**Interview-Based Human Behavior Analysis with Machine Learning**

**Objective:**

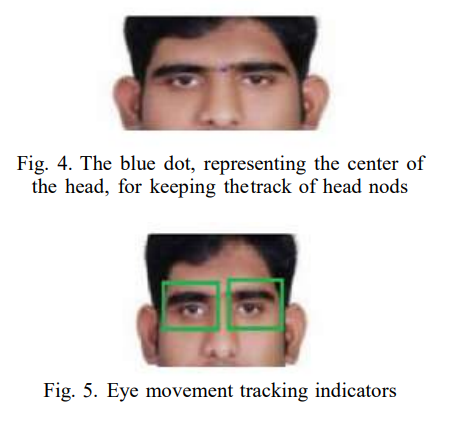
To analyze human behavior during interviews using machine learning by extracting and integrating audio, visual, and language cues.

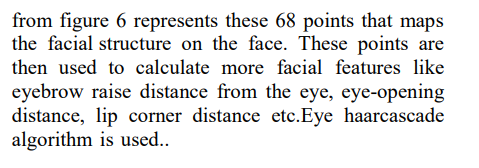
**Methodology:**

1. **Audio Cues: Prosodic Features**
   * **Speaker Identification**: Differentiate between interviewer and interviewee.
   * **Speech Segmentation**: Identify speech and non-speech segments in the audio.
   * **Noise Elimination**: Remove background noise for clearer audio analysis.
   * **Feature Extraction**:
     + **Pitch**: Measures the frequency of speech.
     + **Band Energy**: Analyzes the energy within specific frequency bands.
     + **Speech Rate**: Calculates the speed of spoken words.
   * **Tools Used**: PRAAT, PyAudio for audio processing and feature extraction.
2. **Visual Cues: Facial Features and Movements**
   * **Face Detection**: Implemented using Haar Cascade Classifier to identify faces in video frames.
   * **Facial Cues**:
     + **Expressions**: Detects emotions such as neutral, joy, sadness, contempt, anger, disgust, fear, etc.
     + **Head Nods**: Analyzes head movements to infer yes or no responses.
     + **Eye Gaze**: Tracks eye movements using Hidden Markov Model to understand attention and focus.
   * **Facial Landmarks**:
     + **Detection**: dlib library used to localize regions like eyes, eyebrows, nose, mouth, and jawline for detailed analysis.
3. **Language Cues: Sentiment Analysis**
   * **Sentiment Classification**: Uses VADER (Valence Aware Dictionary and sEntiment Reasoner) to categorize terms into positive and negative emotions.
4. **Predictive Model Development**:
   * **Feature Integration**: Combines features from facial expressions, audio cues, and verbal language to form a single joint feature vector.
   * **Model Training**: Machine learning models are trained on this joint vector to predict behavior and emotional states during the interview.
5. **Dataset**:
   * **Data Collection**: Manually curated dataset, which involves annotated interviews with labels for different behaviors and emotions.

**Key Tools and Libraries:**

* **PRAAT** and **PyAudio**: For extracting and analyzing prosodic features from audio.
* **Haar Cascade Classifier**: For face detection in video frames.
* **dlib**: For detecting facial landmarks and localizing facial regions.
* **Hidden Markov Model**: For modeling eye gaze and head movements.
* **VADER**: For sentiment analysis based on positive and negative emotion terms.





**2. Leveraging Multimodal Behavioral Analytics for Automated Job Interview Performance Assessment**

**Objective:**

To automate the assessment and feedback of job interview performance by analyzing multimodal data sources such as facial expressions, speech, prosodic information, and text transcripts from recorded interviews.

**Data Sources and Datasets:**

* **MIT Interview Dataset** (Naim et al., 2015):
  + 138 mock job interviews of 69 candidates, recorded pre- and post-intervention.
  + Includes Amazon Mechanical Turk Worker scores for various verbal and nonverbal cues, averaged across 9 workers for ground truth.
* **Audio Files**: Used for prosodic and lexical analysis.

**Methodology:**

1. **Multimodal Data Analysis**:
   * Utilizes video, audio, and text transcripts to construct a comprehensive representation of the interviewee's behavior and emotions.
2. **Facial Feature Extraction**:
   * **Landmarks**: Key points on the face (nose, chin, eyes, mouth corners) extracted using OpenCV.
   * **Head Pose Features**: Pitch, Roll, and Yaw calculated from rotation matrix elements.
   * **Smile Detection**: Pre-trained LeNet CNN model used on the SMILES dataset to detect smiling.
   * **Feature Averaging**: Features extracted from video frames (1-second intervals) and averaged over the video duration.
3. **Audio Feature Extraction (Prosodic Features)**:
   * **Features Extracted**: Frequency, pitch, tone, intensity, spectral energy, spectral centroid, zero-crossing rate.
   * **Time-Domain Features**: Calculated from raw audio signals.
   * **Frequency-Domain Features**: Derived using Discrete Fourier Transform (DFT) and Inverse DFT for cepstral domain features.
   * **Windowing**: Audio signals split into short-term windows (20ms to 100ms) for detailed feature extraction using pyAudioAnalysis and PRAAT.
4. **Lexical Feature Extraction (Text Analysis)**:
   * **Transcription**: Google Cloud Speech-to-Text API used to transcribe audio clips.
   * **Text Cleaning**: Lowercasing, removing punctuation, accents, and extra spaces.
   * **Tokenization**: Splitting text into words using Natural Language Toolkit (NLTK).
   * **Feature Extraction**: Average words per minute, unique words per minute, filler words, sentiment scores, named entities (nouns, adjectives, verbs) using Stanford NER.
   * **Sentiment Analysis**: Tone Analyzer used to classify emotions in text (Joy, Sadness, Tentative, Analytical, Fear, Anger).
5. **Feature Fusion and Machine Learning**:
   * **Composite Feature Vector**: Combining audio, video, and lexical features.
   * **Target Labels**: Ratings on a scale of 1-7 based on parameters like Eye Contact, Speaking Rate, Engagement, Pauses, Calmness, Authenticity, etc.
   * **Feature Selection**: Techniques like K-best selection, correlation matrices, and Benjamini-Hochberg procedure to retain the most predictive features and reduce redundancy.
   * **Model Training**: Lasso and SVR models trained using the fused feature vector to predict interview ratings. Hyperparameter tuning and 3-fold cross-validation used to optimize model performance.
6. **Performance Evaluation**:
   * **Parameters Assessed**: Eye Contact, Speaking Rate, Engagement, Calmness, etc.
   * **Feature Combination Analysis**: Various combinations of lexical, prosodic, and facial features tested to find optimal configurations for each parameter.
   * **Hyperparameter Tuning**: Extensive search over parameter values to avoid poor model performance.

**Key Tools and Libraries:**

* **SentiBank**: For visual concept detection and extracting adjective-noun pairs (ANPs).
* **pyAudioAnalysis**: For audio feature extraction.
* **OpenCV**: For facial landmark detection.
* **LeNet CNN**: For smile detection.
* **Natural Language Toolkit (NLTK)**: For text tokenization.
* **Tone Analyzer**: For emotion classification in text.
* **Stanford Named Entity Recognizer (NER)**: For identifying parts of speech in text.