Real-Time Feedback for Verbal and Non-Verbal Communication Skills in Mock Interviews

By:
L Gyan Rao Nazre (1NT22IS080)
Madhumitha R Aithal (1NT22IS087)

1. Data Collection

a. Datasets Used

- RAVDESS (Ryerson Audio-Visual Database of Emotional Speech and Song)
- AffectNet
- MIT Interview Dataset

b. Source Type

- RAVDESS and AffectNet are publicly available
- MIT Interview Dataset is provided by MIT

c. Data Description

- RAVDESS: Speech recordings labeled with emotions and vocal expressions
- AffectNet: Facial expressions with labeled emotions

- MIT Interview Dataset: Interview recordings with expert evaluations

d. Data Reliability

- RAVDESS: High quality
- AffectNet: Well-labeled real-world data
- MIT Interview Dataset: Expert-annotated
- The combination enables comprehensive communication analysis

---

2. Data Analysis and Processing

a. Preprocessing Techniques

Audio (RAVDESS, MIT):

- MFCC Extraction (Mel-Frequency Cepstral Coefficients)
- Pitch and Energy Analysis
- Spectrogram Generation

Video (AffectNet, MIT):

- Facial Expression Detection using Dlib or OpenFace
- Posture Analysis via OpenPose or MediaPipe

Text (MIT):

- Lowercasing and Tokenization
- Stopword Removal
- Lemmatization

Labels:

- Expert annotations for fluency, confidence, eye contact, speech clarity

b. Tools and Frameworks

- Programming Language: Python (Jupyter Notebooks)
- Audio Processing: Librosa, PyDub
- Video Processing: OpenCV, Dlib, MediaPipe
- Text Processing: NLTK, SpaCy
- Modeling: TensorFlow, Keras, PyTorch
- Evaluation: Scikit-learn

c. Handling Missing or Inconsistent Data

- Removed corrupted files or missing annotations

- Mean imputation for continuous metrics
- Z-score filtering for outlier removal

---

3. Modeling and Experimentation

a. Problem Statement

Create a system that provides real-time feedback on verbal and non-verbal communication during mock interviews—assessing speech clarity, tone, fluency, facial expressions, posture, and confidence.

b. Model Architecture

- Audio: CNN-LSTM for temporal speech features
- Video: CNN for expression and posture classification
- Text: Transformer-based models like BERT
- Fusion Layer: Combines all modalities using late fusion or attention
- Output: Scores for fluency, confidence, clarity, etc.

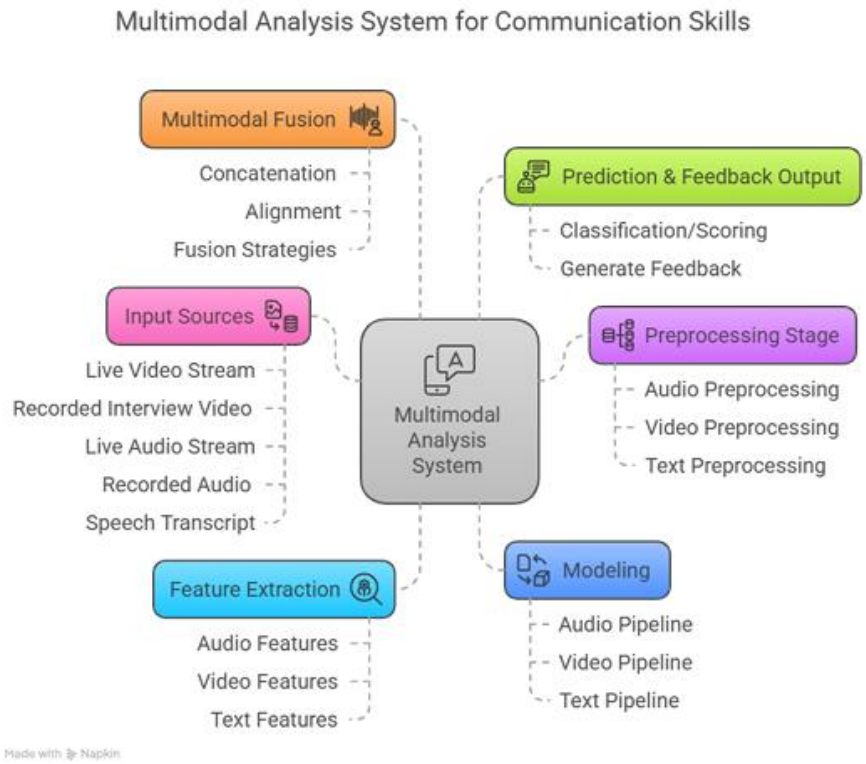
c. Training Details

- Data Split: 70% training / 15% validation / 15% testing
- Epochs: 50

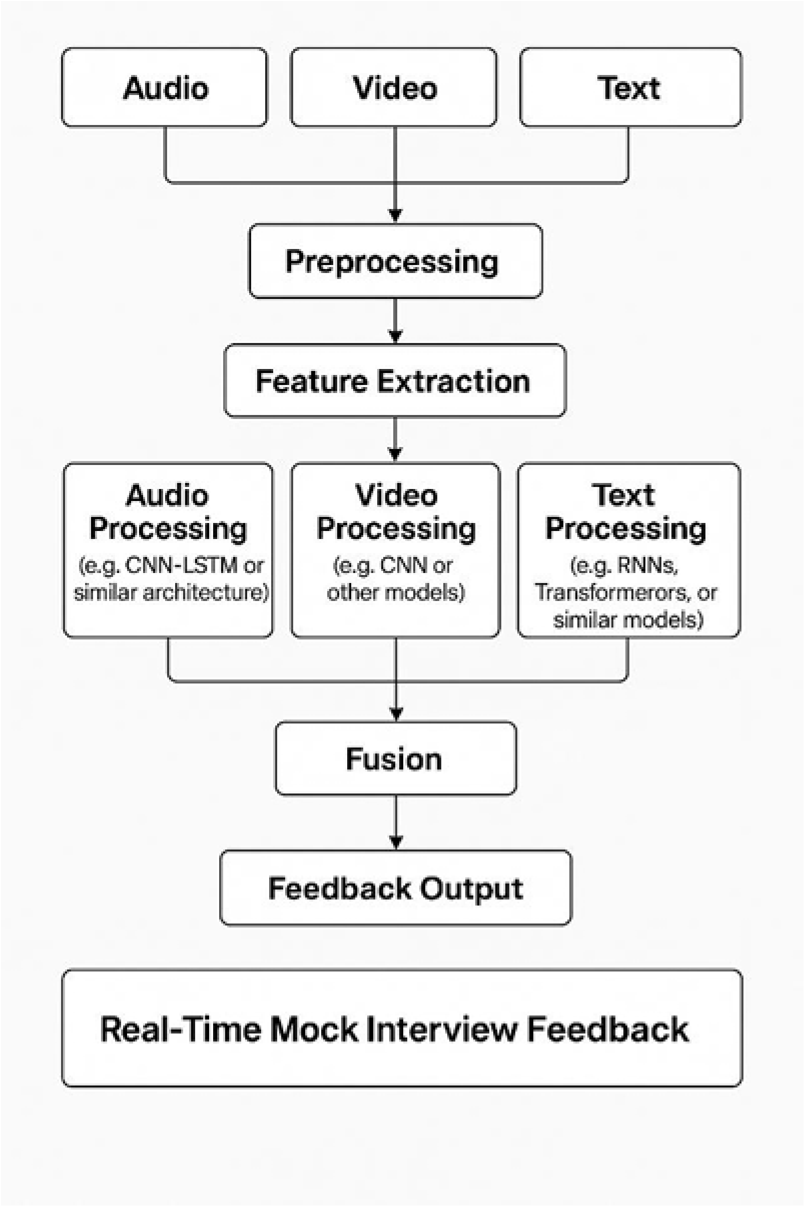
- Batch Size: 32
- Optimizer: Adam
- Loss Function: Categorical Cross-Entropy
- Regularization: Dropout (0.3), L2
- Metrics: Accuracy, Precision, Recall, F1-score

4. Visual Representation of Methodology

a. Data Flow Diagram



b. Architecture Diagram


Audio
Video
Text
Preprocessing
Feature Extraction
Audio Processing (e.g. CNN-LSTM or similar architecture)
Video Processing (e.g. CNN or other models)
Text Processing (e.g. RNNs, Transformerors, or similar models)
Fusion
Feedback Output
Real-Time Mock Interview Feedback

Summary

This project focuses on building a system that gives real-time feedback on how well a person communicates during mock interviews. It checks both verbal (how you speak) and non-verbal (your facial expressions, posture, and body language) communication.

To do this, the system uses three well-known datasets: RAVDESS (for speech), AffectNet (for facial expressions), and the MIT Interview Dataset (for real interviews with expert feedback).

The data is carefully processed—speech is turned into features like pitch and tone, video is used to analyze expressions and body language, and text is cleaned and simplified for analysis. The system uses machine learning models to understand this data: CNNs and LSTMs for audio, CNNs for video, and transformer models like BERT for text. All this information is combined to judge how confident, clear, and fluent a person is in their communication.

The final system helps users improve their interview skills by giving useful, automated feedback based on how they sound, look, and speak.