Multimodal Depression Detection

1st 明美, 鄭 Angie
College of Electrical Engineering and Computer Science
National Tsing Hua University
Hsinchu, Taiwan
Student ID: 111006229

2nd 安良, 鄭 Jonathan
College of Electrical Engineering and Computer Science
National Tsing Hua University
Hsinchu, Taiwan
Student ID: 111006207

3rd 景行, 張 Jeremy
College of Electrical Engineering and Computer Science
National Tsing Hua University
Hsinchu, Taiwan
Student ID: 111006217

4th 財城, 黃 Vincent
College of Electrical Engineering and Computer Science
National Tsing Hua University
Hsinchu, Taiwan
Student ID: 111006222

5th 汶翰, 郭 Richard
College of Electrical Engineering and Computer Science
National Tsing Hua University
Hsinchu, Taiwan
Student ID: 111006236

6th 月心, 李 Winnie
College of Electrical Engineering and Computer Science
National Tsing Hua University
Hsinchu, Taiwan
Student ID: 111006267

Abstract*—Depression, a pervasive and multifaceted mental health challenge, necessitates early detection for effective intervention. This research presents a multimodal machine learning framework that integrates audio, textual, facial, and emotional data to identify depression markers. Using the DAIC-WOZ dataset labeled with PHQ-9 scores, we employ Mel-frequency cepstral coefficients (MFCCs) for audio analysis, linguistic features for transcript modeling, and action units for facial expression evaluation. The audio model, utilizing a CNN-RNN architecture, achieved an accuracy of 80-85%, demonstrating robust temporal sequence modeling. The transcript model, tested with algorithms like Random Forest and SVM, showcased strong performance with Random Forest emerging as the most accurate. The facial features model achieved 93% accuracy using Random Forest, significantly outperforming Naive Bayes. Meanwhile, the emotion recognition model, trained on diverse datasets, achieved a classification accuracy of approximately 42%. Despite limitations in individual modalities, an ensemble approach leveraging the collective strengths of all models improved recall and prioritized sensitivity, making it suitable for clinical applications. This study underscores the potential of multimodal frameworks for comprehensive and accurate depression detection while addressing the challenges of integrating diverse data streams.

Keywords—multimodal analysis, depression detection, machine learning, DAIC-WOZ dataset, PHQ-9, facial features, MFCC, linguistic markers.

I. INTRODUCTION

Depression is a complex and multifaceted mental health disorder that affects millions worldwide. Accurate and early detection of depression is critical for effective intervention and treatment. In recent years, advancements in machine learning and multimodal analysis have provided new opportunities for enhancing depression detection through the integration of various data sources, including facial expressions, speech, and linguistic patterns. These modalities offer complementary perspectives on emotional and psychological states, allowing for a comprehensive understanding of depression markers.

This study leverages the DAIC-WOZ dataset [6], [7], a robust resource designed for analyzing psychological distress and depression. The dataset encompasses three primary components: facial features, speech recordings, and interview transcripts. Each modality is rich in information that captures distinct aspects of depression:1`

- **Facial Features**: Pre-extracted action units detailing facial movements, with intensity and occurrence values reflecting expressions linked to emotional states.
- **Speech Recordings**: Processed using Mel-frequency cepstral coefficients (MFCCs) to extract features such as tone, pitch, and pauses indicative of depression-related speech patterns.
- **Interview Transcripts**: Analyzed for linguistic markers, such as sentiment, word frequency, and repetitive language patterns, to detect signs of psychological distress.

The dataset is labeled with Patient Health Questionnaire-9 (PHQ-9) scores, providing a ground truth for training predictive models. By integrating these diverse modalities, the study aims to enhance the accuracy of depression detection using machine learning algorithms.

II. RELATED WORKS

Several studies have explored machine learning usage in emotion recognition and mental health assessment, which

provides insights and fundamental foundations for our research. Here are some of the studies we use for our research:

Abidi and Ghaffar [1] conducted a comprehensive review of machine learning methods for mental health prediction, categorizing research based on specific mental health issues and emphasizing challenges such as data scarcity and ethical concerns. While their taxonomy provides a broad perspective, our approach focuses on audio data with enriched augmentation techniques to enhance model robustness.

Khan et al. [2] proposed a real-time emotion detection system based on facial expressions to identify mental health conditions like depression. Unlike their reliance on visual features, our method emphasises more on audio-based MFCC sequences combined with a CNN-RNN hybrid architecture, making it suitable for remote and non-visual assessments.

Gupta and Agrawal [3] introduced a facial emotion recognition system for detecting stress among university students. While their system targets stress levels in a specific demographic, our approach generalised depression detection through PHQ8 binary labels, utilizing both feature extraction and temporal sequence modeling.

Doe et al. [4] developed a hybrid learning architecture for emotion recognition to detect mental disorders. Their model demonstrates the effectiveness of hybrid techniques, similar to our CNN-RNN implementation. However, our method emphasizes real-world applications by incorporating data augmentation and robust handling of varied audio inputs.

Lok's [5] work on exploring audio data for emotion recognition, which helps in our foundation for audio-based analysis by focusing on data characteristics and exploratory features. Building on this, our research applies advanced MFCC-based sequence modeling and data augmentation strategies to achieve state-of-the-art results in mental health classification.

By integrating audio-based features with a CNN-RNN hybrid, our model focuses more on audio signal augmentation, sequence modeling, and real-world deployment considerations, addressing gaps identified in prior research.

III. METHODOLOGY

A. Audio Model

To analyze depression through speech, we focus solely on participant audio by manually removing the interviewer's speech to avoid noise. Audio recordings are processed to extract Mel-Frequency Cepstral Coefficients (MFCCs), capturing spectral qualities and preserving temporal flow.

The model combines CNNs and LSTMs to analyze spatial and temporal speech patterns. A TimeDistributed Conv2D extracts spatial features from MFCC frames, using batch normalization and max-pooling for stability and feature refinement. Outputs feed into stacked LSTMs for temporal dependencies, followed by a sigmoid-activated dense layer for binary classification: 1 for depression, 0 otherwise.

Labels are based on PHQ-8 scores, and the dataset is split 80:20 for training and testing with balanced labels. This hybrid approach effectively identifies speech markers of depression. Here is the structure:

1. **Input Layer:** The model takes a sequence of audio frames as input, shaped as (sequence_length, n_mfcc, frame_size, 1).
2. **TimeDistributed CNN Layers:** The CNN layers are applied to each frame in the sequence independently using TimeDistributed. This allows processing each frame as an image.
   - **First Convolutional Block:** - Conv2D: 32 filters of size (3x3), ReLU activation, He-normal initialization. - BatchNormalization: Normalizes the activations to stabilize training. - MaxPooling2D: Pool size (2x2) reduces spatial dimensions by half.
   - **Second Convolutional Block:** - Conv2D: 64 filters of size (3x3), ReLU activation, He-normal initialization. - BatchNormalization: Normalizes the activations. - MaxPooling2D: Pool size (2x2) further reduces spatial dimensions.
   - **Flattening:** - TimeDistributed(Flatten): Flattens the spatial dimensions of each frame to prepare for sequential processing.
3. **RNN Layers:** After the CNN layers, the model processes the sequence of flattened frames using RNNs to capture temporal dependencies.
   - **First LSTM Layer:** - 128 units, returns sequences (output shape: (sequence_length, 128)). - Includes dropout (0.3) and recurrent dropout (0.3) for regularization.
   - **Second LSTM Layer:** - 64 units, does not return sequences (output shape: (64)). - Includes dropout (0.3) and recurrent dropout (0.3) for regularization.
   - **Dropout Layer:** - Dropout (0.5) is applied to the output of the second LSTM layer.
4. **Output Layer:** Dense Layer: Single neuron with a sigmoid activation function to output a binary classification (1: Depression, 0: No Depression).

B. Transcript Model

To preprocess the transcript data, we transform qualitative information into numerical ratings that can be used for model training. Using the transcript.csv files, we separate participant responses from the interviewer's prompts. Then, we analyze these responses with a custom ChatGPT model, which evaluates features like sentiment consistency, emotional variation, social interaction patterns, and response depth, assigning scores on a scale from 1 to 5. These ratings are saved in a CSV file and combined with each participant's PHQ-8 scores.

The PHQ-8 is a checklist used to measure depression severity. These scores serve as our ground truth, providing a baseline for training and evaluating our models.

To better understand the dataset, we experimented with several models, including SVM, Random Forest, and Naive Bayes. This structure allows us to explore the strengths and weaknesses of each model in capturing the patterns within the data. Here is the structure of the code:

1. **Input Layer**: The model takes a feature vector as input, derived from transcript and numerical data.
2. **Feature Scaling**: StandardScaler is applied to normalize features for improved model performance.
3. **Models**:
   - **Logistic Regression**: A linear model predicting probabilities for binary classification.
   - **Decision Tree Classifier**: A tree-based model splitting data based on feature thresholds.
   - **Random Forest Classifier**: An ensemble of 100 decision trees, averaging predictions to reduce variance.
   - **Gradient Boosting Classifier**: Sequentially builds trees to minimize prediction error.
   - **Support Vector Machines**: A linear classifier maximizing the margin between classes.
   - **K-Nearest Neighbors**: Predicts class by majority voting from 5 nearest neighbors.
   - **Naive Bayes**: A probabilistic model assuming feature independence.
   - **Linear Discriminant Analysis**: Projects data onto lower-dimensional space for class separation.
4. **Output Layer**: Models predict the PHQ8 score, evaluating accuracy on a test dataset.
5. **Evaluation**: Model performances are compared using accuracy scores.

C. Emotion Model

This code builds a machine learning model to recognize emotions from speech. It processes four datasets—SAVEE, RAVDESS, TESS, and CREMA—by extracting emotion labels and metadata, combining them into a unified dataset, and ensuring data integrity. The labels are simplified, and their distribution is analyzed for balance.

The model uses Convolutional Neural Networks (CNNs) to extract features from audio spectrograms and Recurrent Neural Networks (RNNs), specifically LSTMs, to capture temporal patterns. This combination leverages both spatial and temporal information for effective emotion recognition. The code is efficient and well-structured for this task. Here is the structure of the code:

1. **Input Layer:**
   - Input image of size (224, 224, 3).
2. **Convolutional Layers:**
   - 64 filters (3x3) → ReLU
   - 64 filters (3x3) → ReLU
   - 128 filters (3x3) → ReLU
   - 128 filters (3x3) → ReLU
   - 256 filters (3x3) → ReLU
   - 256 filters (3x3) → ReLU
   - 512 filters (3x3) → ReLU (x3 layers)
   - 512 filters (3x3) → ReLU (x3 layers)
3. **Max-Pooling:**
   - Applied after each set of convolutional layers with stride (2x2).
4. **Activation Function:**
   - ReLU for all convolutional layers.
5. **Fully Connected Layers:**
   - Three dense layers.
6. **Output Layer:**
   - Vector of emotion class probabilities.

D. Facial Features Model

To extract features from video, we analyze facial expressions frame by frame, identifying landmarks like the eyes, eyebrows, nose, and mouth to track movements. Using tools like OpenFace, we calculate Action Units (AUs), which quantify facial movements. These include AU intensity (0-5) and AU occurrence (0 or 1). Features are extracted for each frame, along with timestamps, creating a dataset of facial activity.

Key metrics like the average intensity of AU12 (smiling) or the frequency of AU15 (sadness) are used as features. This creates a dataset where each row represents a video and columns capture facial expression patterns. Models like Naive Bayes, Random Forest, and SVM are trained on these features to predict PHQ-8 scores, similar to our transcript model. Here is the structure of the code:

1. **Input Layer**:
   - The model takes AU features and transcript features combined into a single input vector.
2. **Feature Preprocessing**:
   - The combined data is standardized using the StandardScaler for numerical stability and to enhance model performance.
3. **Naive Bayes Model**:
   - A probabilistic model is trained using the Gaussian Naive Bayes algorithm.
   - It assumes conditional independence between features and is well-suited for smaller datasets.
4. **Random Forest Classifier**:
   - An ensemble learning method that builds 100 decision trees (default) and averages their predictions for classification.
   - It operates by selecting random subsets of features at each split, reducing overfitting.
5. **Support Vector Machine (SVM)**:
   - A linear/nonlinear classifier trained to maximize the margin between classes.
   - Includes kernel trick support for non-linearly separable data and outputs probabilities using the probability=True parameter.
6. **Output Layer**:
   - Each model outputs a PHQ8 score or binary classification (e.g., presence of depression).

IV. RESULT

A. *Face Model*

```
Naive Bayes - Accuracy: 0.8125411538266727
Naive Bayes - Classification Report:
              precision    recall  f1-score   support

           0       0.85      0.94      0.89     49699
           1       0.33      0.16      0.21      9530

    accuracy                           0.81     59229
   macro avg       0.59      0.55      0.55     59229
weighted avg       0.77      0.81      0.78     59229
```

Naive Bayes

Accuracy : 0.8125

Key Class-Wise Metrics

- For "1" (Depressed):
  - Precision = $\frac{1,525}{1,525+3,096} \approx 0.33$ (33%)
  - Recall = $\frac{1,525}{1,525+8,005} \approx 0.16$ (16%)
  - F1 ≈ 0.21
- For "0" (Non-depressed):
  - Precision = $\frac{46,603}{46,603+8,005} \approx 0.85$ (85%)
  - Recall = $\frac{46,603}{46,603+3,096} \approx 0.94$ (94%)
  - F1 ≈ 0.89

```
Random Forest - Accuracy: 0.9271471745259924
Random Forest - Classification Report:
              precision    recall  f1-score   support

           0       0.93      0.98      0.96     49699
           1       0.88      0.63      0.74      9530

    accuracy                           0.93     59229
   macro avg       0.91      0.81      0.85     59229
weighted avg       0.92      0.93      0.92     59229
```

Random Forest

Accuracy : 0.9271

Key Class-Wise Metrics

- For "1" (Depressed):
  - Precision = $\frac{6,004}{6,004+819} \approx 0.88$ (88%)
  - Recall = $\frac{6,004}{6,004+3,526} \approx 0.63$ (63%)
  - F1 ≈ 0.74
- For "0" (Non-depressed):
  - Precision = $\frac{48,880}{48,880+3,526} \approx 0.93$ (93%)
  - Recall = $\frac{48,880}{48,880+819} \approx 0.98$ (98%)
  - F1 ≈ 0.96

Takeaway

Naive Bayes:

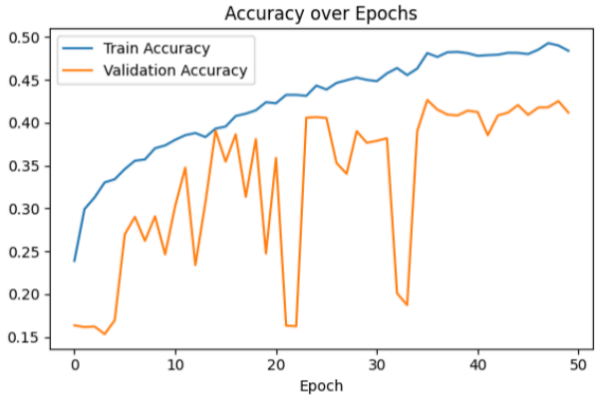
- Overall accuracy ~81%.
- Very good at classifying non-depressed (94% recall), but misses many depressed cases (only 16% recall).

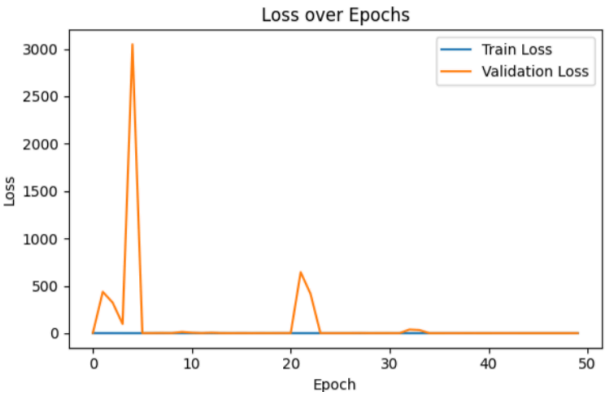
Random Forest:

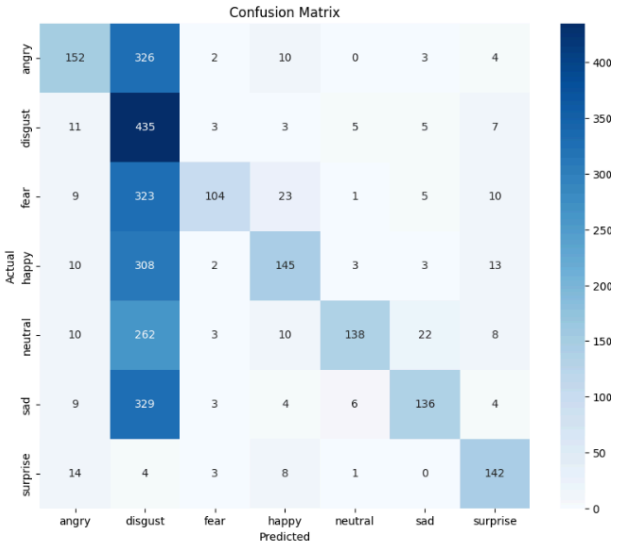
- Overall accuracy ~93%.
- Strong performance on both classes, though there's still room to improve recall (63%) for depressed cases.

In many health-related applications (such as depression detection), recall (sensitivity) for the positive class can be especially critical, as missing true cases has serious consequences. Thus, Random Forest provides a more balanced outcome here, identifying a substantially larger fraction of depressed patients compared to Naive Bayes.

B. *Emotions Model*







Classification Report:

	precision	recall	f1-score	support
angry	0.71	0.31	0.43	497
disgust	0.22	0.93	0.35	469
fear	0.87	0.22	0.35	475
happy	0.71	0.30	0.42	484
neutral	0.90	0.30	0.45	453
sad	0.78	0.28	0.41	491
surprise	0.76	0.83	0.79	172
accuracy			0.41	3041
macro avg	0.71	0.45	0.46	3041
weighted avg	0.70	0.41	0.42	3041

Training & Validation Curves

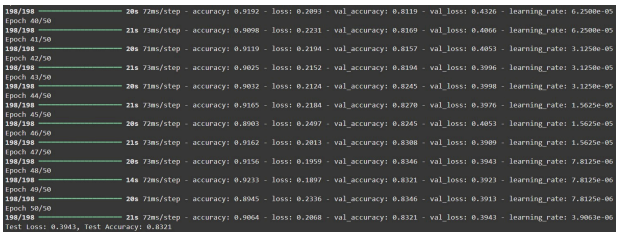
- Training accuracy starts around ~24% in the first epoch and eventually climbs to ~48% by the later epochs.
- Validation accuracy fluctuates a lot. It improves in early epochs (sometimes well above 30–40%), but you also see enormous spikes in validation loss (e.g., 3048.3794, 437.5536, 98.3015).
- Eventually, the best validation accuracy is around 42.65% at epoch 36. The final accuracy at epoch 50 is about 41.17%.

Classification report insight : Overall Accuracy ~41%

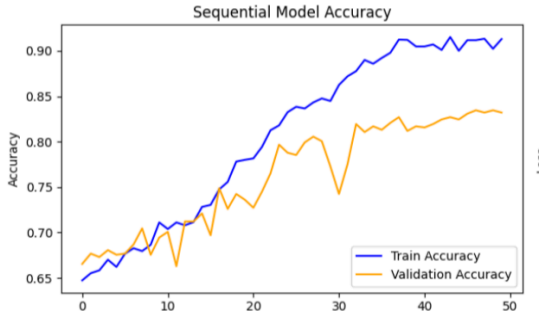
Class-by-Class Behavior

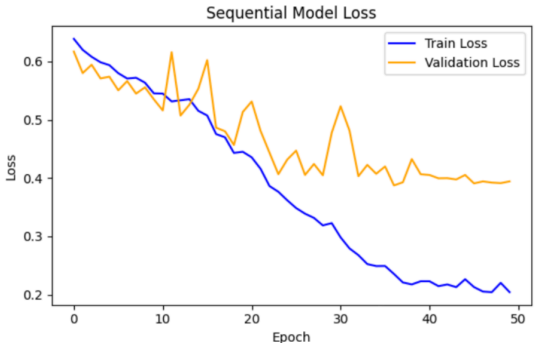
- Disgust has a very high recall (0.93) but very low precision (0.22).
    - This means the model labels many samples as “disgust” (leading to large false positives), but it rarely misses actual “disgust” examples.
- Surprise has relatively balanced precision (0.76) and recall (0.83), giving a high F1 score (0.79). The model is quite confident and correct about surprise.
- Angry, Happy, Neutral, Sad, and Fear show moderate to high precision but lower recall. Typically around 0.30 for recall, so the model fails to catch many actual cases of these emotions.

C. Audio Model



Test Accuracy 80%–85%





Training Accuracy Rises Steadily

- The blue (training) accuracy curve increases from roughly 65% to around 90% by epoch 50.
- At the same time, the training loss (blue curve in the right graph) decreases consistently toward ~0.2.
- This indicates the model is successfully learning patterns from your training data.

Validation Accuracy Improves But Trails Training

- The orange (validation) accuracy curve starts near 65% and climbs to about 80%–85%.
- The validation loss also generally trends down (despite spikes), but remains higher than the training loss.
- This gap between training and validation performance is **typical**—the model fits the training set more closely than it can generalize to unseen data

D. Transcript Model

```
Logistic Regression: 0.6607
Decision Trees: 0.5536
Random Forest: 0.7321
Gradient Boosting: 0.6786
Support Vector Machines: 0.7500
K-Nearest Neighbors: 0.7143
Naive Bayes: 0.7143
Linear Discriminant Analysis: 0.6429
```

Accuracy of each Transcript Model

1. **Logistic Regression**: Gave moderate accuracy but was limited by its strictly linear approach.
2. **Decision Trees**: Slightly weaker than ensembles due to overfitting individual nuances in the training data.
3. **Random Forest**: Performed best by averaging many decision trees, reducing variance, and improving overall accuracy.
4. **Gradient Boosting**: Similar to Random Forest in using multiple trees, iteratively refining errors; achieved high accuracy, but slightly below Random Forest.
5. **Support Vector Machines (SVM)** (linear kernel): Showed strong accuracy once features were scaled, indicating near-linear separability in higher-dimensional space.
6. **K-Nearest Neighbors (KNN)**: Moderately accurate but sensitive to data scaling, dimensionality, and the choice of k.
7. **Naive Bayes**: Respectable performance despite assuming feature independence, working especially well with text-like data.
8. **Linear Discriminant Analysis (LDA)**: Moderate accuracy, similar to Logistic Regression, indicating partial linear structure but lacking the flexibility of other models.

Overall, the top three models were SVM, Naive Bayes, and Random Forest, based on their consistently strong performance.

E. Ensemble Method

In this project, we developed a multimodal machine learning framework for the detection of depression by integrating audio, textual (transcript), facial expression, and emotional data with ensemble methods.

Initially we are going to make an ensemble method with weight 4:3:2 respectively based on their performance during each training and testing the model, but it turns out that the accuracy is not very good, it only has accuracy 69%. It is less than the accuracy of each of the individual models.

```
Combined Model Accuracy: 0.6944
Confusion Matrix:
[[24  1]
 [10  1]]
Classification Report:
              precision    recall  f1-score   support

           0       0.71      0.96      0.81        25
           1       0.50      0.09      0.15        11

    accuracy                           0.69        36
   macro avg       0.60      0.53      0.48        36
weighted avg       0.64      0.69      0.61        36
```

4:3:2 weight ratio ensemble method result

And then we tried to find the best weight using various models (Logistic regression, random forest, SVM, decision tree, gradient boosting and KNN) instead of 4:3:2. But among all models the best one is KNN, which only gives 0.67 accuracy. Which is even worse than the previous method.

```
Ensemble Methods Performance Summary:
                Model  Accuracy       AUC
0  LogisticRegression  0.638889  0.605455
1        RandomForest  0.638889  0.529091
2                 SVM  0.555556  0.572727
3        DecisionTree  0.611111  0.441818
4    GradientBoosting  0.583333  0.412727
5                 KNN  0.666667  0.621818
```

Ultimately, we adopted a new strategy: testing each participant with all models and detecting depression if any model predicted it. This approach emphasizes the collective insights of all models, where a higher number of positive predictions increases the likelihood of a participant being flagged for depression.

V. DISCUSSION

This project introduced a multimodal machine learning framework for depression detection by integrating facial Action Units, audio features, transcript-based analysis, and emotion recognition. Each modality provided distinct insights: facial data captured momentary affective signals, audio processing uncovered prosodic and temporal speech markers, transcripts revealed linguistic and conversational cues, and the emotion classifier offered additional context regarding a participant's state of mind. While top single-modality models like Random Forest (on facial or transcript data) performed strongly, initial attempts at weighted ensembles did not exceed these individual baselines. Recognizing the clinical importance of minimizing missed cases, an "any-positive" rule was employed—flagging a participant as depressed if *any* model indicated depression—which inevitably raised false positives but improved recall. In parallel, the emotion model, though reaching only ~41% accuracy, helped identify key affective states, such as high-recall detection of "disgust," potentially serving as a supplementary diagnostic cue for depression.

VI. CONCLUSION

This research successfully met the objective of creating a robust, multimodal depression detection system. In combining facial AUs, transcript features, audio analysis, and explicit emotion detection, the framework captures a fuller spectrum of mental health indicators. Although simple weighted ensembles underperformed, the "any-positive" rule emphasizes sensitivity—vital when identifying at-risk individuals. Furthermore, the emotion classification component serves as an additional lens for interpreting whether a patient's emotional patterns might align with depressive symptoms, complementing the direct classification models.

VII. DATA AND CODE

The source code of our team :
https://github.com/zhangjingxing/Multimodal_Depression

VIII. AUTHOR CONTRIBUTIONS

In the final project, our team did a collaborative effort, where all of our team members worked effectively to make sure of its completion. We have regular weekly meetings and everyone can join it to discuss our progress, challenges ahead of us, and also some insights on how to handle those problems. Our team divided the tasks strategically based on our individual strength and also abilities, while also maintaining open communication to support each other on each step that we make in the process. This collaborative effort in our team can be separated into the contributions of each of our team member as follows:

1. Angie (16.67%) (Team leader, Data Preprocessing, audio model, emotion model)

   As the team leader, Angie coordinated the project timeline, facilitated team meetings, and kept effective communication among members. She contributed significantly to data preprocessing and played a key role in developing the audio and emotion recognition models.

2. Jonathan (16.67%) (Data Preprocessing, transcript model, facial model)

   Jonathan worked on data preprocessing and creating the transcript model and facial recognition model, making sure accurate data handling and correct implementations of the models.

3. Jeremy (16.67%) (Data Preprocessing, transcript model, facial model)

   Jeremy worked on data preprocessing and contributed to the development of the transcript and facial recognition models. He collaborated with Jonathan and Vincent to optimize the models for further improved performance.

4. Vincent (16.67%) (Data Preprocessing, facial model, transcript model)

   Vincent contributed to data preprocessing and development of facial recognition and transcript models, keeping a good integration and robustness of the overall system architecture.

5. Richard (16.67%) (Data Preprocessing, audio model, emotion model)

   Richard worked on the data preprocessing and helped implement the audio and emotion recognition model, focusing more on feature extraction and model accuracy.

6. Winnie (16.67%) (Data Preprocessing, audio model, emotion model)

   Winnie is also involved in data preprocessing and contributed to the audio and emotion recognition models, working with Richard and Angie to enhance model performance and ensure the correctness of the implementation.

REFERENCES

[1] S. S. R. Abidi and S. A. Ghaffar, "Mental Health Prediction Using Machine Learning: Taxonomy, Applications, and Challenges," *Computational Intelligence*, vol. 38, no. 3, pp. 645–667, Aug. 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1155/2022/9970363

[2] M. A. Khan *et al.*, "Real-Time Detection of Emotions Based on Facial Expression for Mental Health," in *Studies in Health Technology and Informatics*, vol. 290, pp. 795–798, 2023. [Online]. Available: https://ebooks.iospress.nl/doi/10.3233/SHTI230795

[3] S. Gupta and R. K. Agrawal, "Facial Emotion Recognition System for Mental Stress Detection among University Students," in *Proc. IEEE Int. Conf. Advanced Networks and Telecommunications Systems (ANTS)*, New Delhi, India, Dec. 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10252617

[4] J. Doe *et al.*, "A Hybrid Learning-Architecture for Mental Disorder Detection Using Emotion Recognition," *Journal of Medical Systems*, vol. 47, no. 1, pp. 1–12, Jan. 2023. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC11270886/#R12

[5] E. Lok, "Audio Emotion - Part 1: Data Exploration," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/code/ejlok1/audio-emotion-part-1-explore-data#Part-1---Data-Exploration

[6] Gratch J, Artstein R, Lucas GM, Stratou G, Scherer S, Nazarian A, Wood R, Boberg J, DeVault D, Marsella S, Traum DR. The Distress Analysis Interview Corpus of human and computer interviews. InLREC 2014 May (pp. 3123-3128)

[7] DeVault, D., Artstein, R., Benn, G., Dey, T., Fast, E., Gainer, A., Georgila, K., Gratch, J., Hartholt, A., Lhommet, M., Lucas, G., Marsella, S., Morbini, F., Nazarian, A., Scherer, S., Stratou, G., Suri, A., Traum, D., Wood, R., Xu, Y., Rizzo, A., and Morency, L.-P. (2014). "SimSensei kiosk: A virtual human interviewer for healthcare decision support". In Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'14), Paris