Audio-Driven Emotion Recognition and Contextual Summarization for Mental Health Conversations

Ravi Raj
22BDS051
22bds051@iiitdwd.ac.in

Parishri Rakesh Shah
22BD0S43
22bds043@iiitdwd.ac.in

Pakhi Singhal
22BDS042
22bds042@iiitdwd.ac.in

Preethi Varshala S
22BDS045
22bds045@iiitdwd.ac.in

Here is a link to my GitHub repository: GitHub Repository

Abstract - We developed an AI-driven system for mental health applications that focuses on Emotion Detection and Text Summarization, aimed at enhancing support for patient-therapist interactions. Our workflow begins with Audio-Driven Emotion Recognition (ADER), where text is converted to audio using Google Text-to-Speech (GTTS). Key audio features are then extracted through Mel-frequency cepstral coefficients (MFCC), enabling robust feature analysis. To classify emotions, we train the XG Boost and Random Forest machine learning models. The system predicts emotional states using these trained models on new sample inputs. In Contextual Summarization and Emotion Prediction (CSEP), we preprocess the dataset by tokenizing the utterances and label encoding the Dialogue Act and Emotion columns. Using a custom BERT model with embeddings, data is structured for training in PyTorch, and K-fold cross-validation is applied to enhance generalization. After training the BERT model, a summarization model using BART with a threshold of 30 is applied, enabling emotion predictions on summarized text. This workflow, combining cross-validation and summarization, provides robust emotion analysis and summarization, enhancing the effectiveness of AI-driven mental health support systems.

Keywords: Emotion Detection, Text Summarization, MFCC Feature Extraction, Custom BERT Model, BART

I. Introduction

Mental health disorders impact millions worldwide, underscoring the urgent need for accessible and effective support systems. Recent advancements in AI have enabled the creation of virtual mental health assistants capable of analyzing emotions and summarizing patient-therapist dialogues. This project focuses on two AI-driven approaches for mental health support: Audio-Driven Emotion Recognition (ADER) and Contextual Summarization and Emotion Prediction (CSEP). We implement Emotion Detection by converting text to audio, extracting key features, and training two machine learning models - XGBoost and Random Forest to classify emotional states. Additionally, we apply Text Summarization using BART to input text from user whose word count is more than 30 and then train the model using BERT and applying K-fold cross validation predict it's emotion.

Dialog Act	Dialog Act meaning	Dialog Act	Dialog Act meaning
nan	Not Applicable or No Act	com	Complaint
crq	Clarification-Request	hp	Humor/Playfulness
ack	Acknowledgment	irq	Information-Request
id	Information-Delivery	yq	Yes/No Question
orq	Opinion-Request	gc	General Chit-Chat
cv	Correction/Verification	cd	Clarification-Delivery
op	Offer/Proposal	ci	Confirmation/Instruction
ap	Apology	on	Opinion-Negative
od	Opinion-Delivery	gt	Greeting
prq	Permission Request		

Fig. 1. Dialog Act.

II. Dataset Description

The dataset contains a total of 5,247 utterances. Each entry is marked by a unique ID. The dataset captures specific Dialog Acts as shown in the Fig 1, which represent the function of each utterance in a conversation, such as Information-Request, Acknowledgment, General Chit-Chat, and Clarification-Request. The dataset includes an Emotion column with three values: -1 (negative), 0 (neutral), and 1 (positive), representing the emotional tone of each utterance. These values provide a clear indication of the level of emotional engagement, aiding in the classification process. These annotations provide crucial context for understanding the emotional tone and purpose of each interaction, aiding in building accurate model for emotion classification. 5,247 rows are used for emotion detection from text using BERT. 1,163 rows are used for emotion detection from audio using MFCC features

III. Implementation Workflow

A. *Audio-Driven Emotion Recognition (ADER)*

*1) Convert Text to Audio using **GTTS**:* The Google Text-to-Speech (GTTS) tool is used to convert the text-based utterances into speech (audio) files. Each text entry is processed and saved as an individual audio file (typically in .mp3 or .wav format). These audio files provide a medium for extracting speech-based features, which are crucial for emotion detection.

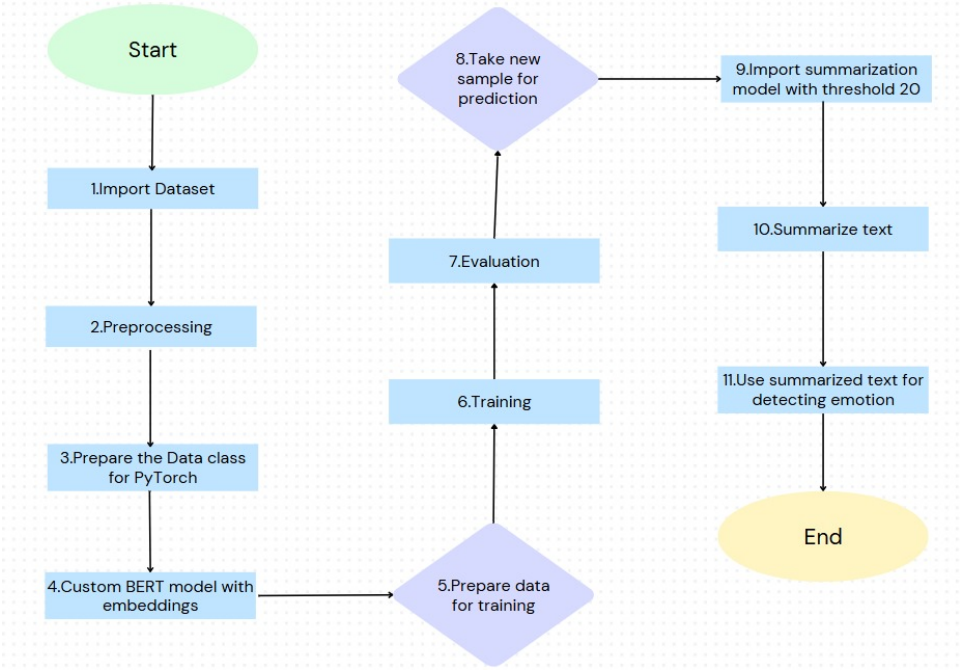

Fig. 2. Flowchart for Audio-Driven Emotion Recognition (ADER)

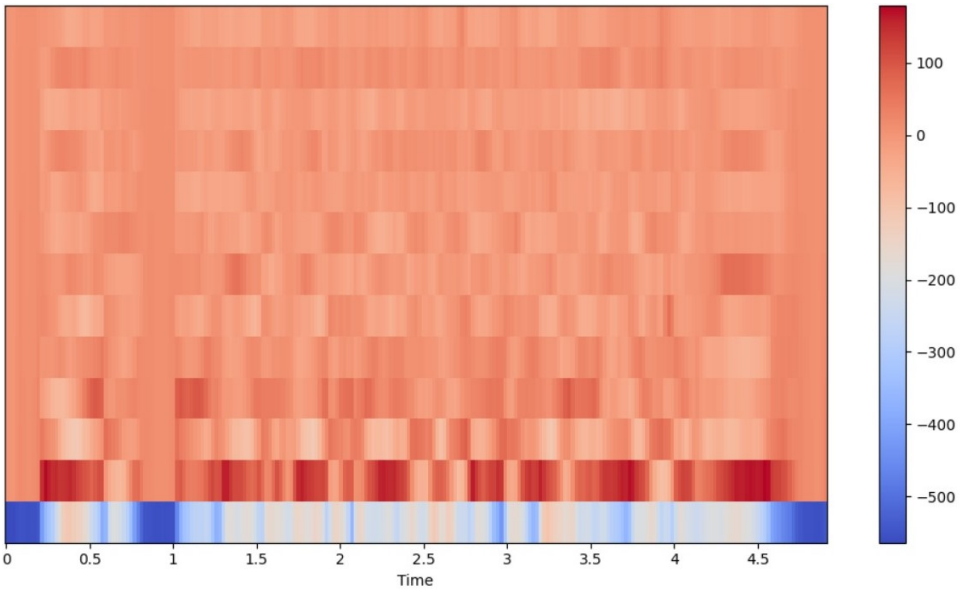

Fig. 3. MFCC Plot for an audio file

2) Extract MFCC Features from Audio:

- **Mel-Frequency Cepstral Coefficients (MFCC) Extraction** From the generated audio files, MFCC features are extracted. MFCCs are widely used in speech recognition tasks because they capture essential audio characteristics such as tone, frequency variations, and emphasis patterns. The MFCC extraction involves breaking down the audio signals into their frequency components, identifying the significant patterns that correspond to various emotional expressions. The MFCC plot in Fig 3 represents the Mel-Frequency Cepstral Coefficients for an audio file. The X-axis represents time in seconds, showing how the audio characteristics change over its duration, while the Y-axis represents different MFCC coefficients, each capturing features across specific frequency bands (with higher coefficients corresponding to higher frequencies). The color intensity of each cell indicates the amplitude of the MFCC value at a given time and frequency coefficient. Red areas reflect higher energy, where the sound is more pronounced, while blue areas show lower energy. Overall,

MFCC plots help analyze the spectral characteristics of an audio signal over time, making it easier to identify sound patterns or phonetic elements in speech.

- **Feature Preparation** The MFCC features from each audio file are converted into numerical representations, which can be fed into machine learning models for training. This step ensures that the data is in a suitable format for further analysis and classification.

3) Model Training: The dataset is split into training and testing subsets to ensure robust model performance evaluation. The training phase involves feeding the MFCC features to the machine learning models, which learn patterns from the data and map them to emotion labels (-1 for negative, 0 for neutral, 1 for positive). XG Boost and Random Forest models are employed to train on the MFCC features.

4) Evaluation: The report in Fig 3 reveals that the Random Forest model performs well in predicting neutral emotions (class 0), with high precision (0.77) and recall (0.93), but struggles significantly with other emotions. The low precision (0.59) and very poor recall (0.26) for negative emotions (class -1) suggest that the model fails to identify many true negative emotion instances. Additionally, the model completely fails to detect positive emotions (class 1) due to the extremely small number of examples in the dataset. Overall, while the model achieves a decent accuracy of 0.75.

accuracy			0.75	233
macro avg	0.45	0.40	0.40	233
weighted avg	0.71	0.75	0.71	233

Fig. 4. Random Forest classification report

The report in Fig 4 shows that the XGBoost model performs best for neutral emotions (class 1) with high precision (0.80), recall (0.90), and an F1-score of 0.85, indicating strong detection of neutral emotions. However, it struggles with negative emotions (class 0), where recall is low (0.44) and the F1-score is only 0.51, meaning many true negative emotions are missed. Similar to the Random Forest model, XGBoost fails to detect positive emotions (class 2) due to the very low support (only 2 instances). The overall accuracy is 0.77, but the model's macro averages are lower, highlighting the challenges with imbalanced classes.

accuracy			0.77	233
macro avg	0.47	0.44	0.45	233
weighted avg	0.75	0.77	0.75	233

Fig. 5. XGBoost classification report

5) Predict on New Samples: For any new text input or audio input, the same workflow is followed. If the input is text, it is first converted to audio using GTTS. If the input is already an audio file as shown in Fig 5 this step is bypassed. Whether the audio comes from GTTS or directly as an .mp3 file, MFCC features are then extracted from the audio. Once the MFCC features are obtained, they are fed into the trained model to predict the corresponding emotion (negative, neutral, or positive).

```
Randomly selected Audio File: audio_2268.mp3
Predicted Emotion: 0
```

Fig. 6. Emotion prediction of audio

B. Contextual Summarization and Emotion Prediction (CSEP)

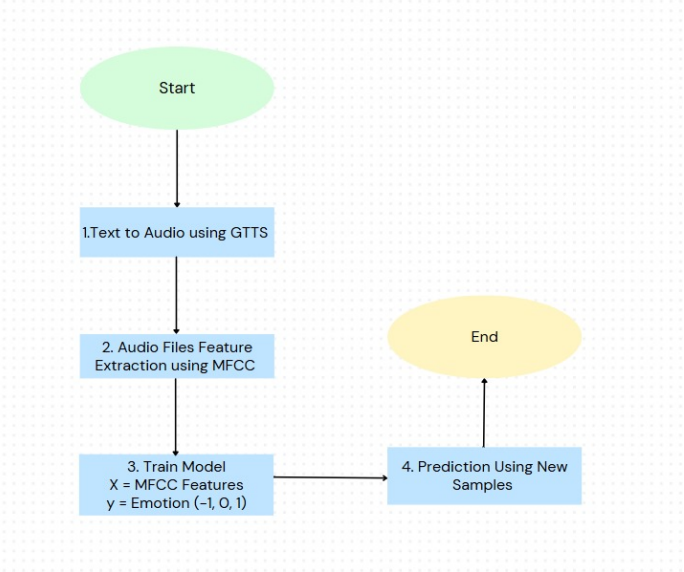

Fig. 7. Flowchart for Contextual Summarization and Emotion Prediction (CSEP)

1) Preprocessing: The text data is tokenized and labeled for efficient processing and embedding. A custom dataset class is implemented using PyTorch to organize the tokenized data and label-encoded dialogue acts and emotions. A custom BERT-based model is used to embed the tokenized utterances, enhancing the model's ability to understand dialogue context and emotional nuances, improving performance in tasks like emotion detection and dialogue act classification.

2) Model Training: During training, K-fold cross-validation is applied to evaluate the custom BERT model's performance on different subsets of the dataset, improving its generalization. The input features are the BERT-embedded representations of the text, while the output labels are the encoded emotions. The training set is split into K folds, where K-1 folds are used for training, and the remaining fold is used for validation. This process repeats for all K folds, and the model's performance is averaged across them. After cross-validation, the final model is trained on the entire training set and evaluated on the test set for generalization.

3) Evaluation: In Figure 7, we observe that the model performs well for neutral and negative emotions, achieving high precision and recall scores for these classes. However, it encounters significant difficulty with positive emotions, as indicated by a recall of only 0.04 and an F1-score of 0.07. This discrepancy may stem from an imbalance in the dataset, with fewer examples of positive emotions. Although the weighted averages indicate strong overall performance (F1-score of 0.94).

Test Accuracy: 0.9421

	precision	recall	f1-score	support
-1	0.90	0.87	0.88	1070
0	0.95	0.98	0.97	4098
1	1.00	0.04	0.07	79
accuracy			0.94	5247
macro avg	0.95	0.63	0.64	5247
weighted avg	0.94	0.94	0.94	5247

Fig. 8. BERT classification report

4) Prediction with Summarization Model: The system processes input by checking if the sentence has more than 30 words. If it does, the text is summarized using BART. Then the BERT model predicts the emotion (negative, neutral, or positive). If the sentence is 30 words or fewer, the system skips summarization and directly processes the text for emotion detection. The Fig 9, 10 and 11 displays the outputs for summarization and prediction of positive, negative and neutral emotion respectively.

```
Original Utterance: Well, because I don't want to stay like this. And, you know, I've already talked to her before. She was
really nice. So I wouldn't mind going back to her and she told me you know, when I saw her last time, if I ever needed I cou
ld come back.

Summarized Utterance: Well, because I don't want to stay like this. And, you know, I've already talked to her before. She wa
s really nice. So I wouldn't mind going back to her. When I saw her last time, if I ever needed I could come back.

Predicted Emotion:  1
```

Fig. 9. Text Summarization for positive emotion

```
Original Utterance: I've had a lot on my plate lately, both personally and professionally. I've been trying to stay on top o
f everything, but sometimes it feels like it's just too much. My schedule has been packed, and I've had very little time to
relax or take a break. I know I need to manage my time better and take care of my mental health, but it's hard to find that
balance. I feel like I'm constantly running from one task to the next.

Summarized Utterance: I've had a lot on my plate lately, both personally and professionally. I feel like I'm constantly runn
ing from one task to the next. I know I need to manage my time better and take care of my mental health.

Predicted Emotion: -1
```

Fig. 10. Emotion Detection for negative emotion

```
Original Utterance: I've had a lot on my plate lately, both personally and professionally. I've been trying to stay on top o
f everything, but sometimes it feels like it's just too much. My schedule has been packed, and I've had very little time to
relax or take a break. I know I need to manage my time better and take care of my mental health, but it's hard to find that
balance. I feel like I'm constantly running from one task to the next.

Summarized Utterance: I've had a lot on my plate lately, both personally and professionally. I feel like I'm constantly runn
ing from one task to the next. I know I need to manage my time better and take care of my mental health.

Predicted Emotion: -1
```

Fig. 11. Emotion Detection for neutral emotion

IV. Architecture Overview

A. *GTTS + MFCC*

The GTTS + MFCC pipeline focuses on converting text to speech and analyzing the emotional tone in the generated audio. GTTS (Google Text-to-Speech) takes raw text and synthesizes it into spoken words, creating an audio file that mimics natural human speech. This speech carries emotional cues through intonation, pitch, and rhythm, which are crucial for emotion recognition. To extract these features, MFCC (Mel-frequency cepstral coefficients) is used. MFCC analyzes the speech signal by capturing its frequency components, which are closely tied to the speaker's emotional expression. By analyzing the MFCC features, the system can detect emotions such as happiness, sadness, or anger based on how the speech sounds, rather than just the content of the words. This method is effective for recognizing emotions conveyed through vocal tone and speech patterns.

B. *BART + BERT*

The BART + BERT pipeline focuses on text-based emotion prediction. First, BART is employed for text summarization, condensing lengthy dialogues or written content into concise versions without losing essential meaning, which helps in processing long conversations efficiently. Then, BERT is used to generate contextual embeddings for the summarized text. These embeddings capture deep semantic relationships and contextual understanding of the input, enabling a more accurate emotion prediction. The embeddings are passed through a classifier, which predicts the emotional tone (e.g., happy, sad, or angry) based on the text's content. This pipeline is highly effective for detecting emotions from written language where sentiment is conveyed through words and phrasing.

Acknowledgment

We would like to express our sincere gratitude to our Professor Animesh Chaturvedi, for their invaluable guidance, constant support, and insightful feedback throughout the development of this project. Their expertise and encouragement have been instrumental in refining our ideas and approaches. We am also grateful to our college IIIT Dharwad, for providing the resources and platform that enabled us to carry out our work.

References

[1] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 1, 4171–4186.