Xia, "Cross-Corpus Speech Emotion Recognition Using Deep Local Domain Adaptation," *IEEE Transactions on Affective Computing*, vol. 13, no. 2, pp. 636-647, 2022.

[6] Y. Zadeh, P. P. Liang, S. Poria, E. Cambria, and L.-P. Morency, "Multi-Modal Emotion Recognition from Speech and Text using Deep Learning," *IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1-6, 2018.

[7] R. Ravi and K. S. Rajasekaran, "Emotion Detection in Text using Pre-Trained Transformers," *IEEE Access*, vol. 9, pp. 78634-78647, 2021.

[8] R. Davis and C. Busso, "Multimodal Emotion Recognition Using Speech and Text Data," *IEEE Transactions on Affective Computing*, vol. 14, no. 1, pp. 123-136, 2023.

[9] A. Vasilenko, I. A. Sokolov, and O. V. Kudryavtseva, "Multimodal Sentiment and Emotion Recognition with Cross-Attention Mechanisms," *Neural Computing and Applications*, vol. 35, pp. 10021-10038, 2023.

[10] Y. Xu, Z. Zhang, and L. Xie, "Emotion-Aware Chatbots: Enhancing Conversational AI with Speech Emotion Recognition," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7454-7458, 2021.

[11] T. Brown, B. Mann, N. Ryder, et al., "Language Models are Few-Shot Learners," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 1877-1901, 2020.
[12] Y. Gong, K. Honda, and S. Nakamura, "Incorporating Speech Context into Text-Based Sentiment Analysis," *IEEE Transactions on Affective Computing*, vol. 15, no. 2, pp. 789-801, 2024.
[13] Suraj Pradeepkumar, Ragul K, Raghul S, & Stalin M. *Speech Emotion Recognition Using Deep Learning*. Journal of Current Research in Engineering and Science, January 2024. Available: https://www.psvpec.in/jcres/2024_2/A21.pdf
[14] Shaila S. G., Reddy B. S., and Kumar K. S., "Speech Emotion Recognition Using Machine Learning Approach," *Proceedings of the International Conference on Advances in Management, Information, and Data Sciences (ICAMIDA-22)*, Atlantis Press, pp. 593-600, 2022.
[15] J. Guru Monish, Amartya S., and Magesh Kumar, "Speech Emotion Recognition in Machine Learning to Improve Accuracy using Novel Support Vector Machine and Compared with Random Forest Algorithm," *2022 5th International Conference on Contemporary Computing and Informatics (IC3I), IEEE Xplore*, December 2022.