EmoLingo: An Emotion-Aware Conversational Agent for Personalized Emotional Support

1st 戴凱麗 Tay Khai Li
Department of Computer Science
National Tsing Hua University
Hsinchu, R.O.C.
taykhaili88@gmail.com

1st 農政宇 LIONG ZHENG EE
Department of Computer Science
National Tsing Hua University
Hsinchu, R.O.C.
lze0603@gmail.com

1st 謝嘉銘 Michael Andrew Sucahyo
Department of Computer Science
National Tsing Hua University
Hsinchu, R.O.C.
michael.sucahyo@gmail.com

1st 王籽穎 Wong Zi Ying
Interdisciplinary Program of EECS
National Tsing Hua University
Hsinchu, R.O.C.
noobiestwong@gmail.com

1st 施淙綸 Tsung-Lun Shih
Interdisciplinary Program of Science
National Tsing Hua University
Hsinchu, R.O.C.
x69599596@gmail.com

***Abstract*—This project aims to develop an intelligent chatbot capable of providing real-time emotional support to users, particularly students. Traditional text-based chatbots often lack the ability to fully understand emotional nuances. In contrast, this chatbot leverages advanced Speech Emotion Recognition (SER) to detect emotions directly from voice inputs. By integrating Natural Language Processing (NLP), it generates meaningful and context-sensitive responses. This innovative system aims to improve emotional well-being through empathetic and supportive interactions, showcasing the potential of artificial intelligence (AI) to advance human emotional health and well-being.**

***Index Terms*—chatbot, machine learning, auto-reply, large language model, natural language processing.**

I. Introduction

In the modern educational landscape, students are experiencing unprecedented levels of academic pressure, contributing to an alarming rise in stress and mental health issues. Providing adequate emotional support for students has become a critical priority. However, traditional psychological counseling methods often lack the accessibility and scalability needed to address this growing demand.

Recent advancements in Natural Language Processing (NLP) and Large Language Models (LLMs) have opened up new possibilities for addressing this challenge. These state-of-the-art technologies enable the development of intelligent conversational agents capable of understanding and responding to human emotions with remarkable accuracy. By leveraging these innovations, it is now feasible to design personalized, real-time emotional support systems that complement traditional mental health resources.

This paper introduces *EmoLingo*, a chatbot designed to explore the potential of emotion-aware conversational systems in supporting students' emotional well-being. By integrating sentiment analysis and emotion recognition, *EmoLingo* aims to generate contextually appropriate responses to various emotional states, including offering empathy, encouragement, and celebration. Although still under development, this approach seeks to contribute to the growing field of technology-assisted mental health support, offering insights into how artificial intelligence (AI) can be harnessed to address emotional needs in an educational context.

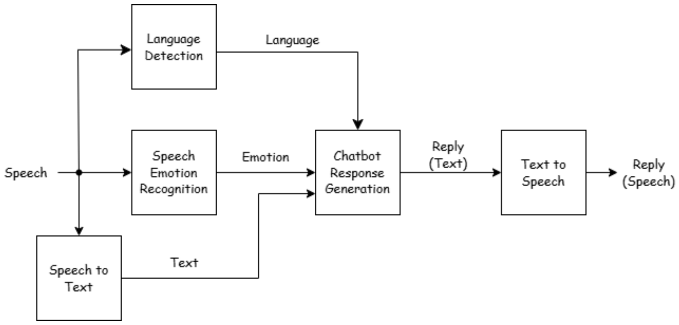


Fig. 1. The Complete Architecture of the Proposed EmoLingo Chatbot

II. Methods

This paper outlines the methodologies employed for Speech Language Detection, Speech-to-Text (STT), Speech Emotion Recognition (SER), Generative Pre-Trained Transformer (GPT) responses, and Text-to-Speech (TTS). The complete structure of the proposed *EmoLingo* is reveals in Fig. 1. Each section provides a concise explanation of these tasks and details the approaches implemented to address them.

A. Speech Language Detection

To overcome language barriers and enable seamless communication with *EmoLingo*, we use OpenAI's *Whisper* model to detect the language of user input [3]. This language detection determines how the STT model processes the original user input and ensures that *EmoLingo* generates responses in the appropriate language, improving the user experience.

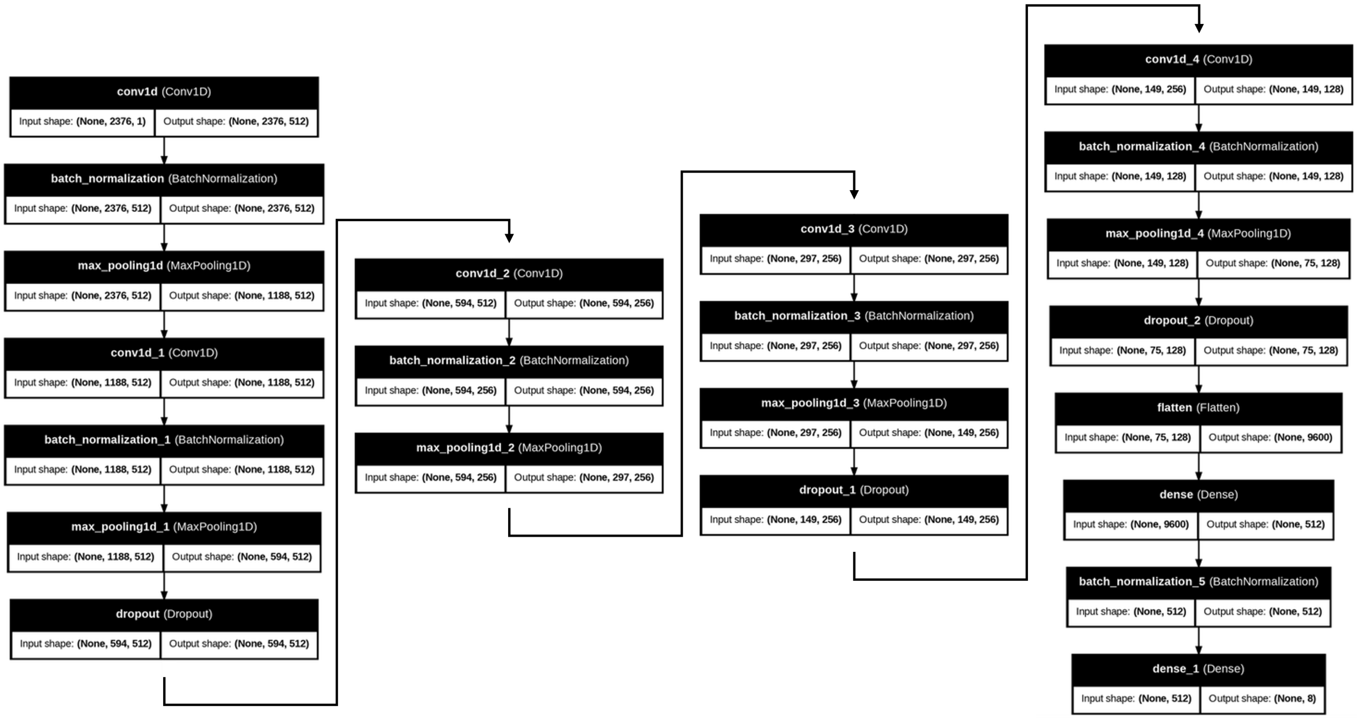

conv1d (Conv1D)
Input shape: (None, 2376, 1)
Output shape: (None, 2376, 512)
batch_normalization (BatchNormalization)
Input shape: (None, 2376, 512)
Output shape: (None, 2376, 512)
max_pooling1d (MaxPooling1D)
Input shape: (None, 2376, 512)
Output shape: (None, 1188, 512)
conv1d_1 (Conv1D)
Input shape: (None, 1188, 512)
Output shape: (None, 1188, 512)
batch_normalization_1 (BatchNormalization)
Input shape: (None, 1188, 512)
Output shape: (None, 1188, 512)
max_pooling1d_1 (MaxPooling1D)
Input shape: (None, 1188, 512)
Output shape: (None, 594, 512)
dropout (Dropout)
Input shape: (None, 594, 512)
Output shape: (None, 594, 512)
conv1d_2 (Conv1D)
Input shape: (None, 594, 512)
Output shape: (None, 594, 256)
batch_normalization_2 (BatchNormalization)
Input shape: (None, 594, 256)
Output shape: (None, 594, 256)
max_pooling1d_2 (MaxPooling1D)
Input shape: (None, 594, 256)
Output shape: (None, 297, 256)
conv1d_3 (Conv1D)
Input shape: (None, 297, 256)
Output shape: (None, 297, 256)
batch_normalization_3 (BatchNormalization)
Input shape: (None, 297, 256)
Output shape: (None, 297, 256)
max_pooling1d_3 (MaxPooling1D)
Input shape: (None, 297, 256)
Output shape: (None, 149, 256)
dropout_1 (Dropout)
Input shape: (None, 149, 256)
Output shape: (None, 149, 256)
conv1d_4 (Conv1D)
Input shape: (None, 149, 256)
Output shape: (None, 149, 128)
batch_normalization_4 (BatchNormalization)
Input shape: (None, 149, 128)
Output shape: (None, 149, 128)
max_pooling1d_4 (MaxPooling1D)
Input shape: (None, 149, 128)
Output shape: (None, 75, 128)
dropout_2 (Dropout)
Input shape: (None, 75, 128)
Output shape: (None, 75, 128)
flatten (Flatten)
Input shape: (None, 75, 128)
Output shape: (None, 9600)
dense (Dense)
Input shape: (None, 9600)
Output shape: (None, 512)
batch_normalization_5 (BatchNormalization)
Input shape: (None, 512)
Output shape: (None, 512)
dense_1 (Dense)
Input shape: (None, 512)
Output shape: (None, 8)

Fig. 2. The Full Structure of the SER Model

B. STT

We begin by converting speech into text to handle potential errors effectively. We setup a recognizer object from the *speech_recognition* library. The recognizer processes the audio content to prepare it in a format suitable for recognition. Finally, the `recognizer.recognize_google` method then transcribes the audio data to text using the Google's Speech Recognition API.

C. SER

The proposed model utilizes a Convolutional Neural Network (CNN) architecture as Fig. 2. The SER model consists of multiple convolutional and pooling layers, followed by dense layers for emotion classification [1]. The model is trained on the RAVDESS dataset, a widely recognized labeled dataset for emotion recognition. To preprocess the audio data, features such as MFCCs, zero-crossing rate (ZCR), and spectral contrast are extracted from the raw audio. Additionally, data augmentation techniques, including noise addition and pitch shifting, are applied to enhance the dataset. Noise augmentation simulates real-world noisy environments, while pitch shifting introduces variations in tone. Features from the original and augmented audio are combined to form a richer input dataset, improving the model's robustness and generalization. The processed audio features are converted into spectrograms, which serve as input to the CNN. During training, the Adam optimization algorithm refines the model by mapping these features to their corresponding emotion labels, enhancing performance over time.

To further improve the training process, two additional techniques are employed. First, early stopping is implemented to prevent overfitting by monitoring validation accuracy (`val_acc`). If the validation accuracy does not improve for 5 consecutive epochs, the training is halted, and the model's weights are restored to the epoch with the best validation performance. This is achieved using the `EarlyStopping` callback with parameters such as `patience=5` and `restore_best_weights=True`.

Second, learning rate reduction is applied to adjust the training dynamics when the validation accuracy plateaus. The `ReduceLROnPlateau` callback monitors the validation accuracy and reduces the learning rate by a factor of 0.5 if no improvement is observed for 3 consecutive epochs. This ensures that the model can continue fine-tuning with smaller learning steps, while the minimum learning rate is capped at `0.00001` to avoid stagnation.

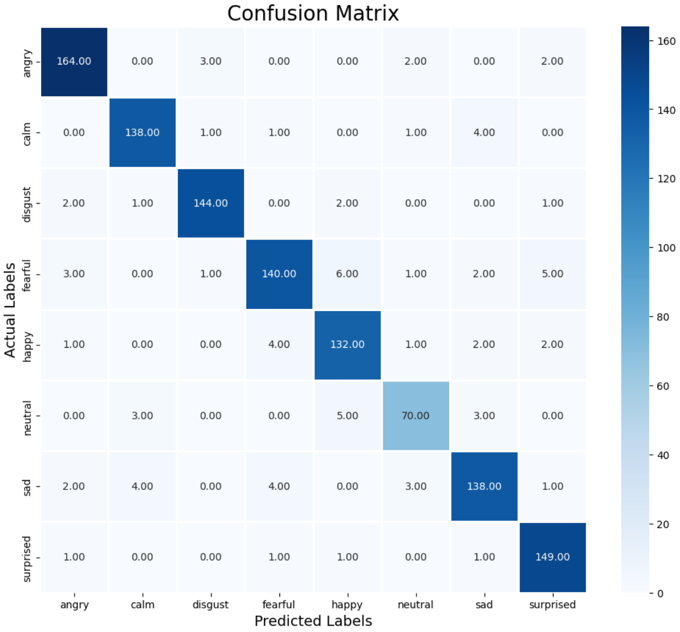
To evaluate the model, the RAVDESS dataset is partitioned into training and testing subsets. Following training, the model's performance is assessed on the testing set to ensure its effectiveness in recognizing emotions from spectrograms. This evaluation process validates the model's generalization capability and its reliability when applied to unseen data.

Before running the pipeline, several pre-trained components are loaded, including *scaler 2* and *encoder 2*, which were carefully trained and saved as part of our Speech Emotion Recognition (SER) pipeline development. Both *scaler 2* and *encoder 2* play essential roles in handling data preprocessing and postprocessing to ensure the model's inputs and outputs are appropriately transformed. The purpose of *scaler 2* is to normalize the features extracted from audio files to a consistent scale. Audio features, such as MFCC, ZCR, and RMSE, often have vastly different ranges. For instance, MFCC values might range from -100 to +50, while RMSE values typically range from 0 to 1. On the other hand, *encoder 2* converts numeric predictions back into human-readable emotion labels. During training, emotion labels like “*Happy*,” “*Sad*,” and “*Angry*” are encoded into numeric class indices (e.g., 0, 1, 2, etc.) to serve as target labels for the model. By using these components, which we trained and saved as part of the SER pipeline, the system ensures robust performance for both data normalization and result interpretation.

D. GPT

The *system_message* is inserted at the beginning of the conversation history. GPT generates responses based on the provided conversation context, starting with the first message in the *messages* list (i.e., *messages[0]*). Adding the *system_message* at the start ensures GPT is aware of the rules or context governing the conversation.

- **Emotion-specific adjustments:** If an emotion is detected, the assistant adapts its behavior by appending emotion-specific instructions to the *system_message*. These instructions are retrieved from the *emotion_guidelines* dictionary. For example, if the detected emotion is ”*Sad*”, the assistant might include the instruction: ”*Provide comforting and encouraging words to uplift their mood.*”
- **Handling Unintelligible Audio:** If GPT’s response contains the phrase ”*Sorry, I couldn’t understand the audio.*”, the handling function generates a predefined, canned response in the appropriate language (e.g., ”*Sorry, I couldn’t understand what you said. Could you repeat it one more time?*”). The tailored *system_message* is prepended to the conversation history (message list) to ensure that GPT processes the conversation with the desired context.
- **Streaming GPT API Call:** The OpenAI API is utilized to enable streaming responses. The function *client.chat.completions.create* is called with the GPT-3.5 model and the *messages* list. Setting the stream=True option allows for real-time generation of responses. The function accumulates chunks of the generated response in reply as they arrive, enabling a seamless and interactive user experience. The *messages* list maintains the entire conversation history between the user and the assistant. Each entry in the list is a dictionary containing the key ”*role*”, which specifies the speaker (e.g., ”*user*” for the user , ”*assistant*” for GPT).
- **Language Context Setup:** The assistant generates an initial ”*system_message*” based on the user’s language, utilizing the *language_context* dictionary. This message provides GPT with contextual instructions on how to respond (e.g., empathetically and in the specified language). For example, in the case of French(*fr*): ”*Vous êtes un assistant qui répond en français. Répondez avec empathie en tenant compte de l’émotion de l’utilisateur.*”



	angry	calm	disgust	fearful	happy	neutral	sad	surprised
angry	164.00	0.00	3.00	0.00	0.00	2.00	0.00	2.00
calm	0.00	138.00	1.00	1.00	0.00	1.00	4.00	0.00
disgust	2.00	1.00	144.00	0.00	2.00	0.00	0.00	1.00
fearful	3.00	0.00	1.00	140.00	6.00	1.00	2.00	5.00
happy	1.00	0.00	0.00	4.00	132.00	1.00	2.00	2.00
neutral	0.00	3.00	0.00	0.00	5.00	70.00	3.00	0.00
sad	2.00	4.00	0.00	4.00	0.00	3.00	138.00	1.00
surprised	1.00	0.00	0.00	1.00	1.00	0.00	1.00	149.00

Fig. 3. The Confusion Matrix of the SER Models for Different Emotions

E. TTS

The `tts.tts_to_file` method is employed to generate an audio file by specifying key parameters, such as the destination file path for the generated audio, the language in which the content should be spoken, and the intended emotional tone to effectively convey the desired sentiment. This method supports a wide range of languages, including English (*en*), Spanish (*es*), French (*fr*), German (*de*), Italian (*it*), Portuguese (*pt*), Polish (*pl*), Turkish (*tr*), Russian (*ru*), Dutch (*nl*), Czech (*cs*), Arabic (*ar*), Simplified Chinese (*zh-cn*), Hungarian (*hu*), Korean (*ko*), Japanese (*ja*), and Hindi (*hi*).

III. Results

Our work successfully aligns with our initial goal of providing precise responses and accurate emotional detection in user messages. The following sections present a detailed analysis of our final results:

A. Training

Datasets. To train our SER model, we use the RAVDESS emotional speech audio dataset. The RAVDESS dataset comprises 1,440 audio files recorded by 24 professional actors (12 male and 12 female) vocalizing two lexically-matched statements in a neutral North American accent. The recordings capture multiple emotions, including calm, happy, sad, angry, fearful, surprised, and disgusted. Each emotion is expressed at two intensity levels (normal and strong), along with an additional neutral expression [7].

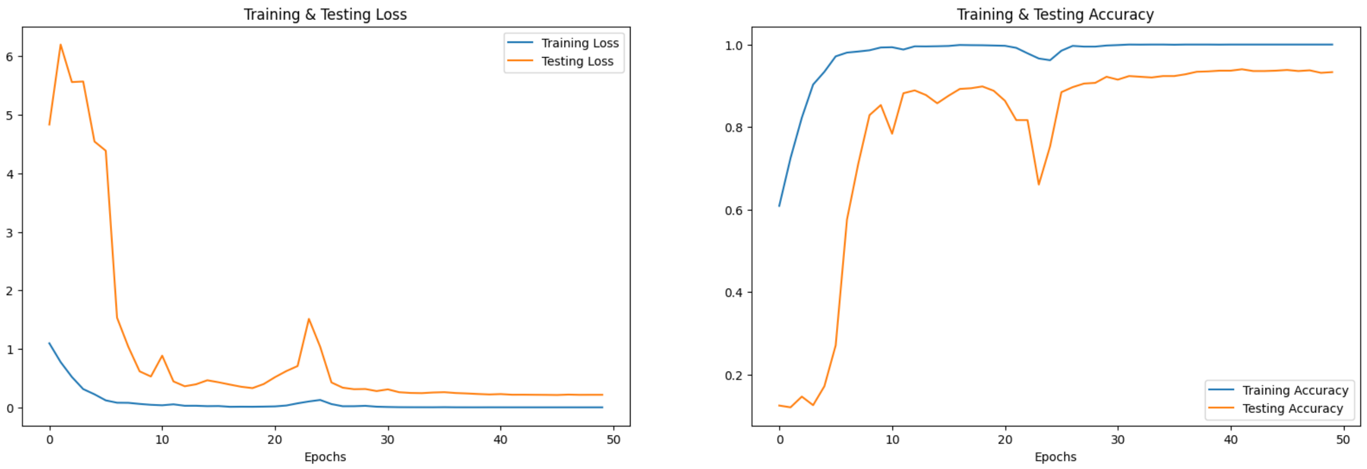


Fig. 4. Training and Testing Loss and Accuracy of the SER Models

TABLE I
QUANTITATIVE EVALUATION RESULTS

	Precision	*Recall*	*F1-score*	*Support*
Angry	0.95	0.96	0.95	171
Calm	0.95	0.95	0.95	145
Disgust	0.97	0.96	0.96	150
Fearful	0.93	0.89	0.91	158
Happy	0.90	0.93	0.92	142
Neutral	0.90	0.86	0.88	81
Sad	0.92	0.91	0.91	152
Surprised	0.93	0.97	0.95	153
Accuracy			0.93	1152
Macro_Avg	0.93	0.93	0.93	1152

Training Process. In the left plot in Fig. 4, the training loss starts at a high value and steadily decreases, indicating that the model is learning effectively. Meanwhile, the testing loss exhibits some initial fluctuations before eventually stabilizes. In the right plot, the training accuracy progressively increases throughout the training process, approaching 1.0 by the end. The testing accuracy displays a similar upward trend but experiences minor fluctuations, suggesting the possibility of slight overfitting.

B. Evaluation

Quantitative Evaluation. Fig. 3 and Table I present the quantitative evaluation results for the RAVDESS dataset. The precision for all labeled emotions (e.g., *Angry*, *Calm*, *Fearful*, etc.) exceeds 90%. With the exception of *Neutral* emotion, all other emotions achieve an F1-score above 90%. Quantitatively, the *Neutral* emotion demonstrates the lowest performance among the emotions. A possible explanation for this is the limited number of *Neutral* emotion samples—only 81 audio files were available for training, compared to over 140 samples for each of the other emotions.

Actual Sentences. To evaluate our model in real-world scenarios, we tested several sentences expressing different emotions in both English and Chinese. Table II provides examples of sentences categorized under *Neutral*, *Calm*, and *Sad* emotions, along with the corresponding responses generated by *EmoLingo*.

IV. DISCUSSION

The results demonstrate the feasibility of an emotion-aware conversational agent in providing personalized emotional support. The high precision and recall rates across most emotion categories—particularly for *Angry* (0.95), *Calm* (0.95), and *Disgust* (0.97)—suggest that the CNN-based Speech Emotion Recognition model effectively captures emotional nuances in speech. However, the relatively lower performance for the *Neutral* emotion (F1-score of 0.88) highlights the challenges in detecting subtle emotional states, likely due to the limited training data available for this category.

Testing with real-world sentences in multiple languages demonstrated the system's practical applicability. The chatbot exhibited appropriate emotional awareness and contextual understanding in its responses, as illustrated by the examples in Table II. For instance, when presented with emotionally sad content, the system generated empathetic and supportive replies while maintaining conversation coherence. These results validate the effectiveness of integrating Speech Emotion Recognition (SER) with GPT-based response generation to create meaningful emotional support interactions.

The current implementation does face certain limitations. The system's performance may be influenced by environmental noise, accent variations, and the complexity of real-world emotional expressions, which often involve mixed or transitioning emotions. Furthermore, although the RAVDESS dataset provided a solid foundation for training, its controlled nature may not fully represent the diversity of emotional expressions encountered in real-world applications, particularly across different cultural and linguistic contexts.

Despite these challenges, our results indicate that AI-driven emotional support systems can serve as valuable complementary tools in educational settings. The integration of real-time emotion recognition and context-aware response gener-

TABLE II
TESTING ACTUAL SENTENCES FOR DIFFERENT EMOTIONS

Emotion	User Input	EmoLingo Reply
Neutral	The weather is quite nice today.	That sounds wonderful, I’m glad you’re enjoying the nice weather today.
Neutral	今天的天氣還不錯。	是的，今天的天氣確實很宜人，希望你能好好享受這美好的一天。
Calm	We can take our time; there’s no rush.	I appreciate your laid-back approach, it’s great to take things at your own pace.
Calm	我們可以慢慢來，不用急。	沒問題，慢慢來，一切都會好的。
Sad	I really don’t know how to deal with all of this.	I’m sorry to hear that you’re feeling overwhelmed, it’s okay to take a step back and give yourself some time to process everything.
Sad	我真的不知道該怎麼面對這一切。	我了解，有時候事情讓人感到很困惑，但是你並不孤單。

ation provides a scalable solution for delivering immediate emotional support, particularly in scenarios where traditional counseling resources are limited or inaccessible.

V. CONCLUSION

In conclusion, this paper presents an integrated system for real-time emotional support that combines STT, SER, GPT-driven text generation, and TTS. The results demonstrate that leveraging CNN-based SER models and large language models significantly enhances the responsiveness and empathetic quality of the chatbot’s interactions. Although certain limitations—such as performance in noisy environments and the challenges of detecting subtle emotions—remain, our approach highlights the potential of AI-driven conversational agents in enhancing mental health support.

VI. FUTURE WORK

In this paper, we propose a chatbot architecture that integrates contemporary advancements in AI and Large Language Models (LLMs). However, several challenges persist and should be addressed in future work.

- **Enhancing Robustness to User Input:** We aim to overcome current limitations—such as performance degradation in noisy environments and difficulties in detecting subtle emotions—by exploring more robust and real-time audio preprocessing and noise-reduction techniques.
- **Establishing a Suitable Evaluation Metric:** Evaluating the proposed pipeline poses a challenge due to the absence of standardized datasets specifically designed for this task. To address this limitation, we utilize self-recorded speech data in multiple languages, tones, and emotions to assess the model’s ability to accurately interpret inputs. While this approach provides insights to gauge the model’s performance, it underscores the need for a more robust and standardized evaluation metric.
- **Incorporating Temporal Dependencies for Emotional Transitions:** The current model processes input segments independently, limiting its ability to capture emotional transitions across consecutive audio frames due to the absence of temporal dependency analysis. To address this limitation, we propose integrating a CNN-LSTM architecture. The CNN will extract local acoustic features, such as MFCCs, while the LSTM layers will model sequential patterns over time. This hybrid approach aims to enhance the model’s ability to detect emotional transitions, represent mixed emotions, and improve its applicability in real-world scenarios where emotions often shift dynamically during conversations.
- **Advancing Generalization and Practical Integration:** To improve generalization, we will expand the training datasets to include diverse languages, accents, and emotional expressions, enabling the model to perform effectively in real-world scenarios. Comprehensive user studies will be conducted to evaluate the chatbot’s effectiveness in reducing stress and promoting emotional well-being over prolonged usage. Additional enhancements include personalizing responses through reinforcement learning, ensuring ethical data handling and privacy, and integrating the system with institutional mental health services and user interface (UI) to provide scalable, real-time support.
- **Provide More Humanized Speech Response:** To make the chatbot’s responses more natural and emotionally engaging, we plan to propose enhancing the speech synthesis system with more human-like characteristics. This includes generating replies with emotional tones aligned to the detected user emotions, such as calming tones during stress or empathetic tones for sadness.

DATA AND CODE AVAILABILITY

Our data and code are available on GitHub: **https://github.com/KhaiLiTay/EmoLingo**

AUTHOR CONTRIBUTION STATEMENTS

- Tay Khai Li 戴凱麗 (20%): study design, SER model, overall model integration, data interpretation, PPT, leader.
- Liong Zheng Ee 農政宇 (20%): study design, data analysis, demo video, final presentation, help with every issue in each section.
- Michael Andrew Sucahyo 謝嘉銘 (20%): study design, SER model, data collection, proposal video, documentation.
- Wong Zi Ying 王籽穎 (20%): study design, STT model, GPT API stream, report.
- Tsung-Lun Shih 施淙綸 (20%): study design, TTS model, report.

REFERENCES

[1] M. Abdul Hamed, *"Speech emotion recognition 97.25% accuracy," Kaggle*, Jul. 01, 2023. [Online]. Available: https://www.kaggle.com/code/mostafaabdlhamed/speech-emotion-recognition-97-25-accuracy/notebook

[2] Coqui-Ai, *"A deep learning toolkit for Text-to-Speech, battle-tested in research and production," GitHub*, 2023. [Online]. Available: https://github.com/coqui-ai/TTS

[3] OpenAI, *"Robust Speech Recognition via Large-Scale Weak Supervision," GitHub*, 2023. [Online]. Available: https://github.com/openai/whisper

[4] Chatanywhere, *"Free ChatGPT API Key," GitHub*, 2023. [Online]. Available: https://github.com/chatanywhere/GPT_API_free/tree/main

[5] R. Nicole, *"Title of paper with only first word capitalized," J. Name Stand. Abbrev.*, in press.

[6] K. Chaovavanich, *"Record audio in Colab using getUser-Media(audio: true)," Gist*, Jul. 2023. [Online]. Available: https://gist.github.com/korakot/c21c3476c024ad6d56d5f48b0bca92be ?permalink_comment_id=3691958

[7] *"RAVDESS Emotional Speech Audio," Kaggle*, Jan. 19, 2019. [Online]. Available: https://www.kaggle.com/datasets/uwrfkaggler/ravdess-emotional-speech-audio