Ain Shams University

Faculty of Computer and Information Sciences

ARTIFICIAL INTELLIGENCE PROGRAM

Pre-Recorded Video Interview Analysis

By

MOHAMED SAMY MOHAMED

MOHAMED ASHRAF MAHRAN

AMMAR MOHAMED HASSAN

YOMNA MOHAMED BASSAM

NADINE HAITHAM ALI

YOUSSEF TAMER MAHMOUD

Under Supervision of

DR. MONA ABDELAZIM

Assistant Professor in Information Systems Department,
Faculty of Computer and Information Sciences,
Ain Shams University

TA. AYA NASSER

Assistant Lecturer in Scientific Computing Department
Faculty of Computer and Information Sciences,
Ain Shams University

2024/2025

Acknowledgement

We would like to express our heartfelt gratitude to **Dr. Mona Abdelazim**, our project supervisor, for her continuous support, expert guidance, and thoughtful mentorship throughout every stage of this project. Her insights, encouragement, and constructive feedback played a vital role in shaping the direction and overall quality of our work.

We also extend our thanks to **TA. Aya Nasser**, our teaching assistant, for her assistance during the development process. Her clarifications and responses to our inquiries were helpful in resolving technical issues and ensuring steady progress.

Finally, we extend our appreciation to everyone who supported us academically or morally during this journey. This project would not have been possible without the contributions and encouragement we received, and we are deeply grateful for the experience and knowledge we gained throughout this endeavor.

Abstract

In response to the growing demand for scalable recruitment solutions, this project introduces **PRVIA**—a lightweight, AI-driven web application designed to automate the evaluation of pre-recorded video interviews. As modern organizations increasingly shift to virtual hiring practices, the need for tools that reduce human workload and maintain evaluation consistency has become critical. PRVIA addresses these needs by combining state-of-the-art machine learning models in a modular pipeline that analyzes both **verbal and non-verbal cues** from candidate-submitted video responses.

The system is capable of extracting and processing audio, facial expressions, and textual content from interview videos. Key modules include **speech-to-text transcription**, **English proficiency scoring**, **semantic relevance assessment using large language models (LLMs)**, **response summarization**, and **personality and expression analysis** through computer vision.

Final system testing demonstrated the platform's ability to generate comprehensive candidate reports within a total processing time of under 2.5 minutes per interview, providing detailed scores across language, personality, and content relevance dimensions. The platform's dual-user interface supports both job applicants and HR staff, simplifying application submission, job creation, and candidate review. By automating the most time-consuming phases of the recruitment cycle, PRVIA enhances efficiency and empowers organizations to make faster, more informed hiring decisions.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

- **AI** : Artificial Intelligence
- **APA**: Automatic Pronunciation Assessment
- **ASR**: Automatic Speech Recognition
- **ATS**: Applicant Tracking System
- **AUC:** Area Under the Receiver Operating Characteristic Curve
- **AffectNet**: Affective Facial Expression Dataset
- **BART**:Bidirectional and Auto-Regressive Transformers
- **BERT**: Bidirectional Encoder Representations from Transformers
- **BLEU**:Bilingual Evaluation Understudy
- **CAPT**: Computer-Assisted Pronunciation Training
- **CNN**: Convolutional Neural Network
- **3D CNN** : Three-Dimensional Convolutional Neural Network
- **FER-2013**: Facial Expression Recognition 2013 Dataset
- **FLAN-T5:**Fine-tuned Language Net T5 (Text-to-Text Transfer Transformer with instruction-tuning)
- **GELU**: Gaussian Error Linear Unit
- **GMM**: Gaussian Mixture Models
- **GOP**: Goodness of Pronunciation
- **HR:** Human Resources
- **I3D**: Inflated 3D Convolutional Network
- **LCS:** Longest Common Subsequence
- **LLM**: Large Language Model
- **LSTM**: Long Short-Term Memory
- **MSE**: Mean Squared Error
- **MTCNN**: Multi-task Cascaded Convolutional Networks
- **NLG**: Natural Language Generation

- **NLP**: Natural Language Processing
- **PCC** : Pearson Correlation Coefficient
- **RNN**: Recurrent Neural Network
- **ROC:** Receiver Operating Characteristic
- **ROUGE:** Recall-Oriented Understudy for Gisting Evaluation
- **SDK**: Software Development Kit
- **SVR:** Support Vector Regressor
- **WER:** Word Error Rate
- **X3D**: Expandable 3D Convolutional Neural Network
- **TF-IDF**: Term Frequency-Inverse Document Frequency

Chapter 1

Introduction

1.1 Motivation

In today's job market, several trends are shaping how companies approach hiring, particularly the increasing reliance on virtual platforms. As remote work becomes more common, businesses are moving away from traditional in-person interviews to more flexible methods, such as online and pre-recorded interviews. This shift allows organizations to expand their talent pools globally and adapt to the fast paced, remote nature of modern work environments.

The hiring process typically follows structured stages, starting with job postings and candidate screenings, followed by interviews and evaluations, and concluding with final decisions. Interviews are generally classified into three types: offline (in-person), live virtual (real-time), and pre-recorded. Pre-recorded interviews, where candidates respond to a set of predefined questions, have gained popularity as part of the initial stages of hiring. They enable companies to assess multiple candidates without the constraints of scheduling conflicts. However, manually evaluating these recordings can be time-consuming and is often susceptible to bias or discrimination.

Our project introduces an AI-powered system specifically designed to automate the assessment of pre-recorded video interviews, effectively mirroring the evaluation process typically carried out by HR professionals. By analyzing both verbal and non-verbal cues presented by each candidate, the system generates a comprehensive analysis that includes detailed scores

across multiple key attributes—such as English language proficiency, personality traits, facial expressions, and the semantic relevance between interview questions and responses—along with an overall score that reflects the candidate's suitability for the role. This automated evaluation model addresses the time-consuming and exhausting nature of large-scale recruitment processes, where organizations often face challenges in reviewing high volumes of candidate submissions through traditional methods. Manually assessing each interview can lead to significant delays, inconsistencies, and an increased risk of bias or human error. By implementing advanced artificial intelligence techniques, our system streamlines the hiring workflow, ensuring faster, fairer, and more consistent assessments. This allows recruitment teams to focus on higher-level decision-making rather than repetitive evaluation tasks, ultimately enhancing the quality of hires while reducing operational costs. As organizations continue to scale and compete for top talent, the need for a reliable, efficient, and scalable hiring solution becomes increasingly vital, positioning this project as a critical advancement in the evolution of modern recruitment practices.

1.2 Problem Definition

Modern hiring processes, while increasingly digital, still face a range of critical issues that impact their effectiveness and fairness. One of the key challenges is the variation in technical quality during pre-recorded interviews—candidates using low-quality cameras, microphones, or unstable internet connections may be unfairly judged due to factors unrelated to their actual performance or qualifications.

Another pressing concern is the lengthy duration of recruitment cycles. In countries like the United States, the average hiring process can take up to 27 days, creating delays that affect organizational efficiency and the ability to

secure top talent quickly. This inefficiency becomes even more pronounced when dealing with large volumes of candidates.

In addition, bias in candidate evaluation continues to be a major obstacle to objective hiring. Reports show that 42% of recruiters acknowledge that biased judgments can lead to the selection of underperforming or mismatched candidates. This bias can take many forms, including stereotyping, favoritism, first impression bias, recency bias, and discrimination against disabilities, all of which undermine efforts to create a fair and inclusive recruitment process.

1.3 Objective

The primary goal of the **PRVIA "Pre-Recorded Video Interview Analysis"** project is to streamline and enhance the recruitment process by employing artificial intelligence (AI) technologies to automate and optimize pre-recorded video interview assessments. The following objectives outline the project's aims and the steps taken to achieve them:

1. Simplify Pre-Recorded Video Interview Processes

- Implement AI-driven technologies to automate the analysis of pre-recorded video interviews, reducing manual effort required by HR teams.
- Develop a user-friendly system that integrates seamlessly into existing recruitment workflows.

2. Evaluate Candidates Based on Verbal and Non-Verbal Cues

Apply advanced AI algorithms to assess candidates comprehensively, focusing on:

- **Personality Traits**: Identify attributes relevant to job performance, such as communication skills and adaptability.

- **English Proficiency**: Evaluate language fluency and clarity to ensure effective communication.

- **Facial Expressions**: Analyze non-verbal cues such as confidence and engagement.

- **Relevance Between Questions and Answers**: Determine how well responses align with the context and requirements of the questions.

3. Reduce Time and Effort in Large-Scale Hiring

- Accelerate the recruitment process by automating evaluations, significantly reducing time-to-hire, especially in high-volume scenarios.

- Enable HR teams to set specific criteria or filtering ratios for efficient candidate shortlisting.

4. Improve HR Decision-Making

- Minimize bias in evaluations by standardizing assessments through objective AI-driven analysis, addressing issues like stereotyping, recency bias, and first impressions.

- Provide concise, AI-generated reports that summarize candidate insights, supporting informed and data-driven decisions.

5. Additional Benefits

- **Facilitate Remote Hiring**: Support geographically distributed recruitment with seamless video interview evaluations.
- **Reduce Costs**: Lower expenses by minimizing manual reviews and reducing reliance on in-person interviews.
- **Enhance Scalability**: Deliver a consistent and efficient hiring solution for organizations expanding their workforce.

1.4 Document Organization

- **Chapter 2 – Background:** This chapter provides the foundational context of the project. It introduces the field of automated interview analysis, detailing its relevance within modern recruitment processes. The chapter covers the scientific background related to audio, video, and text-based assessment, and presents a comprehensive survey of prior research and existing systems. It highlights the limitations of current solutions and defines the gap that PRVIA aims to address.
- **Chapter 3 – Analysis and Design:** This chapter presents a detailed analysis of the system's goals, stakeholders, and requirements. It outlines both functional and non-functional requirements and provides a high-level system overview. The design section includes architectural diagrams, use case models, and behavioral descriptions, explaining how the system components interact to fulfill project objectives. Both candidate and HR user perspectives are considered throughout the analysis.
- **Chapter 4 – Implementation:** This chapter describes how the system and its underlying AI models were developed and trained. It outlines the full implementation process, including data preparation, model selection, and integration of components such as video analysis, speech

recognition, and text evaluation. The chapter presents the techniques used to extract and process multi-modal features and details how various machine learning models were fine-tuned for the specific task of candidate evaluation. It also highlights the system's overall performance and presents the results obtained through testing and evaluation. This chapter captures the complete development effort and demonstrates how the system meets its intended goals.

- **Chapter 5 – User Manual:** This chapter serves as a guide for users interacting with the system. It explains how both candidates and HR administrators can access and operate the platform. The manual walks through key functionalities such as job browsing, form submission, video interview uploading, job posting, and candidate review. Screenshots are provided to illustrate the user interface and ensure ease of use for all user roles.
- **Chapter 6 – Conclusions and Future Work:** The final chapter summarizes the accomplishments of the project, emphasizing the significance of automated video interview evaluation in modern hiring. It discusses the benefits achieved through the system and outlines areas for future enhancements, such as expanding scalability, improving security, integrating more advanced AI models, and deploying the platform to cloud environments for real-world use.

Chapter 2

Background

2.1 Field Description:

Automated video interview analysis represents a dynamic convergence of artificial intelligence (AI) and human resource (HR) technology, leveraging advanced computational techniques to evaluate candidate performance in pre-recorded job interviews. This field integrates three core AI disciplines—Speech Processing,Computer Vision, and Natural Language Processing (NLP)—to provide objective, data-driven insights that enhance the efficiency and fairness of recruitment processes. The growing demand for scalable and unbiased hiring solutions, particularly in global and remote work environments, has driven the development of such systems, addressing the limitations of traditional CV-based screening and subjective interview assessments.

Speech Processing, a critical AI subfield, focuses on analyzing and interpreting spoken language. In the context of video interview analysis, it encompasses **Automatic Speech Recognition (ASR)** for transcribing audio into text and **Automatic Pronunciation Assessment (APA)** for evaluating fluency, pronunciation,and language effectiveness. **Computer Vision** enables the analysis of visual data from interview videos.Techniques such as face detection, facial expression recognition, and eye-tracking are employed to infer personality traits, emotional states, and attentiveness.

Deep learning models, particularly convolutional neural networks (CNNs) and transformer-based architectures, are trained on large-scale visual datasets to

extract features like gaze direction and micro-expressions, providing insights into candidate behavior beyond verbal responses.

Natural Language Processing (NLP) focuses on understanding and generating human language. In this project, NLP models, including large language models (LLMs) like Gemini, analyze transcribed interview responses to assess grammatical accuracy, semantic alignment with interview questions, and response coherence. The integration of these AI disciplines enables a holistic evaluation of candidates, aligning with HR objectives of objectivity, scalability, and efficiency. However, challenges such as mitigating algorithmic bias and ensuring ethical AI deployment remain critical considerations in this interdisciplinary field.

2.2 Scientific Background:

2.2.1 Audio Module

The audio module of this system is based on research areas like **automatic speech recognition (ASR)**, **speech processing**, and **deep learning for audio classification**. These areas have developed greatly over time and are now key to building accurate and efficient pronunciation assessment tools.

Early ASR and Forced Alignment Methods: In the early stages of pronunciation assessment, systems mainly used statistical models such as Hidden Markov Models (HMMs) and Gaussian Mixture Models (GMMs). These systems matched the learner's speech to a written reference using a process called forced alignment. This allowed the system to break the speech into phonemes (the basic sound units of language) and measure how closely they matched correct pronunciations. The results were summarized into scores like the Goodness of Pronunciation (GOP), which were then passed to machine learning models to estimate pronunciation quality.

Although this method worked for basic evaluation, it had some important drawbacks. It required a lot of manual setup and didn't work well with different accents or spontaneous speech. Also, because it relied on aligning every sound in the audio to a written script, it was slow and not suitable for real-time use.

Deep Learning and Self-Supervised Representations: With the rise of deep learning, pronunciation systems became much more powerful and flexible. Instead of using manually designed features, deep learning models can learn useful patterns from audio data on their own. Early deep models used layers like convolutional or recurrent neural networks to process the audio signal and extract meaningful features.

A major breakthrough came with self-supervised learning (SSL) models such as wav2vec, HuBERT, and data2vec. These models are trained on large audio datasets without needing labeled data. They learn to understand speech by predicting missing parts of the audio, which helps them develop a strong sense of how speech sounds in different contexts. These learned features can then be used to assess pronunciation or fluency more accurately, without needing to align each sound to a script.

Transformers and Multi-Aspect Learning: Another important improvement is the use of Transformer models, which are especially good at understanding sequences, like speech or text. These models can process the entire spoken sentence at once and learn how different parts relate to each other. In pronunciation assessment, Transformers are used to evaluate many aspects of speech at the same time, such as accuracy, fluency, completeness, and rhythm.

Many modern systems also use multi-task learning, where the model learns to perform several related tasks together. This helps the model become more

accurate and general, as it learns to consider speech from different perspectives.

As pronunciation assessment moves beyond language learning and into areas like job interviews, the focus changes. Instead of checking for small mistakes in pronunciation, the system must judge overall communication ability. This includes how fluent and clear the speaker is, how well they organize their answers, and how understandable they are to others.

2.2.2 Video Module

The video module in this system is grounded in foundational concepts from machine learning, computer vision, and psychometrics. Early facial analysis techniques relied on handcrafted features and facial landmarks to detect expressions and behavioral cues, but these approaches lacked robustness to variations in lighting, pose, and facial structure. With the advent of deep learning, Convolutional Neural Networks (CNNs) became the standard for extracting spatial features from facial images, enabling more accurate detection of subtle patterns like eyebrow movement or mouth shape linked to emotion and personality. To model temporal dynamics—such as gaze shifts, blink rate, and smile progression—3D CNNs like X3D were introduced, allowing the system to analyze video sequences over time rather than isolated frames. Accurate face detection and alignment, performed using MTCNN, ensure consistency across frames by compensating for camera angles and head movement. The module also leverages multi-task learning to simultaneously analyze multiple visual cues, such as emotion, engagement, and personality, improving generalization. Emotion classification is informed by valence theory, which groups emotions into positive and negative categories, simplifying interpretation and aligning with human perception. Personality trait prediction is treated as a regression problem, producing

continuous scores in line with the Big Five model: Extraversion, Agreeableness, Conscientiousness, Neuroticism, and Openness. As applications move beyond academic settings to real-world scenarios like job interviews, the focus expands from isolated facial expressions to broader communication indicators such as confidence, attentiveness, and expressiveness across time.

2.2.3 Text Module

The text module of this system is built upon advancements in natural language processing (NLP), and large language models (LLMs). Over the past decade, these fields have rapidly evolved, driven by deep learning architectures that significantly improved machine understanding of human language. The primary functions of this module involve automatic speech transcription, abstractive summarization and relevance check between the questions and the candidate's answers.

Speech-to-Text using Whisper Models: Transcription is the first step in the text pipeline, where the audio input is converted into written text using automatic speech recognition (ASR). In this project, the Whisper-medium model developed by OpenAI is used for transcription. Unlike traditional ASR systems that rely on manually engineered acoustic and language models, Whisper is trained end-to-end on a large-scale, diverse, multilingual dataset. It is robust to various accents, background noise, and spontaneous speech, which makes it particularly effective in real-world settings like interviews.

The Whisper model leverages the Transformer architecture, which processes the entire audio sequence holistically and can handle long dependencies across the signal. By combining supervised training with large amounts of

unlabeled audio, Whisper achieves state-of-the-art transcription performance across many languages and domains. This enables the system to provide highly accurate and fluent textual representations of spoken input, which are crucial for downstream text analysis.

Text Summarization: Once transcription is complete, the system applies text summarization to condense long, spoken responses into brief, informative summaries. Text summarization has evolved from traditional extractive methods, which select key sentences, to modern abstract methods that can paraphrase, rephrase, and generate novel sentences while preserving the original meaning.

Abstractive summarization models like BART-Large, FLAN-T5-Large, and Google's Gemini are used in this project. These models are based on Transformer encoder-decoder architectures and are pre-trained on large text corpora using denoising or instruction-tuned objectives. BART (Bidirectional and Auto-Regressive Transformers) is particularly powerful in handling noisy and unstructured input, which makes it ideal for summarizing transcribed speech that may contain disfluencies or informal expressions.

Relevance Checking: The relevance checking task is grounded in decades of research in NLP and computational linguistics. Foundational work on semantic similarity began with vector space models [19], where documents are represented as vectors, and similarity is measured using metrics like cosine similarity, as expressed by .

$$\cos(\theta) = \frac{A \cdot B}{|A||B|}$$

The development of word embeddings [20] and contextual embeddings [21] enabled deeper semantic understanding, which is critical for distinguishing relevant from irrelevant answers in interview contexts. TF-IDF [22] remains a robust method for keyword extraction, as shown in ,

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \cdot \log\left(\frac{N}{\text{DF}(t)}\right)$$

allowing the system to emphasize domain-specific terms like “collaboration” or “problem-solving.” Recent advancements in large language models, leveraging transformer architectures [23] and prompt engineering, achieve human-like contextual reasoning, as demonstrated in the final relevance checking approach. Challenges include subjectivity in defining relevance, transcription errors from automatic speech recognition, and difficulties in setting universal thresholds due to overlapping score distributions. These are addressed through validation against human judgments and hybrid methods combining semantic and keyword scores, as in ,

$$\text{Score} = w_1 \cdot \text{SemanticScore} + w_2 \cdot \text{KeywordScore}$$

which balances contextual understanding with keyword relevance.

Personality Traits Analysis: Personality traits analysis builds on psycholinguistic research linking language use to personality characteristics. The Linguistic Inquiry and Word Count (LIWC) framework [24] demonstrated that linguistic markers, such as word frequency and sentiment, correlate with personality traits. The Big Five personality model [25] provides a standardized taxonomy for assessing traits like Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism, widely used in psychological research. Early NLP studies [26] showed that machine learning models can predict personality traits from text using features like word choice, syntactic patterns, and emotional tone. For example, frequent use of positive emotion words may indicate Extraversion, while structured language suggests Conscientiousness. Modern transformer-based models, such as BERT [27], enhance prediction accuracy by capturing contextual semantics, with fine-tuning on datasets like myPersonality or the Essays dataset enabling robust trait inference. Challenges include linguistic variability across cultures, noise in transcribed speech, and ethical concerns about bias in predictive models. These are mitigated through diverse training data, transparent

validation against psychometric benchmarks, and careful feature selection to ensure fairness and reliability. This interdisciplinary approach combines computational techniques with psychological insights to support automated personality assessment in hiring systems.

2.3 Survey of Work Done

2.3.1 Audio Module

Over the past two decades, numerous systems have been developed to automatically assess pronunciation, each reflecting the technological advancements of its time. These systems have evolved from early alignment-based models with handcrafted features to recent end-to-end, multi-modal pipelines powered by deep learning and large language models. The following is a survey of key research efforts and systems that shaped the field of automatic pronunciation assessment (APA):

[18]: Goodness of Pronunciation (GOP): One of the earliest influential works in APA, Witt & Young (2000), introduced the Goodness of Pronunciation (GOP) framework. This system relied on a deep neural network–hidden Markov model (DNN-HMM) based ASR to perform forced alignment between learner speech and a reference transcript. The system computed frame-level posterior probabilities for each phoneme and derived the GOP score by comparing expected and observed phonetic likelihoods. These scores were used as features for Support Vector Machine (SVM) regressors, which provided phoneme-level pronunciation ratings. This approach laid the foundation for many subsequent systems, though it was limited by its dependence on handcrafted features and rigid alignment processes.

[15]: GOPT – A Multi-Aspect Transformer-Based Model: With the rise of deep learning,this paper introduced **GOPT** (Goodness of Pronunciation

Transformer), a model designed to improve both accuracy and feedback richness in APA systems. GOPT uses traditional GOP features but feeds them into a Transformer-based self-attention network, allowing the model to jointly assess multiple aspects of pronunciation. These include accuracy, fluency, completeness, prosody, and stress, assessed across different linguistic granularities (phoneme, word, utterance). GOPT adopts a multi-task learning framework, where each output head learns a specific evaluation metric. This design allows the system to provide detailed and multi-level feedback to learners, making it a strong candidate for educational applications. However, due to its reliance on alignment-based features, GOPT inherits some of the computational limitations of earlier systems.

[16]: Alignment-Free Scoring with Self-Supervised Learning: To address the complexity of forced alignment, Fu et al. proposed a method that eliminates the alignment step altogether. Their system uses self-supervised learning (SSL) models such as wav2vec2 to extract rich, contextual audio embeddings directly from raw speech. These features are then fed into a scoring model that predicts fluency and intelligibility without any need for forced alignment or phoneme segmentation. This approach greatly reduces preprocessing time, improves robustness to diverse speech inputs, and enables real-time assessment, making it well-suited for practical applications like automated interview screening.

[17]: Pronunciation Assessment with Multi-Modal LLMs: The most recent development in the field is the work by Fu et al. (2024), which proposes a **multi-modal, alignment-free pipeline** using large language models. The system combines **data2vec2**, a contextual speech encoder, with a **modality adapter** that transforms audio embeddings into a format compatible with text-based models. These features are then passed to a **decoder-only large language model (Qwen-7B)** that directly predicts utterance-level scores for

fluency and **accuracy**. This architecture removes the need for intermediate alignment or phoneme-based scoring entirely, and instead treats pronunciation assessment as a **text-conditioned sequence modeling problem**. By leveraging the generalization capabilities of LLMs, the system can understand both the speech signal and its semantic content, offering more reliable and scalable assessments across varied accents and speaking styles.

2.3.2 Video Module

Over the past decade, video-based personality and emotion analysis has evolved from facial-expression-only systems to multimodal, context-aware, and transformer-based architectures. This section surveys key developments in personality trait prediction and emotion recognition from facial and scene-level cues in job interview contexts.

[6] CR-Net: Classification-Regression Network for Personality Prediction: Li et al. (2020) introduced **CR-Net,** a deep learning framework designed for predicting Big Five personality traits from multimodal video data. The model processes both full-scene and facial video frames using **CNNs** and applies a classification-regression pipeline to improve accuracy. A novel Bell Loss function was proposed to address regression-to-the-mean issues. This architecture significantly outperformed prior models on the ChaLearn First Impressions dataset, establishing CR-Net as a robust solution for interview-based personality assessment.

[12] Multimodal Interview Judgment Using Video Corpus: Chen et al. (2017) presented one of the largest annotated corpora for video interviews (1,891 videos, 63 hours), focusing on hiring recommendation and trait prediction. Their system applied clustering on multimodal features (facial expressions, prosody, speech content) and transformed them into pseudo-documents processed by text

classifiers. The study concluded that textual content was most predictive, but facial and prosodic cues contributed marginally and needed further refinement.

[11] CAER-Net: Context-Aware Emotion Recognition: Lee et al. (2019) addressed a limitation in conventional emotion recognition — the lack of context. CAER-Net introduced a two-stream model where one stream processes facial expressions, and the other captures contextual scene cues (e.g., background, objects). These streams are fused adaptively using attention mechanisms, significantly boosting performance on the CAER benchmark.

[10] Vision-Language Models for In-Context Emotion Understanding: Xenos et al. (2024) leveraged the power of Vision-and-Language Models (VLLMs) in a two-stage pipeline: the model first generates natural language descriptions of apparent emotions from video scenes and then fuses these with visual features using a transformer. This approach outperformed single-modality baselines on datasets like CAER-S and BoLD while requiring less training complexity.

[14] AI-Human Hybrid Model for Sales Hiring: Chakraborty et al. (2023) proposed an AI-human hybrid pipeline for analyzing back-and-forth conversational interviews. Their model fused text, voice, and body language features to estimate latent sales ability. Results showed that integrating minimal human input alongside AI substantially improved hiring performance — a practical benchmark for AI-assisted recruitment systems.

2.4 Existing Similar Systems

Several systems, both research-based and commercial, provide automated video interview analysis, offering insights into their features and applications.

2.4.1 Commercial Platforms:

- **HireVue :**HireVue is a widely adopted AI-driven hiring platform that analyzes candidates' verbal and nonverbal cues during structured video interviews. It provides real-time scoring on traits like communication, professionalism, and job fit, using machine learning models. Integrated with major Applicant Tracking Systems (ATS).HireVue focuses primarily on audio-based assessments and speech-based trait evaluation, serving as a benchmark for enterprise-grade scalability.
- **TalView:** Talview is a remote hiring platform that combines video interviews, LLM-based scoring, and AI-driven behavioral analysis. It supports fluency evaluation, cheating/gaze detection, and personality profiling through facial analytics. While powerful for enterprise hiring, Talview lacks detailed pronunciation scoring, semantic answer-question evaluation, and academic benchmarking, which PRVIA uniquely provides.

Table 2.1 Existing Similar Systems Comparison

Platform	Emotion Detection	Personality Trait Evaluation	English Scoring	Summarization	Answer–Question Relevance	Gaze/Cheating Detection
HireVue	❌	✔️	❌	❌	❌	❌
TalView	✔️	✔️	❌	✔️	❌	✔️
PRVIA	✔️	✔️	✔️	✔️	✔️	✔️

Chapter 3

Analysis and Design

This chapter provides an in-depth analysis and design of the Pre-Recorded Video Interview Analysis system, designed to facilitate automated candidate evaluation for job applications. The system integrates advanced deep learning modules to assess video interviews, supporting both candidates and human resource (HR) professionals. The analysis outlines the system overview, including its architecture, functional and nonfunctional requirements, and intended users. The design section presents the system's structural and behavioral models to meet project objectives.

3.1 System Overview

3.1.1 System Architecture

The PRVIA system is architected as a three-layered application, comprising the Presentation Layer, Application Layer, and Data Layer, as shown in Figure 3.1

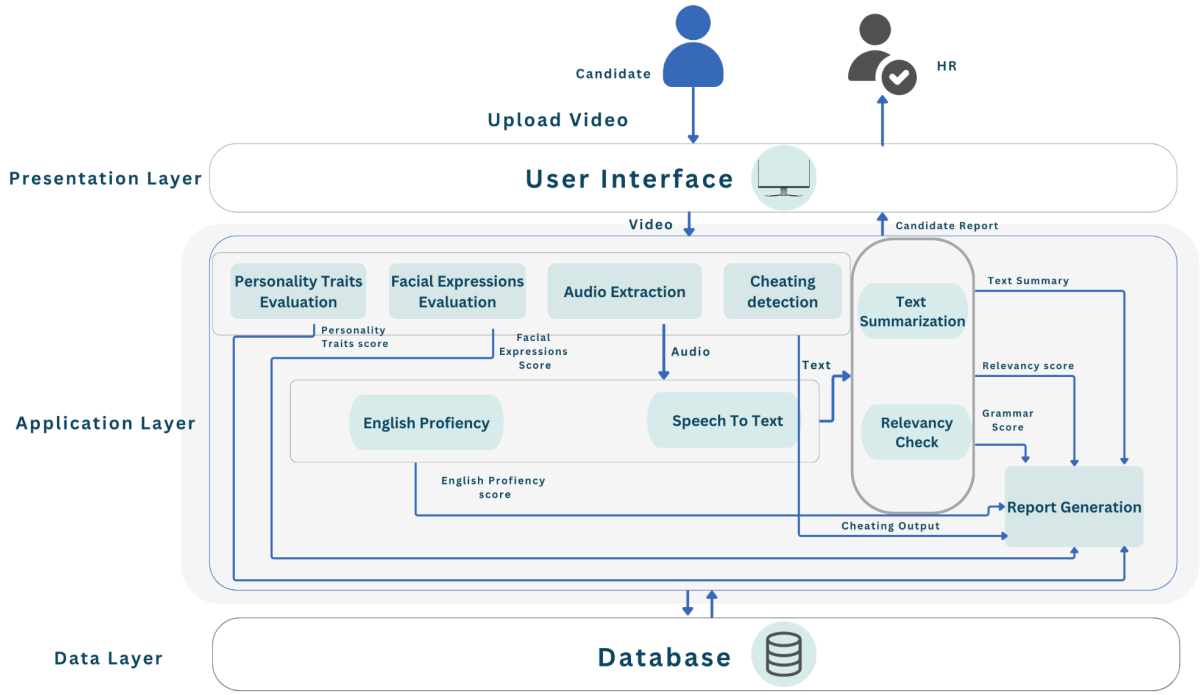


Fig 3.1 System Architecture

The system architecture consists of 4 primary modules—Video Module, Audio Module, Text Module, and Report Generation aside user interface components and a database. Each module consists of a set of sub-modules to handle aspects of the interview analysis pipeline that will be discussed in detail :

Modules Description:

- **Video Module:** This is the first module that takes the video input from the user in the presentation layer and starts the processing it consists of 4 sub modules: (Personality Traits Evaluation, Facial Expressions Evaluation, Cheating Detection, Audio extraction)
    - **Personality Traits Evaluation:** This module is responsible for analyzing and evaluating the traits of the candidate's personality traits from (Openness, Conscientiousness, Extraversion, Agreeableness, Neuroticism) , and returns the personality evaluation of the candidate through the video interview based on the scores of the model.
    - **Facial Expressions Evaluation:** This module is responsible for understanding the candidate's face during the video and how open the candidate to the interview through his facial movements and expressions.
    - **Cheating Detection:** This module is responsible tracking the candidate eye during the interview, in order to detect if he is reading from a paper or searching for something when answering technical questions and how long is his eye distracted away from the camera during the video
    - **Audio Extraction:** This module is responsible for extracting the audios from the uploaded videos to further analyze the user audio and complete the analysis pipeline.

- **Audio Module:** This is the second module, it takes the extracted audio from the video module and uses the audio for analyzing the user english level throughout the interview, it mainly consists of 2 sub-modules: (English Level Evaluation, Speech To Text Conversion)
    - **English Level Evaluation:** This is the module responsible for the evaluation of the english level of the user, as the users are asked to submit the videos in english, where some jobs require high english level and average ones,It mainly takes the user audio spoken in the video and returns an overall english level score that describes either his english is low,average or high.
    - **Speech To Text Conversion:** This module is responsible for converting the user audio to text, to further analyze the text spoken by the user during the interview in the third module.

- **Text Module:** This is the third module, this module takes the text from the audio module and start analyzing it through 3 sub modules which are (Text Summarization, Relevancy Checking, Grammar Error Correctness)
    - **Text Summarization:** This module is responsible for making a rich summary of the text spoken by the user during the interview, highlighting key points and information .
    - **Relevancy Checking:** This module is responsible for measuring the relevancy between the question the user is responding to and his answer throughout the interview, this check to avoid any distractions of answers beyond the asked question.

- **Report Generation:** Aggregates outputs from all modules to produce a comprehensive candidate report, including personality traits, English proficiency, cheating detection, and text analysis results, accessible to HR users.

The Presentation Layer hosts the User Interface, enabling candidates to upload videos and HR professionals to access reports. The Application Layer executes the analysis pipeline, while the Data Layer stores processed data in the system database, ensuring persistent storage and retrieval.

3.1.2 Functional Requirements

The system supports the following core functional functions:

- **Video Upload:** Candidates can upload three pre-recorded video responses per job application via the website interface.
- **Jobs Exploration:** Candidates seeking jobs are able to navigate through the home page and explore the available positions and apply for any.
- **Job Application:** Each Candidate can apply to any of the available positions, by submitting their personal information and CV, for HR to evaluate their suitability and send them a link for interview for the second phase.
- **Admins Registration/Login**: Each HR admin associated with company, will have to register to the system and have credentials to be able to view candidates applied for the jobs they have created.

- **Create /Add jobs:** Each HR admin will be able to create the jobs in the company he is associated with and will only be able to view the positions he has created only.
- **Link Generation:** HR admins are able to see all the applicants have submitted applications and their CVs to any job, and if they find them suitable, they can generate a link specially for them and send them it through email to proceed to the next phase.
- **Report Generation:** Produce a detailed report for HR about each candidate completed the video interviews, including all evaluation scores.
- **Video Processing:** This includes all the functions of the analysis from (**Personality Traits Evaluation - Facial Expressions Evaluation-Cheating Detection-English Proficiency Evaluation -Speech to Text Conversion-Audio Extraction-Text Summarization- Relevancy Checking)** for each video interview uploaded by the users, and provide the report for HR accordingly.

3.1.3 Nonfunctional Requirements

Besides Functional requirements, the system adheres to several non-functional requirements that define its quality, performance, and user experience. These aspects ensure that the system operates reliably, securely, and efficiently under expected conditions

- **Performance:** The system processes each stage of candidate evaluation within a consistent and efficient timeframe. On average, the full evaluation pipeline—including video handling, audio transcription, text processing, language scoring, and result generation—completes in

under 2.5 minutes per candidate. This makes the platform well-suited for asynchronous HR workflows, The system's average processing speed ensures that recruiters receive timely, actionable insights while maintaining the depth and quality of assessment.

- **Usability:** PRVIA is designed for both non-technical job applicants and HR administrators with basic computer proficiency. The interface is intuitive and user-friendly, using clean layouts, gradient-colored buttons, and minimal instructions to guide users through the process of job application and interview submission.

- **Security:** Security measures have been implemented as Authentication and authorization mechanisms are in place to restrict access for HR administrators.User data and video files are stored securely in the local environment.

- **Maintainability:** The backend is developed using object-oriented programming (OOP) principles, with modular components for models, routes, and utilities, making it moderately maintainable. While feature extensions are possible, improvements in documentation and deployment structure would enhance long-term maintainability.

- **Portability:** It is platform-independent system and can be run on different operating systems such as Windows, macOS, and Linux, provided the required dependencies (Python, Node.js, PostgreSQL) are installed.

- **Scalability:** At its current stage, PRVIA supports small-scale usage by a limited number of users and job postings. While the architecture could potentially be adapted for horizontal scaling (e.g., increased volume of video uploads), this has not yet been implemented or tested. Future improvements may include load balancing, parallel processing, or cloud integration to enhance scalability.

3.1.4 System Users

The system is designed for two primary user groups:

A. Intended Users:

- **Candidates:** Individuals seeking employment who use the system to apply for jobs, submit CVs, and upload pre-recorded video interviews in response to specific job-related questions. They interact with the website to navigate job listings, track application status, and receive email notifications.
- **HR Professionals:** Company representatives who register, create job postings, manage applicants, shortlist candidates for video interviews, review detailed evaluation reports, and make hiring decisions. They use the admin interface to oversee their company's recruitment process and communicate with candidates via email.

B. User Characteristics

- **Candidates:** Only basic computer literacy is required. Candidates should be familiar with how to navigate a website, fill out online forms, and upload video files in a simple and guided manner. No prior knowledge of video recording tools, artificial intelligence, or technical systems is necessary.
- **HR Professionals:** should be trained recruiters or professionals with foundational human resources skills. They should be capable of reviewing CVs, evaluating candidate applications, and making informed decisions based on the reports generated by the system. They should

also have proficiency in web-based administration tools, including user registration, data export, and email management,which includes managing applicant data, exporting links, and sending emails.No technical expertise in AI or data interpretation is needed, as the system provides clear and user-friendly reports.

3.2 System Analysis & Design

3.2.1 Use Case Diagram

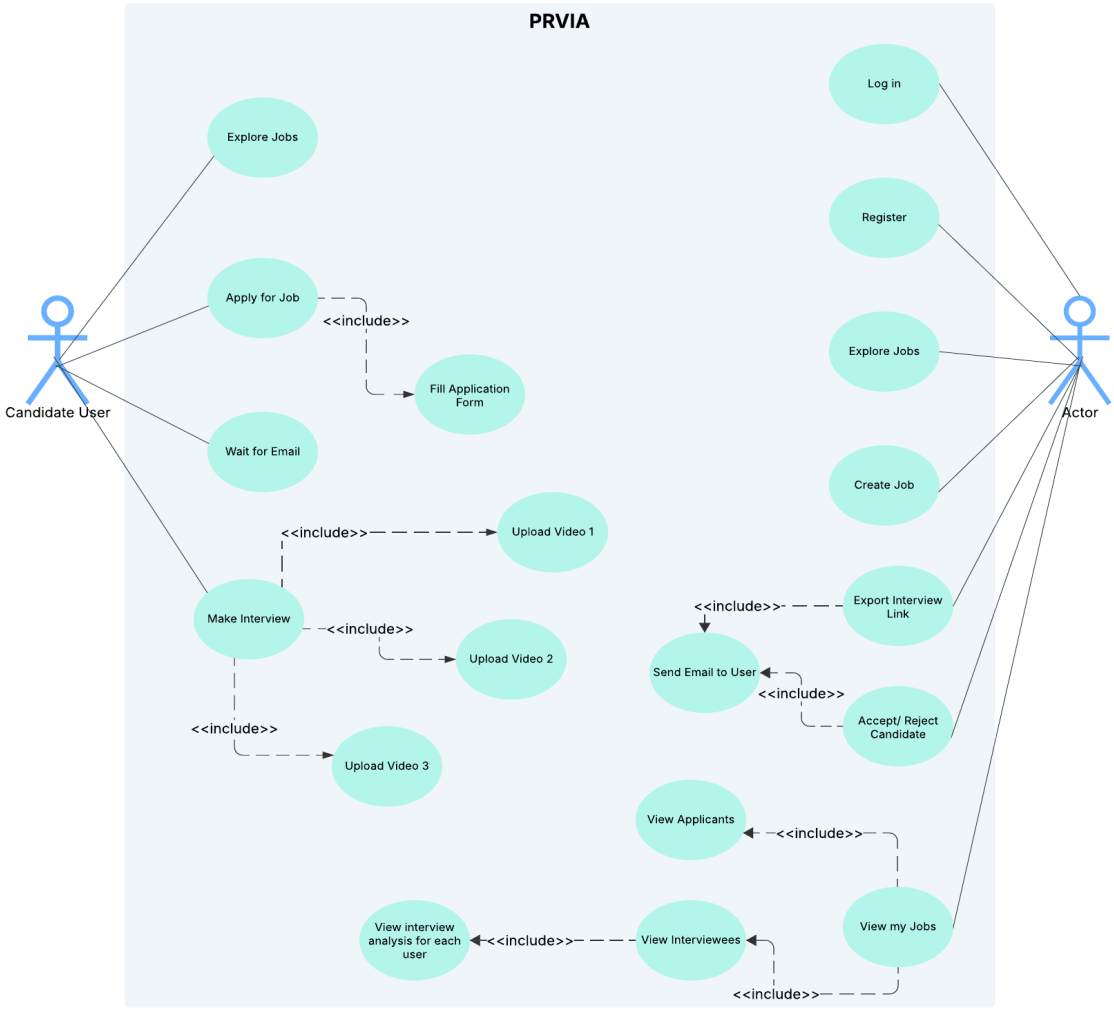


Fig 3.2 Use Case Diagram

3.2.2 Class Diagram

To support the flow of data and processing across users, job postings, video submissions, and AI-generated reports, our platform is structured using a well-defined set of classes. The following **Class Diagram** represents the key components of our backend design and how they interact with each other.

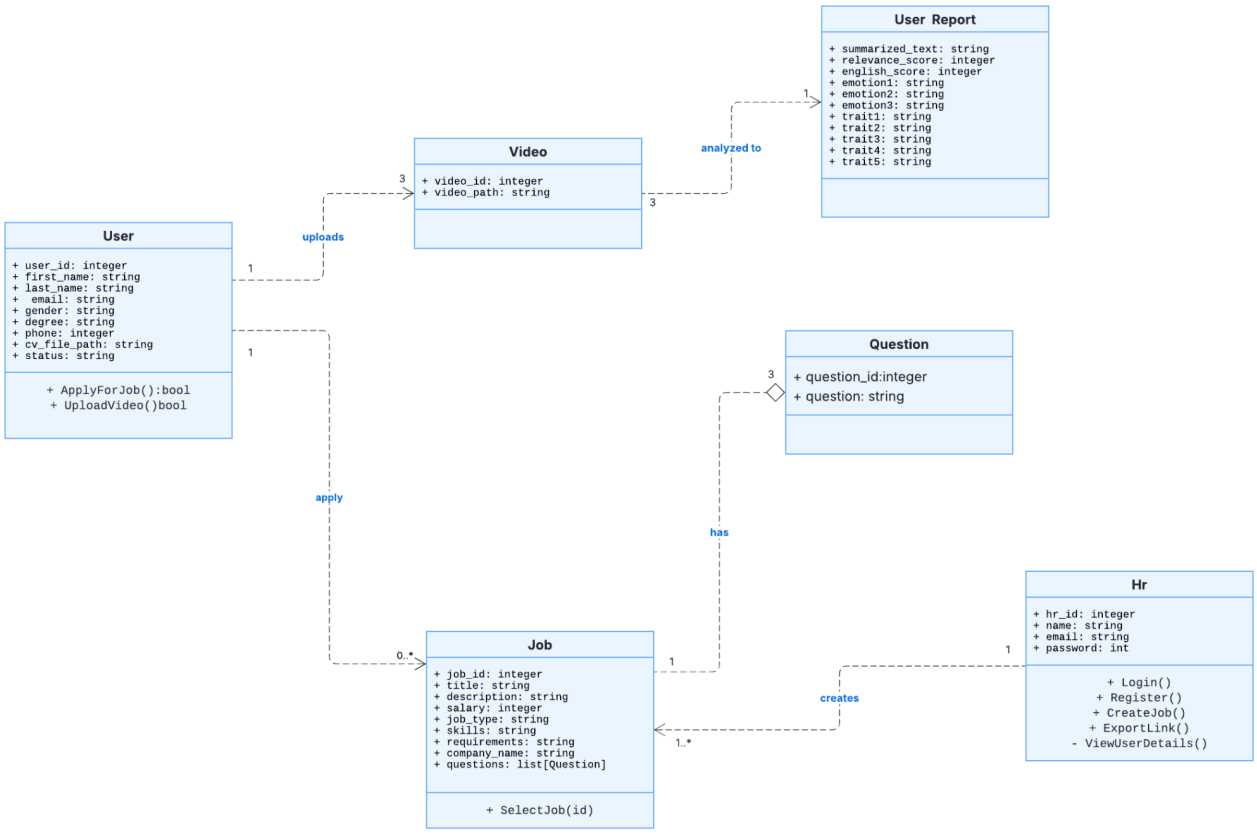


Fig 3.3 Class Diagram

In the class diagram, we have several classes such as **User**, **HR**, **Job**, **Question**, **Video**, and **UserReport**.

The **User** class represents job applicants and contains attributes like user_id, first_name, last_name, email, gender, degree, phone, cv_file_path, and status. This class includes user-related actions such as applying to jobs and submitting videos. Every user can apply to multiple jobs, and each user can have multiple submitted videos and reports, but each video/report is linked to only one user.

The **HR** class represents the human resources personnel managing job posts. It contains fields like hr_id, name, email, and password. Each HR can create and manage multiple jobs, forming a one-to-many (1:m) relationship between HR and Job.

The **Job** class includes job-related data such as job_id, title, description, salary, job_type, skills, requirements, company_name, questions, and hr_id. Each job post is created by an HR member and can contain multiple interview questions. Therefore, the relation between Job and Question is 1:m, meaning each job has multiple questions but each question belongs to only one job.

The **Question** class stores each interview question separately. It includes attributes such as question_id, job_id, and question. Every question is connected to a job and is answered by users in the form of videos.

The **Video** class represents the videos submitted by users. It includes video_id, user_id, question_id, and job_id. A user submits a video for a specific question of a specific job. This creates a many-to-one relation from videos to user, job, and question.

The **UserReport** class captures the full results of AI-driven analysis conducted on each submitted video response in the virtual interview system. It contains key attributes such as user_id, question_id, video_id, summarized_text, relevance_score, and english_score. This class has a 1:1 relation with the **Video** class as every submitted video generates exactly one report.

The **UserReport** class also includes **emotional assessments** and **personality trait predictions** that provide a deeper insight into the candidate's behavior and communication during the interview.

- The fields emotion1, emotion2, and emotion3 store textual summaries based on facial emotion recognition from each of the user's three submitted videos. A deep learning model (DeepFace) analyzes expressions across time and classifies emotions into positive (happy, surprise, neutral) and negative (sad, angry, fear, disgust) categories. Depending on the dominant emotion group, the system outputs high-level textual assessments such as:

    - "The candidate appeared confident and engaged."
    - "The candidate might have felt nervous, frustrated, or disengaged."
    - "The candidate had mixed reactions, indicating varying confidence levels."

- The traits trait1 through trait5 represent predictions for the **Big Five** personality dimensions: Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism. These are computed using a spatiotemporal deep learning model that captures non-verbal cues like facial expressions and motion patterns. Each trait is scored on a normalized scale (0–1) and provides insight into the candidate's behavioral tendencies. These fields replace traditional personality surveys by offering objective, video-based evaluations:

    - trait1: "Agreeableness" – Authentic/Self-Interested
    - trait2: "Conscientiousness" – Organized/Sloppy
    - trait3: "Extraversion" – Friendly/Reserved
    - trait4: "Neuroticism" – Comfortable/Uneasy
    - trait5: "Openness" – Imaginative/Practical

This class is essential for generating automated evaluations that combine emotion, language proficiency, and behavioral traits, supporting data-driven and unbiased hiring decisions.

The system is designed to support HR managers in publishing jobs and questions, and users in applying and submitting video responses. The videos are then analyzed for relevance, language skills, emotions, and traits, and the results are stored for review.

3.2.3 Sequence Diagrams

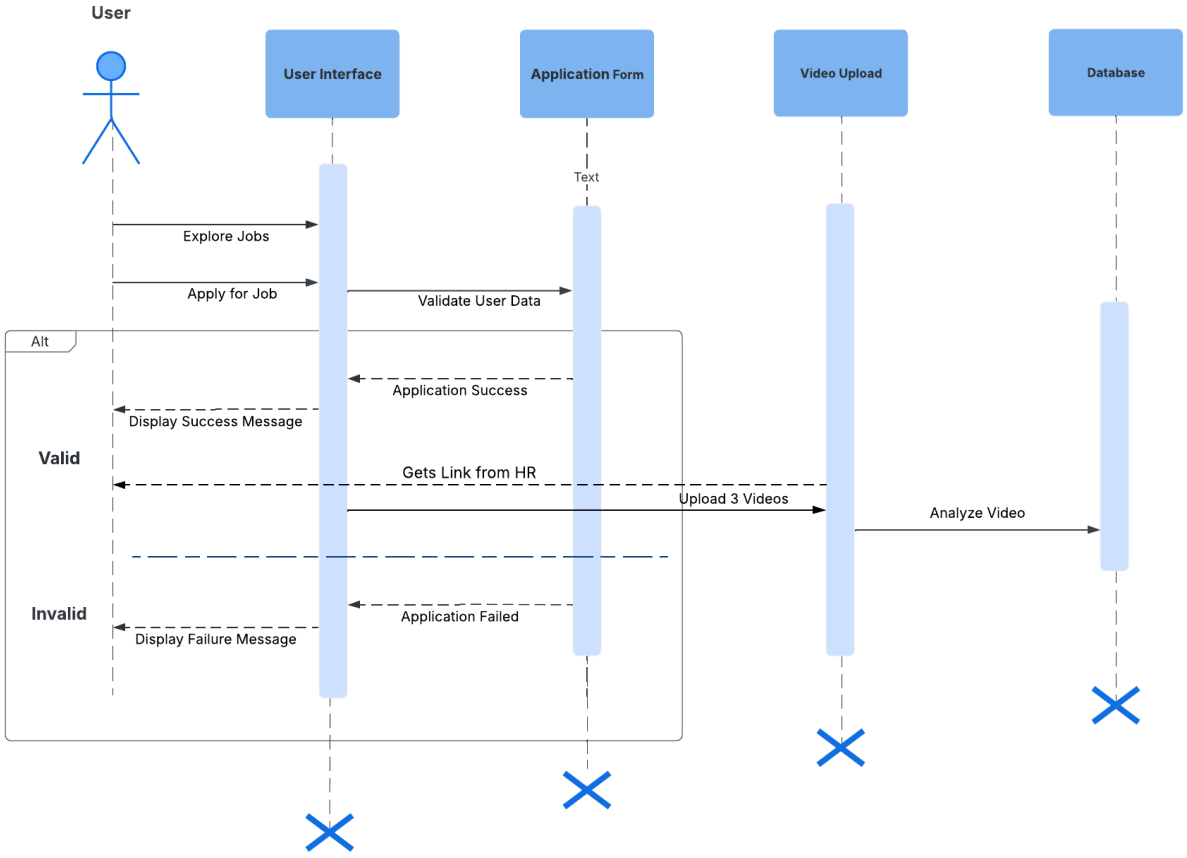


Fig 3.4 User Sequence Diagram

The presented **User Sequence Diagram** illustrates the step-by-step interaction between the system components and the user within the PRVIA recruitment application process. This sequence captures the flow from the initial job exploration to the final video analysis stage, highlighting both successful and failed scenarios.

Sequence Flow:

1. **Job Exploration and Application:**
    - The process begins when the **User** accesses the system via the **User Interface** to explore available job opportunities.
    - Upon selecting a desired position, the user proceeds to **Apply for Job** through the same interface.

2. **Application Validation:**
    - The system transitions to the **Application Form** component, where the submitted user data is validated.
    - Based on the validation outcome:
        - If the application is **valid**, the system confirms the **Application Success** and provides the user with a link from HR to proceed with the next steps.
        - If the application is **invalid**, the system immediately terminates the process by displaying an **Application Failed** message to the user.

3. **Video Upload:**
    - Upon successful application, the user is required to **Upload 3 Videos** through the **Video Upload** module. This step ensures that the required multimedia input for further analysis is provided.

4. **Video Analysis:**

    - The uploaded videos are then forwarded to the **Database**, where the system conducts an in-depth **Video Analysis** to extract

relevant data for the recruitment decision-making process.

5. **Alternative Flows:**
    - The diagram clearly distinguishes between two alternate outcomes:
        - **Valid Path:** Leads to successful application processing, video upload, and video analysis.
        - **Invalid Path:** Immediately results in a failure message without proceeding to further steps.

Components Involved:

- **User:** The job applicant interacting with the system.
- **User Interface:** The primary interaction point for job browsing and application submission.
- **Application Form:** Responsible for user data validation.
- **Video Upload:** Facilitates the collection of candidate videos.
- **Database:** Stores data and supports video analysis processes.

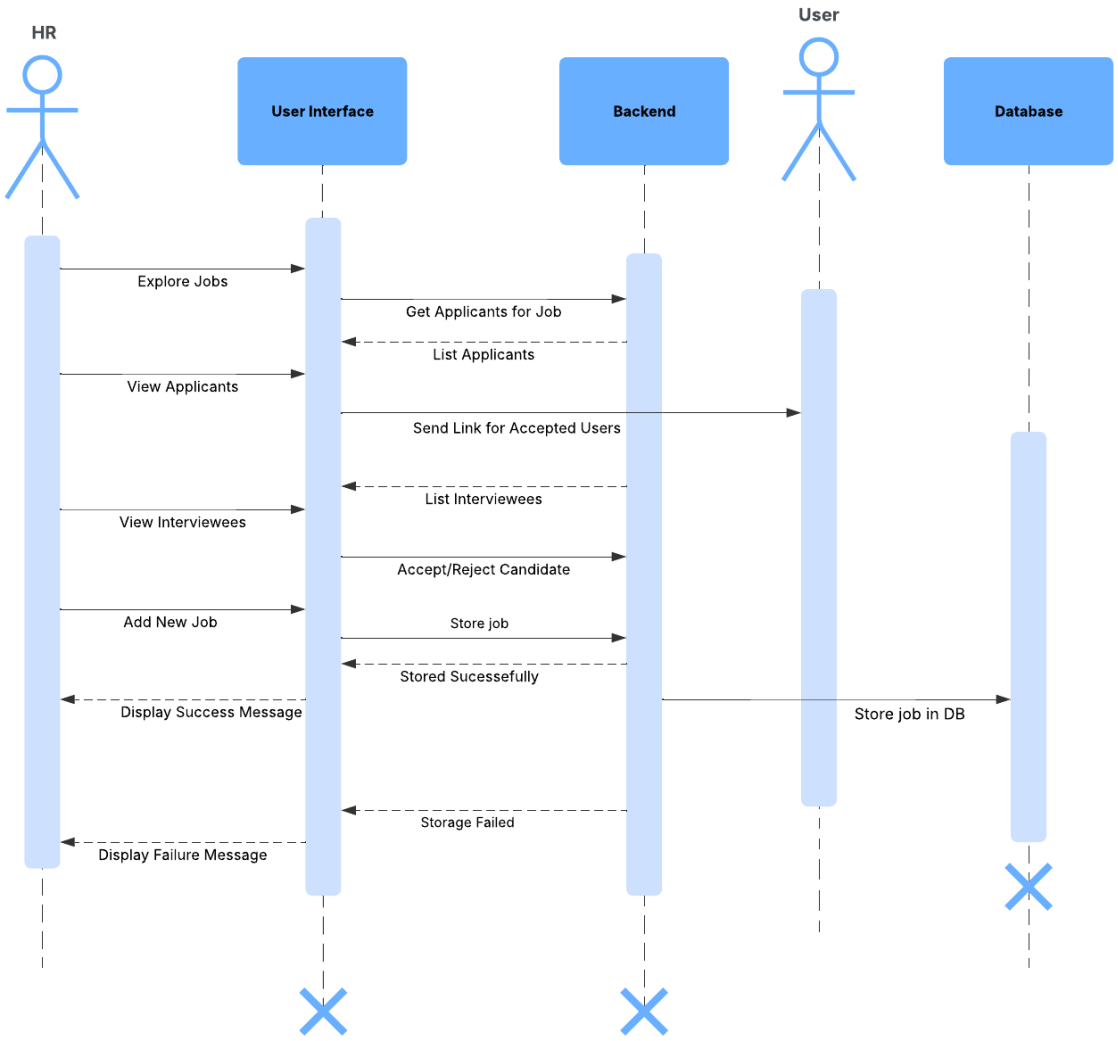


Fig 3.5 HR Sequence Diagram

The HR Sequence Diagram provides a comprehensive visualization of the interaction flow between the Human Resources (HR) personnel, the system components, and the user within the PRVIA recruitment platform. This sequence diagram specifically focuses on the HR operations required for managing job postings, handling applications, and making hiring decisions.

Sequence Flow:

1. **Job Management:**

    - The HR personnel initiates the process by interacting with the User Interface to Explore Jobs and Add New Job opportunities to the system.
    - Once the HR adds a new job, the User Interface sends a request to the Backend to store this job.
    - The Backend communicates with the Database to permanently Store the Job in DB.

    - The outcome of this operation is either:
        - Successful Storage: The system displays a Success Message to the HR personnel.
        - Storage Failure: The system notifies the HR of the failure through a Failure Message.

2. **Application Management:**

    - HR can View Applicants for a specific job through the User Interface.
    - The User Interface requests the Backend to Get Applicants for the Job, which retrieves and Lists Applicants from the system.

- After initial screening, HR sends interview invitations to selected applicants via the Send Link for Accepted Users process.

3. **Interview Management:**
    - HR proceeds to View Interviewees through the system.
    - The Backend Lists Interviewees to the HR for further evaluation.
    - Based on the interview results, HR can Accept or Reject Candidates directly via the User Interface, with the system updating the status accordingly.

Components Involved:

- **HR:** The recruiter or hiring manager who oversees job postings and candidate evaluations.
- **User Interface:** The system's front-end portal through which HR accesses functionalities.
- **Backend:** The server-side component that processes HR requests and coordinates database transactions.
- **Database:** The storage system that holds job postings, applicant data, and interview records.
- **User:** Represents the job applicant interacting with the system (passively involved in this diagram, mainly affected by backend decisions).

3.2.4 Database Diagram

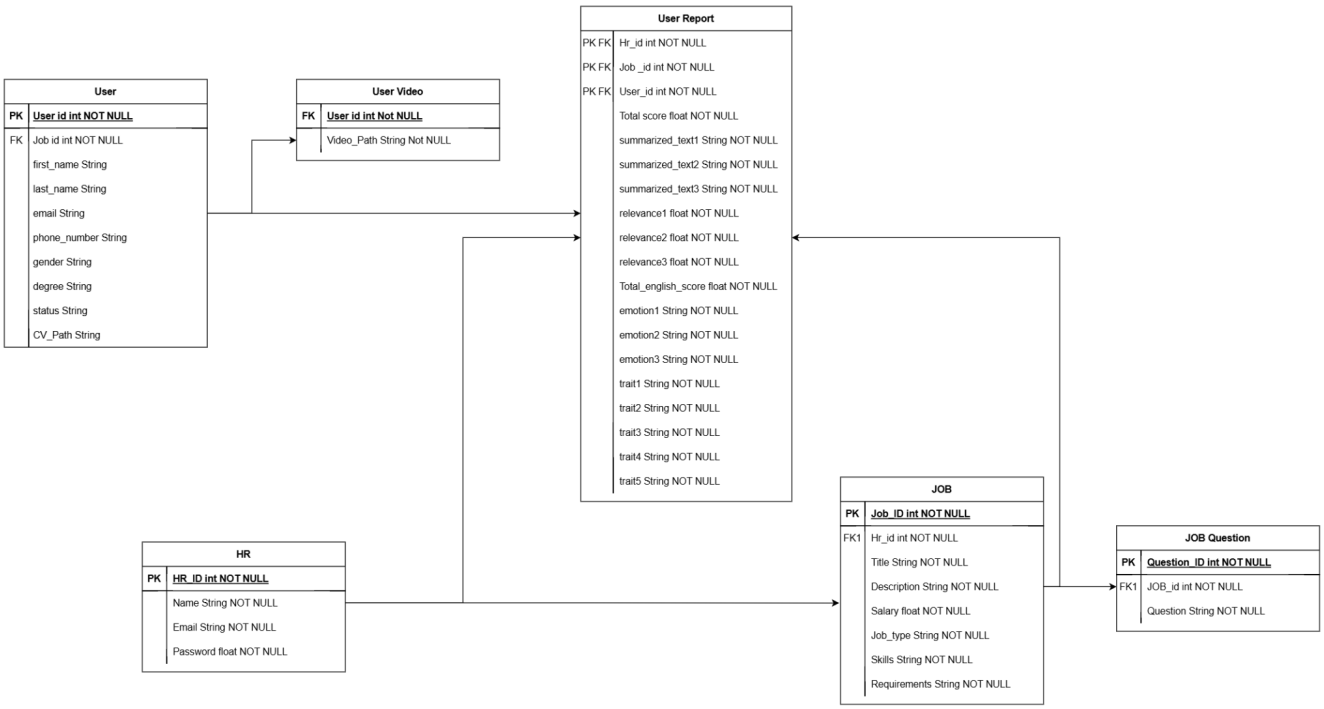


Fig 3.6 Database Diagram

User: Stores personal, contact, and application details of job applicants.

User Video: Contains file paths to users' submitted video interviews.

User Report: Holds evaluation results including scores, emotions, and traits for each user's video.

HR: Stores information about HR users managing job postings and evaluations.

JOB: Represents job listings created by HR with related details and requirements.

JOB Question: Contains interview questions linked to specific job listings.

Chapter 4

Implementation

This chapter details the implementation of the Pre-Recorded Video Interview Analysis (PRVIA) system, focusing on the functions, techniques, algorithms, and technologies used to achieve automated evaluation of candidate performance in pre-recorded job interviews. The system integrates three core modules—Audio, Text (NLP), and Video (Computer Vision)—to provide a comprehensive assessment of pronunciation, semantic content, and visual behavior. Each module employs specialized frameworks and techniques tailored to its specific tasks.

4.1 Audio Module

The Audio Module implements the English Automatic Pronunciation Assessment system, designed to evaluate spoken English proficiency in pre-recorded job interviews. It processes audio extracted from interview videos and corresponding text inputs to produce scores for pronunciation quality, supporting the project's goal of objective, data-driven candidate screening. The implementation uses advanced deep learning models, multimodal feature alignment, and optimization strategies,

4.1.1 System Functions:

Audio Module performs the following core functions to transform video inputs into pronunciation scores:

- **Audio Extraction from Video**: Audio is extracted from pre-recorded interview videos using the MoviePy library. This step converts video files into 16 kHz mono audio waveforms, which are then passed to subsequent processing stages. The extraction ensures that only the

audio component of the interview is analyzed, isolating spoken content from visual data.

• **Input Preprocessing**: Extracted audio is preprocessed to a fixed length of 5 seconds to ensure uniformity across samples. Audio is sampled at 16 kHz, with longer recordings truncated and shorter ones padded with silence. Corresponding text inputs, representing the expected transcript or question context, are tokenized to a maximum of 16 tokens using **ModernBERT** -transformer tokenizer.

- **Feature Extraction:** Audio features are extracted using a pre-trained **wav2vec-xlsr-53 model**, a multilingual speech encoder that captures phonetic and acoustic patterns. The model outputs a sequence of hidden states, from which the last five are retained to preserve temporal context. Text features are derived using ModernBERT, a transformer-based language model, which generates contextual embeddings from tokenized text. Similarly, the last five hidden states are extracted to represent semantic and syntactic information.
- **Model Training:** The system evaluates pronunciation across five metrics: accuracy (correctness of phonetic articulation), fluency (smoothness and naturalness), completeness (coverage of expected content), prosody(intonation and rhythm), and total score (overall quality). These assessments are performed by processing combined audio and text features through dedicated regression heads, producing scores ranging from 0 to 1 for each metric.

4.1.2 Techniques and Algorithms:

This module employs a series of advanced techniques and algorithms to train the model and process audio and assess candidate pronunciation quality, with pre-trained models and multimodal architectures. The

model was trained on the Speechocean762 dataset [?], which includes 5,000 English utterances from 250 non-native speakers, split into 2,500 for training and 2,500 for testing:

- **ASR wav2vec-xlsr-53 Model**: The pre-trained wav2vec-xlsr-53 model, developed through self-supervised learning on multilingual audio datasets,extracts audio features from 16 kHz waveforms. It generates a sequence of 1024-dimensional embeddings, capturing phonetic and acoustic patterns. The last five hidden states are concatenated to form a rich representation of temporal context, ensuring robustness across diverse accents.
- **ModernBERT Model**: ModernBERT, a transformer-based language model, processes tokenized text inputs to generate contextual embeddings.The model outputs 768-dimensional embeddings, and the last five hidden states are concatenated to capture semantic and syntactic relationships. During training, ModernBERT's parameters were frozen, with its pre-trained knowledge without updates, to reduce computational cost and focus training on downstream components.
- **Cross-Attention Mechanism**: A cross-attention layer aligns audio and text features by allowing text embeddings to attend to projected speech features. The speech features are first projected from 1024 to 768 dimensions to match the text embedding space. The cross-attention layer with 32 attention heads and a dropout rate of 0.2, enables bidirectional interaction, enhancing the model's ability to correlate linguistic content with acoustic features.
- **Transformer Encoder:** A stack of six Transformer encoder layers processes the cross-attended features. Each layer includes self-attention with 32 heads, a feed-forward network with 2048

hidden units, and layer normalization, with a dropout rate of 0.2. The encoder refines the feature representations, capturing complex dependencies between audio and text modalities, and produces a sequence of 768-dimensional embeddings.

- **Mean Pooling:** Mean pooling is applied to three feature sets: the Transformer encoder outputs, the last five hidden states from wav2vec-xlsr-53, and the last five hidden states from ModernBERT. This reduces the temporal dimension to a fixed-size vector for each set, resulting in 768-dimensional vectors for the encoder output and ModernBERT features ,and a 5120-dimensional vector for wav2vec features (1024×5). These vectors are concatenated to form a 2560-dimensional combined feature vector that will further be used for output generation.
- **Multi Regression Head Prediction**: Five separate regression heads predict scores for accuracy, fluency, completeness, prosody, and total score. Each head is a multi-layer perceptron (MLP) with varying architectures: three layers ($2560 \rightarrow 1024 \rightarrow 256 \rightarrow 1$) for accuracy, fluency, and prosody, and three layers ($2560 \rightarrow 512 \rightarrow 128 \rightarrow 1$) for completeness and total score. The MLPs use GELU activation and a dropout rate of 0.2 to prevent overfitting, producing scores in the range [0, 10].
- **Training Optimization and Loss:** The model is trained using the AdamW optimizer with a learning rate of 0.001, cosine learning rate decay, and weight decay of 0.0001. Only the cross-attention layer, Transformer encoder, some layers from the Wav2vec model and regression heads were trained, as ModernBERT was frozen. The loss function combines Mean Squared Error (MSE) for score prediction with correlation-aware regularization to maximize Pearson correlation coefficient (PCC) between predicted and

ground-truth scores. Training was conducted for 50 epochs with a batch size of 16, taking approximately 4 hours on GPU hardware.

4.1.3 Technologies:

- **wav2vec-xlsr-53-english Pre-trained Model**: This multilingual speech encoder, pre-trained on large-scale audio datasets, provides high-quality audio features without requiring extensive labeled data. Its self-supervised learning approach ensures robustness across diverse accents.
- **ModernBERT Pre-trained Model**: A state-of-the-art transformer-based language model ,which was published in December 2024, pre-trained on vast text corpora, a massive dataset of 2 trillion English tokens, including web documents, code, and scientific literature. This diverse training data makes it more robust and versatile than traditional BERT models, which were primarily trained on Wikipedia,enhances text feature extraction with contextual understanding and improves multimodal combination features.
- **MoviePy Library**: The MoviePy library is used to extract audio from video inputs, converting video files into 16 kHz mono waveforms. This technology enables seamless integration of video-based interview data into the audio processing pipeline.
- **PyTorch Framework**: PyTorch was used to build and train the Pronunciation Assessment model components, offering flexibility in designing custom layers (e.g., cross-attention,Transformer Encoder) and optimizing performance on GPU hardware. Its dynamic computation graph facilitated rapid prototyping and experimentation.

- **Transformer Architecture**: The implementation adopts the Transformer architecture,but with custom layers and self including self-attention and encoder layers, to handle sequential data efficiently and capture long-range dependencies.

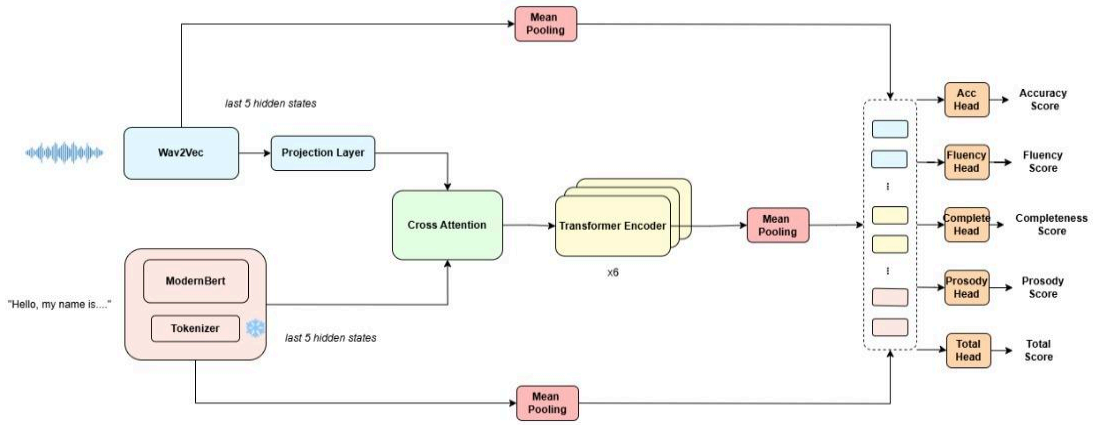


Figure 4.1: Audio Model Architecture

4.1.4 Dataset:

The **Speechocean762 dataset**, A non-native English corpus for pronunciation scoring task, serves as the foundation for training and evaluation. It aims to provide a free public dataset for the pronunciation scoring task. Key features:

- It is available for free download for both commercial and non-commercial purposes.
- The speaker variety encompasses young children and adults.
- The manual annotations are in multiple aspects at sentence-level, word-level and phoneme-level.

This corpus consists of 5000 English sentences. All the speakers are non-native, and their mother tongue is Mandarin. Half of the speakers are children, and the others are adults. The information of age and gender are provided.The data is evenly split into **2,500 training samples and 2,500 testing samples.** Designed for pronunciation assessment, it includes a diverse range of accents and speaking styles, making it suitable for developing robust models to evaluate English proficiency across varied candidate backgrounds.

It was labeled by five experts to make the scores. To avoid subjective bias, each expert scores independently under the same metric.The experts score each audio at three levels: **phoneme-level, word-level, and sentence-level:**

Phoneme level: Score the pronunciation goodness of each phoneme within the words. The Score range: 0 - 2:

- 2: pronunciation is correct
- 1: pronunciation is right but has a heavy accent
- 0: pronunciation is incorrect or missed

Word level: Score the accuracy and stress of each word's pronunciation.The score range 0 - 10:

- **Accuracy:**
    - 10: The pronunciation of the word is perfect
    - 7-9: Most phones in this word are pronounced correctly but have accents
    - 4-6: Less than 30% of phones in this word are wrongly pronounced
    - 2-3: More than 30% of phones in this word are wrongly pronounced. In another case, the word is mispronounced as some

other word. For example, the student mispronounced the word "bag" as "bike"
    - 1: The pronunciation is hard to distinguish
    - 0: no voice
- **Stress:**
    - 10: The stress is correct, or this is a mono-syllable word
    - 5: The stress is wrong

Sentence level: Score the accuracy, fluency, completeness and prosodic at the sentence level.

- **Accuracy:** Score range 0-10
    - 9-10: The overall pronunciation of the sentence is excellent, with accurate phonology and no obvious pronunciation mistakes
    - 7-8: The overall pronunciation of the sentence is good, with a few pronunciation mistakes
    - 5-6: The overall pronunciation of the sentence is understandable, with many pronunciation mistakes and accent, but it does not affect the understanding of basic meanings
    - 3-4: Poor, clumsy and rigid pronunciation of the sentence as a whole, with serious pronunciation mistakes
    - 0-2: Extremely poor pronunciation and only one or two words are recognizable
- **Fluency:** Score range 0-10
    - 8-10: Fluent without noticeable pauses or stammering.
    - 6-7: Fluent in general, with a few pauses, repetition, and stammering.
    - 4-5: the speech is a little influent, with many pauses, repetition, and stammering

    - 0-3: intermittent, very influential speech, with lots of pauses, repetition, and stammering
- **Completeness:** Score range: 0.0 - 1.0 The percentage of the words with good pronunciation.
- **Prosodic:** Score range: 0 - 10
    - 9-10: Correct intonation at a stable speaking speed, speak with cadence, and can speak like a native
    - 7-8: Nearly correct intonation at a stable speaking speed, nearly smooth and coherent, but with little stammering and few pauses
    - 5-6: Unstable speech speed, many stammering and pauses with a poor sense of rhythm
    - 3-4: Unstable speech speed, speak too fast or too slow, without the sense of rhythm
    - 0-2: Poor intonation and lots of stammering and pauses, unable to read a complete sentence

```
{
    "000010011": {

        "text": "WE CALL IT BEAR",
        "accuracy": [7.0, 9.0, 8.0, 8.0, 9.0],
        "completeness": [1.0, 1.0, 1.0, 1.0, 1.0],
        "fluency": [10.0, 9.0, 8.0, 8.0, 10.0],
        "prosodic": [10.0, 9.0, 7.0, 8.0, 9.0],
        "total": [7.6, 9.0, 7.9, 8.0, 9.1],
        "words": [
            {
                "accuracy": [10.0, 10.0, 10.0, 10.0, 10.0],
                "stress": [10.0, 10.0, 10.0, 10.0, 10.0],
                "total": [10.0, 10.0, 10.0, 10.0, 10.0],
                "text": "WE",
                "ref-phones": "W IY0",
                "phones": ["W IY0", "W IY0", "W IY0", "W IY0", "W IY0"]
            },
            {
                "accuracy": [10.0, 8.0, 10.0, 10.0, 8.0],
                "stress": [10.0, 10.0, 10.0, 10.0, 10.0],
                "total": [10.0, 8.4, 10.0, 10.0, 8.4],
                "text": "CALL",
                "ref-phones": "K AO0 L",
                "phones": ["K AO0 L", "K {AO0} L", "K AO0 L", "K AO0 L", "K AO0 {L}"],
            },
            {
                "accuracy": [10.0, 10.0, 10.0, 10.0, 10.0],
                "stress": [10.0, 10.0, 10.0, 10.0, 10.0],
                "total": [10.0, 10.0, 10.0, 10.0, 10.0],
                "text": "IT",
                "ref-phones": "IH0 T",
                "phones": ["IH0 T", "IH0 T", "IH0 T", "IH0 T", "IH0 T"]
            },
            {
                "accuracy": [3.0, 7.0, 10.0, 2.0, 6.0],
                "stress": [10.0, 10.0, 10.0, 10.0, 10.0],
                "phones": ["B (EH0) (R)", "B {EH0} {R}", "B EH0 R", "B (EH0) (R)", "B EH0 [L]
                "total": [4.4, 7.6, 10.0, 3.6, 6.8],
                "text": "BEAR",
                "ref-phones": "B EH0 R"
            }
```


Figure 4.2: Audio Dataset Sample

During training, we restricted our analysis to the **utterance-level annotations** available in the Speechocean762 dataset, specifically the scores for **accuracy, fluency, prosody, and completeness** at the sentence level. While the dataset also provides detailed **phoneme-level** and **word-level** labels—including pronunciation accuracy, stress, and phoneme-specific scoring—we did not utilize these granular annotations. This decision was driven by the nature of our task: in job interview scenarios, the primary concern is **overall spoken communication quality**, not fine-grained phonetic

feedback. Our system is designed to deliver **sentence-level evaluations** that align with how HR professionals assess candidate speech focusing on intelligibility, fluency, and coherence rather than linguistic subtleties.

4.1.5 Experimental Results:

The performance of the Audio Module was evaluated using two key metrics: **Mean Squared Error (MSE)** as the primary evaluation criterion, and **Pearson Correlation Coefficient (PCC)** to facilitate comparison with existing literature.

Mean Squared Error (MSE): MSE quantifies the average squared difference between the predicted scores and the actual ground truth scores $(\widehat{y_i})$, (y_i) across all test samples $i = 1$ to n, and for each of the 5 pronunciation metrics $j = 1$ to 5 . To capture the model's overall performance across all five traits, we compute the final MSE as the average MSE across all dimensions, using the formula **4.1**:

$$\text{MSE} = \sum_{j=1}^{5} \frac{1}{N} \sum_{i=1}^{N} \left(y_i^j - \hat{y}_i^j \right)^2 \tag{4.1}$$

Pearson Correlation Coefficient (PCC): Measures the strength of the linear relationship between predicted and ground truth scores:Here, $\bar{y}$ and $\bar{\hat{y}}$ represent the means of the ground truth and predicted scores, respectively. Higher PCC values (closer to 1) indicate stronger agreement between model predictions and human ratings.The ***(PCC)*** is defined by the following formula:

$$PCC = \frac{\sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N} (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^{N} (y_i - \bar{y})^2}} \quad (4.2)$$

PRVIA outperforms previous systems by combining deep acoustic and contextual modeling in an end-to-end, alignment-free architecture tailored for job interview evaluation. Unlike GOPT by Gong et al., (2022) [15], which uses alignment-dependent GOP features to assess multiple speech attributes, PRVIA removes this dependency and instead leverages wav2vec-xlsr-53 for acoustic features and ModernBERT for semantic representation. Compared to GOPT, **PRVIA achieved stronger performance in fluency (PCC = 0.790 vs. 0.753) and total score (PCC = 0.717 vs. 0.742),** while delivering comparable results in other categories.

Furthermore, relative to the LLM-based system proposed by Fu et al. (2024) [17], which focuses on predicting only fluency and accuracy using a multi-modal LLM architecture, PRVIA extends the scope of evaluation to include prosody and completeness, providing more comprehensive insights. **PRVIA attained a higher PCC in fluency (0.790 vs. 0.777) and matched accuracy scoring performance (0.693 vs. 0.713),** despite using a simpler, non-decoder-based structure. This demonstrates the effectiveness of PRVIA's design in balancing interpretability, performance, and efficiency for practical recruitment scenarios.

Table 4.1: Audio Model Results

Model	PCC Acc ↓	PCC Flu ↑	PCC Pro ↑	PCC Comp ↓	PCC total ↓
GOPT	**0.714**	0.753	0.760	**0.155**	**0.742**
LLM-Based	0.713	0.777	-	**-**	**-**
PRVIA	0.693	**0.790**	**0.770**	0.068	0.717

Each input sample in the audio module yields five individual subscores: **accuracy**, **fluency**, **prosody**, **completeness**, and a **total score**. To compute a single, interpretable score for each candidate's spoken response, these subscores are combined using a **weighted average**, as defined in the formula below (**4.3**).The weighting reflects the relative importance of each metric:This formula is designed to reflect widely accepted **global speaking assessment practices** and aligns with frameworks established by **IELTS**, **TOEFL .**

- **Accuracy (20%)**, which refers to the correct use of grammar and vocabulary, is still important but weighted slightly lower. TOEFL and IELTS recognize that minor grammatical errors often do not obscure meaning In spoken assessments, especially in real-world interview settings, intelligibility and delivery tend to outweigh perfect syntax.
- **Fluency (30%)** is given the highest weight due to its critical role in conveying ideas clearly and smoothly. IELTS allocates 25% to "Fluency and Coherence"], and TOEFL's Delivery descriptor emphasizes "fluid expression" and "well-paced flow". Fluency not only affects intelligibility but also reflects the speaker's coherence and thought organization.

- **Prosody (25%)** which includes pronunciation, intonation, and rhythm—is equally emphasized. IELTS explicitly assess these features , providing detailed rubrics on stress patterns, native-like rhythm, and hesitations. Research also shows that suprasegmental features like stress and intonation account for more variance in perceived fluency than phoneme-level accuracy
- **Completeness (25%)** corresponds to the idea of task fulfillment or topic development. TOEFL penalizes responses that lack content or fail to fully address the question . By including this component with a substantial weight, the system ensures that answers are not only fluent but also relevant and complete.

$$Weighted\ score = (0.2 * accuracy) + (0.3 * fluency) + (0.25 * prosodic) + (0.25 * completeness)$$

4.2 Text Module

4.2.1 Personality Traits from Text

Personality traits can be inferred from linguistic patterns in spoken or written language. Research has shown strong correlations between word choice and the Big Five personality dimensions[1]—for example, extraverts tend to use more positive emotion words, while individuals high in neuroticism frequently use first-person pronouns. In this module, we leverage natural language processing (NLP) techniques and pre-trained language models to analyze textual responses from interviewees and predict their personality traits based on these linguistic cues. Additionally, we incorporate visual signals—such as facial expressions, gaze behavior, and body movements—captured during the interview to enrich the personality

inference process. By combining both text and vision modalities, the system enables a more holistic and accurate estimation of an individual's personality.

A) Dataset:

We used the publicly available Essays Dataset, a well-established resource in personality research and NLP. It contains 2,468 stream-of-consciousness essays written by psychology students, each annotated with binary labels for the Big Five personality traits: Openness (OPN), Conscientiousness (CON), Extraversion (EXT), Agreeableness (AGR), and Neuroticism (NEU). The dataset is text-only and has been widely adopted for studying the relationship between language and personality traits [2].

Table 4.2 Text Dataset Distribution

Distribution of the Dataset				
EXT	**OPN**	**NEU**	**AGR**	**CON**
1276	1271	1233	1310	1253
51.7%	49.9%	53.1%	50.79%	51.52%

B) Feature Extraction

We used large language models like BERT [3] and ALBERT [4] to extract features from text and tested how well they performed with different settings and preprocessing methods. To find the best setup, we ran many experiments and carefully tested different model options and configurations.

First, we looked at how using different layers of the model affects feature quality, since earlier research [5] shows that each layer captures different types of information—some layers focus on grammar, while others focus on meaning. Then, we tried out different ways to represent tokens, including using the [CLS] token and averaging all the token embeddings.

We also tested several ways to prepare the text before feeding it into the model, such as trying different tokenization and cleaning methods. Finally, we looked at how to handle the 512-token limit by comparing three approaches: keeping the beginning of the text, keeping the end, or combining the start and end (for example, the first 256 and last 256 tokens). These tests helped us find the best combination for our task and gave us useful insights into what matters most when using these models in other tasks.

C) Experiments Configuration

Since the input to our models is the embedding output from pre trained language models, we explored a wide range of combinations between embedding models (e.g., BERT, AlBert, Glove) and classical as well as neural machine learning classifiers to identify the optimal configuration for trait prediction. For each psychological trait, a separate binary classification model was trained to determine whether an individual exhibits the trait. We initially focused on a multi-layer perceptron (MLP) architecture comprising three hidden layers with ReLU activations, batch normalization, and dropout layers to address potential overfitting. The model was trained using the Adam optimizer [7] with a categorical cross-entropy loss function. To improve generalization and stability, we employed 10-fold cross-validation, early stopping, and learning rate reduction on plateau. Among the various configurations, the MLP combined with BERT-Large embeddings [3] yielded the highest accuracy across most traits. We also experimented with traditional classifiers such as Support Vector Machines (SVMs), Random

Forests, and Logistic Regression. However, these models generally underperformed compared to neural approaches when used with contextual embeddings, aligning with findings in prior work [6].

D) Experimental Results

The performance of various model configurations for predicting the Big Five personality traits—Openness (OPN), Conscientiousness (CON), Agreeableness (AGR), Neuroticism (NEU), and Extraversion (EXT)—is summarized in the table below. The evaluation metric used is classification accuracy, reported as percentages, based on 10-fold cross-validation on the Essays Dataset.

Table 4.3 Personality Traits Text Model Results

MODEL	OPN	CON	AGR	NEU	EXT
Bert-Base+MLP	65	64	63	61.79	61.54
Bert-Large+MLP	**68.02**	**65.18**	**63.41**	**61.54**	**62.75**
alBert-Base+MLP	66	61.94	60	58.13	60.32
Bert-Large+SVM	62.34	54.87	53.44	58.13	55.46

The BERT-Large + MLP configuration consistently outperformed other models across all five personality traits, achieving the highest accuracies: 68.02% for OPN, 65.18% for CON, 63.41% for AGR, 61.54% for NEU, and 62.75% for EXT. This suggests that the deeper architecture of BERT-Large, combined with the expressive power of the multi-layer perceptron (MLP), effectively captures the linguistic patterns associated with personality traits. The ALBERT-Base + MLP model performed competitively but generally yielded lower accuracies

compared to BERT-Large + MLP, likely due to its parameter-efficient design trading off some representational capacity.

Traditional classifiers, such as the Support Vector Machine (SVM) paired with BERT-Large embeddings, underperformed significantly, with accuracies ranging from 53.44% (AGR) to 62.34% (OPN). This aligns with prior findings [6] that neural approaches, particularly when combined with contextual embeddings like those from BERT, are better suited for personality trait prediction from text. The results highlight the importance of leveraging advanced neural architectures and contextual embeddings for improved performance in this task.

Among the preprocessing strategies, using the [CLS] token representation and averaging token embeddings provided comparable results, with the former slightly outperforming in most cases. Handling the 512-token limit by combining the first and last 256 tokens of the text proved to be the most effective approach, preserving critical contextual information from both the beginning and end of the essays. These findings provide valuable insights for optimizing model configurations in similar NLP-based personality prediction tasks.

E) Comparison with Other Research

Table 4.4 compares the performance of our study (PRVIA, using BERT-Large + MLP) with results from two prior studies on the same task of predicting Big Five personality traits from text using the Essays Dataset.

Table 4.4 Personality Traits Results Comparison

Paper Name	OPN	CON	AGR	NEU	EXT
[7] Novel Curriculum Learning Strategy using Class-Based TF-IDF for Enhancing Personality Detection in Text."	66.97	64.94	63.31	63.29	66.39
[8] "Bottom-up and Top-down: Predicting Personality with Psycholinguistic and Language Model Features."	64.60	59.20	58.80	60.50	60.00
PRVIA (BERT-Large + MLP)	**68.02**	**65.18**	**63.41**	**61.54**	**62.75**

4.2.2 Checking the Answer Relevance.

In the context of video interview analysis to facilitate the hiring process, ensuring that candidate answers, transcribed from video interviews, align with the linguistic patterns and contextual intent of the questions. Relevance checking verifies how well the transcribed answers correspond to the question and textual content, directly impacting the reliability of candidate evaluations and their final scores in the hiring process. A key challenge is the difficulty in definitively discriminating between relevant and irrelevant answers due to the subjective nature of linguistic cues, transcription errors, and the complexity of contextual understanding [18]. Irrelevant or misaligned answers can lead to inaccurate assessments of candidate suitability, potentially affecting hiring decisions. Below, we detail the approaches tested, their configurations, underlying equations, insights gained, and the limitations in distinguishing relevant from irrelevant answers, incorporating techniques tailored to the hiring context using transcribed answers and references to relevant research.

A. Text Similarity Techniques

Cosine Similarity

The first approach tested was computing the cosine similarity between the question and answer to set a threshold for determining relevance. The question and answer texts are preprocessed by removing stopwords, normalizing case, and tokenizing. Then, they are converted into vector representations using a pre-trained embedding model, such as BERT, GloVe, or all-MiniLM-L6-v2, a lightweight sentence transformer suitable for semantic embeddings. The cosine similarity is calculated using the cosine similarity

$$\cos(\theta) = \frac{A \cdot B}{|A||B|} \quad (4.4)$$

, where (A) and (B) are the embedding vectors of the question and answer, respectively. A threshold, typically between 0.7 and 0.9, is set to classify the answer as relevant or irrelevant based on the similarity score. This method is computationally efficient and captures syntactic overlap, such as shared words like “teamwork” or “project.” However, it faces significant limitations. Answers often contain more tokens than questions, leading to inflated similarity scores due to shared generic terms. The method relies heavily on syntactic similarity rather than semantic understanding, resulting in similar scores for both relevant and irrelevant answers. For example, for the question “Tell us about your teamwork experience,” an off-topic answer discussing technical skills might score high due to overlapping terms like “work” or “project.” Determining an appropriate threshold was challenging because the score distributions for relevant and irrelevant answers overlapped, making it difficult to definitively distinguish them.

Semantic Similarity and TF-IDF

To address the limitations of the basic cosine similarity approach, a more sophisticated method was tested, combining semantic similarity with keyword similarity. This approach begins by generating a model answer for the question, either manually crafted by hiring experts or generated using a language model like GPT-4 or BERT. For instance, for the question “Describe a time you solved a problem,” a model answer might be: “I identified a bottleneck in our workflow, collaborated with my team, and implemented a new process that improved efficiency by 20%.” Both the candidate’s answer and the model answer are embedded using a sentence transformer, such as all-MiniLM-L6-v2, to capture semantic meaning. Cosine similarity is computed between these embeddings.

Additionally, keyword similarity is incorporated using Term Frequency-Inverse Document Frequency (TF-IDF). TF-IDF extracts key terms from the question and answer, with the score calculated as

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \cdot \log\left(\frac{N}{\text{DF}(t)}\right) \quad (4.5)$$

, where $(\text{TF}(t, d))$ is the term frequency of term (t) in document (d),$(\text{DF}(t))$ is the document frequency, and (N) is the total number of documents. Cosine similarity is then computed between the TF-IDF vectors of the question and answer. The semantic and keyword similarity scores are combined using a weighted sum, expressed as $\text{Score} = w_1 \cdot \text{SemanticScore} + w_2 \cdot \text{KeywordScore}$, with weights typically set to 0.7 for semantic similarity and 0.3 for keyword similarity to prioritize contextual understanding. This method improves on the previous approach by capturing semantic intent and emphasizing domain-specific keywords like “collaboration” or “problem-solving.” However, limitations persist. The quality of the model answer significantly affects the

semantic similarity score, and manually crafting or generating accurate model answers is resource-intensive. Answers that repeatedly use keywords, such as "teamwork," but lack depth can score high due to TF-IDF, leading to false positives. Determining a reliable threshold for relevance remained problematic, as the combined scores did not clearly separate relevant and irrelevant answers.

Cross-Encoder Model

The next approach tested was a cross-encoder model, specifically a model like cross-encoder/ms-marco-MiniLM-L-12-v2, designed to score the relevance of a question-answer pair directly. The question and answer are concatenated into a single input, typically in the format "[CLS] Question [SEP] Answer [SEP]," and processed through a transformer-based architecture, such as BERT, which models interactions between the question and answer tokens. The model outputs a single relevance score between 0 and 1, which can be thresholded (e.g., 0.5) to classify relevance. If a hiring-specific dataset of labeled question-answer pairs is available, the model can be fine-tuned to improve performance. This approach is powerful because it captures deep contextual relationships between the question and answer, unlike embedding-based methods that compute similarity independently. However, the results were disappointing. The model performed poorly, likely due to insufficient fine-tuning data tailored to the hiring context, as well as transcription errors in the answers that disrupted input quality. The cross-encoder also proved computationally expensive, requiring more resources than cosine similarity methods, which made it less practical for large-scale analysis. For example, nuanced but relevant answers with unconventional phrasing were often misclassified as irrelevant, highlighting the model's sensitivity to training data and input quality.

LLM-Based Prompting

The final approach, which is currently in use, leverages prompt engineering with a large language model, Gemini, to evaluate answer relevance. Multiple prompt variations were tested, including those with additional context like role requirements, but the most effective prompt was:

- **Prompt Design:**

```
"""You are an **AI-powered interviewer assistant** evaluating
the relevance of an interview question.

    ### **Task:**
    - Determine how **relevant** the question is to assessing
the candidate.
    - Use the provided **candidate's answer** to help
contextualize the evaluation.
    - Provide a **single numerical score (0-10)** where:
      - **0** = Completely irrelevant
      - **10** = Highly relevant

    ### **Interview Question:**
    "{question}"

    ### **Candidate's Answer:**
    "{txt}"

    ### **Output Format:**
    Provide **only** a single number between **0 and 10**
representing the relevance score.
    """
```

- **Output Format:**

Provide only a single number between 0 and 10 representing the relevance score." The language model processes this prompt, evaluates the semantic and contextual alignment of the answer to the question, and outputs an integer score from 0 to 10. Scores are validated against human judgments or ground-truth labels to ensure reliability. A threshold, such as scores $\geq$7, is applied to classify answers as relevant, with the threshold adjustable based on hiring requirements. This method excels at capturing nuanced relevance and handling indirect or unconventionally phrased answers. For example, for the question "Describe a time you solved a problem," an answer about a technical fix that addresses problem-solving might score 8/10, while an unrelated answer about hobbies would score 2/10. The prompt outperformed other versions by focusing on the core question-answer relationship and providing clear instructions for scoring. However, limitations include dependence on the language model's quality and potential biases, as well as vulnerability to transcription errors that can mislead the model. Inconsistent scoring across different models or slight prompt variations is another concern, requiring careful prompt tuning to ensure reliability.

B) Comparison of Text Similarity Methods

Each approach tested for relevance checking in video interview analysis presented distinct difficulties, highlighting the complexity of evaluating candidate answers in the hiring context. The cosine similarity approach, while computationally efficient, struggled with over-reliance on syntactic overlap, leading to inflated scores for irrelevant answers due to shared generic terms and difficulty in setting a reliable threshold due to overlapping score distributions. The semantic similarity with model answer and TF-IDF method improved by incorporating semantic intent and keyword emphasis, but its effectiveness was hindered by the resource-intensive need for high-quality

model answers and the risk of false positives from keyword-heavy but shallow responses. The cross-encoder model, despite its potential to capture deep contextual relationships, underperformed due to insufficient fine-tuning data for the hiring context, transcription errors, and high computational costs, making it impractical for large-scale use. The prompt engineering approach with a large language model faced challenges related to model quality, potential biases, and sensitivity to transcription errors, but it mitigated these through careful prompt design and validation against human judgments. The final approach chosen is the prompt engineering method

4.2.3 Text Summarization

Reviewing long interview recordings can be time-consuming for HR professionals, especially when dealing with a large pool of applicants. To address this challenge, this task focuses on implementing **text summarization techniques** to automatically generate concise summaries of candidate responses extracted from their video interviews. The core idea is to reduce each candidate's spoken content into a short textual summary that captures the most important points discussed, such as their qualifications, experience, goals, and personal attributes. This enables HR personnel to efficiently preview key information without having to watch the full videos, thereby streamlining the candidate evaluation process and saving considerable time and effort. In order to achieve this goal, two main functions have to be implemented first : Extracting audio from video then transcribe the extracted audio into text.

1. Audio Extraction from Video

To begin the summarization process, the audio must first be extracted from the candidate's video. This step is implemented using the **MoviePy** Python

library, which allows efficient handling of video files. The extracted audio serves as the input for the next phase which is the transcription.

2. Speech-to-Text Transcription using Whisper

After extracting the audio from each candidate video, the next step is to convert the speech into text. This process is handled using **Whisper**, an open-source automatic speech recognition (ASR) model developed by OpenAI.

The first task performed by Whisper is **language detection**. This step ensures that only audio segments spoken in **English** are processed further. If Whisper detects that the spoken language is not English, the transcription step is skipped. This filtering is essential to maintain consistency in the summarization stage and to avoid processing content that could lead to poor or inaccurate summaries due to language mismatch.

For English-language audio, Whisper proceeds with the **transcription** step. Specifically, we use the **Whisper Medium model**, which provides a strong trade-off between accuracy and computational efficiency. The choice of the Medium model was informed by the Whisper research paper, which compares model performance across **14 benchmark datasets**. The Medium model consistently achieved **low Word Error Rates (WER)** across most datasets, making it a reliable option. Although larger models like Whisper Large have even lower WERs, they require significantly more resources. Given our computational constraints, the Medium model was the most suitable choice for this project.

The final output of this stage is a **text transcript** of the candidate's spoken content, provided that the detected language is English.

A) Dataset

During the development of the summarization component, we searched extensively for publicly available datasets containing interview-style speech transcripts paired with reference summaries. However, we found that such datasets were either unavailable or unsuitable for our specific task which focuses on **candidate-style self-presentations** commonly found in job interviews.

As a result, we decided to **create our own custom dataset** tailored to the requirements of this project. The dataset consists of **800 rows**, each simulating a person talking about themselves, their professional background, experiences, or personal aspirations, closely reflecting what candidates typically share during job interviews.

To generate this data efficiently and realistically, we used **OpenAI's GPT-4 model**, which was prompted to simulate diverse, natural-sounding candidate responses along with their corresponding summaries. Care was taken to ensure that the generated data maintained **realism, linguistic variation, and topic diversity**, thereby providing a robust basis for model evaluation.

Each entry in the dataset contains:

- **Transcripts**: A GPT-4-generated passage representing the candidate's spoken-style response.

- **Reference Summary**: A condensed summary highlighting the key information from the transcript.

B) Summarization Models

To identify the most effective summarization approach for condensing candidate transcripts, we evaluated **multiple pre-trained models** using the custom dataset we created (Section 4.2.3.3). This comparative evaluation helped us determine which model delivers the best summaries in terms of informativeness, fluency, and conciseness.

One of the primary models used in this study is:

1- BART-Large-CNN

We used **BART-Large**, a powerful transformer-based model that was pre-trained on large-scale English text and later fine-tuned on the **CNN/DailyMail** summarization dataset. BART (Bidirectional and Auto-Regressive Transformers) combines the strengths of **BERT-like encoders** and **GPT-like decoders**, making it particularly effective for abstractive summarization tasks.

2- FLAN-T5 Large

Another model we evaluated for the summarization task is FLAN-T5 Large, an instruction-tuned version of the T5 (Text-to-Text Transfer Transformer) model. FLAN-T5 builds on Google's original T5 architecture, which treats every NLP task as a text-to-text problem (i.e., both input and output are in natural language). What distinguishes FLAN-T5 from standard T5 is its fine-tuning on a large and diverse collection of instruction-based tasks, enabling it to follow prompts more effectively, even in zero-shot or few-shot scenarios.

We used the FLAN-T5 Large variant, which offers a good balance between performance and efficiency.

FLAN-T5 was evaluated directly using a simple prompt :

```
"""
You are an AI assistant specializing in summarizing job
interview responses. Your task is to generate a *clear,
well-structured, and natural-sounding summary* of the
candidate's answer while keeping it concise and
professional.
 - Ensure the summary is written *in a natural, flowing
paragraph*, not   bullet points.
 - Maintain *the key ideas* from the response while
eliminating unnecessary details.
 - Keep the tone *formal, coherent, and human-like* as if
it were written by a professional recruiter.


 Here is the candidate's response:
 "{transcript}"
Now, provide a well-written summary in paragraph form that
retains the most important information from the response.
"""
```

3- Gemini 2.0 Flash

We also evaluated Gemini, a state-of-the-art large language model developed by Google DeepMind. Like the other models in our study, Gemini was prompted using the same structured instruction designed to simulate how a professional recruiter would summarize a candidate's response.

Despite being a general-purpose model, Gemini demonstrated strong capabilities in text summarization, particularly when guided by clear, task-specific instructions.

Evaluation Metrics Used

- **ROUGE (Recall-Oriented Understudy for Gisting Evaluation)**: ROUGE is a set of metrics used to evaluate the quality of automatically generated summaries by comparing them to human-written reference summaries. It does this by measuring the overlap of words or phrases (n-grams), and sequences between the two texts.

- **ROUGE-1**

  Measures how many individual words in the generated summary match the reference.

  - ROUGE-1 Recall = $\frac{(Number\ of\ overlapping\ unigrams)}{(Total\ unigrams\ in\ reference\ summary)}$
  - ROUGE-1 Precision = $\frac{(Number\ of\ overlapping\ unigrams)}{(Total\ unigrams\ in\ generated\ summary)}$
  - ROUGE-1 F1 Score = $2 * \frac{(Precision * Recall)}{(Precision + Recall)}$

- **ROUGE-2**

  Evaluates how many word pairs (bigrams) in the generated summary match those in the reference.

  - ROUGE-2 Recall = $\frac{(Number\ of\ overlapping\ bigrams)}{(Total\ bigrams\ in\ reference\ summary)}$

- ROUGE-2 Precision = $\frac{(Number\ of\ overlapping\ bigrams)}{(Total\ bigrams\ in\ generated\ summary)}$
- ROUGE-2 F1 Score = $2 * \frac{(Precision * Recall)}{(Precision + Recall)}$

- **<u>ROUGE-L</u>**

  Captures fluency by identifying the longest sequence of words that appears in both the generated and reference summaries, without requiring them to be consecutive.

  - ROUGE-L Recall = $\frac{(LCS\ Length)}{(Length\ of\ reference\ summary)}$
  - ROUGE-L Precision = $\frac{(LCS\ Length)}{(Length\ of\ generated\ summary)}$
  - ROUGE-L F1 Score = $2 * \frac{(Precision * Recall)}{(Precision + Recall)}$

- **<u>ROUGE-Lsum</u>**

  A variation of ROUGE-L used specifically for multi-sentence summarization. It computes the LCS over multiple sentences rather than a single pair.

  - ROUGE-Lsum Recall = $\frac{(Length\ of\ Longest\ Common\ Subsequence)}{(Total\ number\ of\ words\ in\ the\ reference\ summary)}$
  - ROUGE-Lsum Precision = $\frac{(Length\ of\ Longest\ Common\ Subsequence)}{(Total\ number\ of\ words\ in\ the\ generated\ summary)}$
  - ROUGE-Lsum F1 Score = $2 * \frac{(Precision * Recall)}{(Precision + Recall)}$

- **BLEU (Bilingual Evaluation Understudy)**: Commonly used in machine translation, BLEU measures the precision of n-gram matches between the generated text and reference text. It's stricter and usually penalizes paraphrasing.

BLEU Score = $Brevity\ Penalty\ (BP)\ \ *\ \ \exp(\sum(Wn\ \ *\ \ log(Pn))$

- **Brevity Penalty (BP):**
  A penalty applied when the generated summary is shorter than the reference summary. It prevents the model from producing very short outputs that might match perfectly but are incomplete.
- **exp:**
  The exponential function, which is used to compute the geometric mean of the n-gram precisions.
- $\sum(Wn\ \ *\ \ log(Pn)$**:**

  This is the weighted sum of the logarithms of the n-gram precisions.
- **w:**
  The weight assigned to each n-gram level. In BLEU-4, each weight is typically **0.25** (equal weight for 1-gram, 2-gram, 3-gram, and 4-gram).
- **p:**
  The precision of n-grams, calculated as:
  - $\mathbf{p_1}$**:** Precision of unigrams = $\frac{(Number\ of\ overlapping\ unigrams)}{(Total\ unigrams\ in\ generated\ summary)}$
  - $\mathbf{p_2}$**:** Precision of bigrams = $\frac{(Number\ of\ overlapping\ bigrams)}{(Total\ bigrams\ in\ generated\ summary)}$

- **p_3:** Precision of trigrams = $\frac{(Number\ of\ overlapping\ trigrams)}{(Total\ trigrams\ in\ generated\ summary)}$
- **p_4:** Precision of 4-grams = $\frac{(Number\ of\ overlapping\ 4-grams)}{(Total\ 4-grams\ in\ generated\ summary)}$

In this project, the BLEU score was calculated using the **BLEU-4 configuration by default**, which considers the overlap of **1-gram, 2-gram, 3-gram, and 4-gram sequences** between the generated summaries and the human-written reference summaries. Each n-gram level is **equally weighted at 25%** in the final score calculation. The BLEU-4 configuration provides a balanced evaluation that captures both short and longer matching phrases, ensuring a more comprehensive assessment of summarization quality.

- **BERTScore**: Unlike ROUGE and BLEU, BERTScore uses contextual embeddings from transformer models to evaluate **semantic similarity** between the generated and reference texts. It is more sensitive to meaning than surface-level word overlap.

  BERTScore Precision = (Sum of maximum similarity scores for each token in the generated summary to tokens in the reference summary) / (Total number of tokens in the generated summary)

  BERTScore Recall = (Sum of maximum similarity scores for each token in the reference summary to tokens in the generated summary) / (Total number of tokens in the reference summary)

  BERTScore F1 = 2 × (BERTScore Precision × BERTScore Recall) / (BERTScore Precision + BERTScore Recall)

Table 4.5 : Summarization Models Performance Comparison

Model	Rouge-1	Rouge-2	Rouge-L	Rouge-L sum	BLEU Score	AverageBERTScore F1
BART-Large-CNN	0.4930	0.2686	0.4345	0.4344	0.1738	**0.9110**
FLAN-T5 Large	0.3530	0.2164	0.3425	0.3423	0.1117	0.8996
Gemini	0.4530	0.2373	0.4073	0.4072	0.1047	0.9079

Important Note on Evaluation Metrics:

During our analysis, we observed that the ROUGE and BLEU scores across all models were relatively low. To investigate this, we reviewed relevant research and confirmed that this is a well-documented limitation in Natural Language Generation (NLG) tasks. Both ROUGE and BLEU heavily rely on surface-level word matching (n-grams) and do not account for semantic meaning. As a result, these metrics tend to penalize summaries that use different phrasing or paraphrasing, even if the content is accurate and semantically aligned.

This issue is especially evident in summarization tasks where multiple valid summaries can be expressed using different words or structures. Therefore, low ROUGE and BLEU scores do not necessarily indicate poor summarization quality—they simply reflect lexical differences between the generated summaries and the reference summaries.

In contrast, BERTScore provides a more reliable assessment because it measures semantic similarity using contextual embeddings, making it more sensitive to meaning rather than exact word overlap.

- **BART-Large-CNN:**
  - Achieved the **highest ROUGE-1 (0.4930)**, **ROUGE-2 (0.2686)**, **ROUGE-L (0.4345)**, and **ROUGE-Lsum (0.4344)** scores.
  - Also recorded the **highest BERTScore F1 (0.9110)**, indicating superior semantic similarity to reference summaries.
  - This suggests that **BART-Large-CNN generated summaries that are both lexically and semantically closest** to the summary references.
- **FLAN-T5 Large:**
  - While FLAN-T5 scored **the lowest in ROUGE and BLEU metrics**, its **BERTScore F1 of 0.8996** shows it still captures meaning relatively well.
  - This model likely paraphrased content more than others, which affects n-gram-based metrics like ROUGE and BLEU.
- **Gemini 2.0 Flash:**
  - Gemini placed in the **middle range** for most ROUGE scores, indicating fairly accurate lexical overlap.
  - Its **BLEU score was the lowest (0.1047)**, possibly due to higher variability in word choices.
  - However, it still maintained a **high BERTScore F1 (0.9079)**, close to BART's, suggesting strong semantic preservation.

It is important to emphasize that **human judgment remains the most critical metric** when evaluating the quality of text summarization. While automatic evaluation metrics such as ROUGE, BLEU, and BERTScore provide useful insights, they do not always capture the semantic coherence, informativeness, or readability of a summary from a human perspective.

During our evaluation, we manually reviewed the summaries generated by the three tested models **BART-Large**, **FLAN-T5 Large**, and **Gemini**. It became evident that the summaries produced by **Gemini** were noticeably superior.

They were more informative, retained key details more effectively, and read more naturally compared to those generated by the other models. This was true even though **BART-Large** achieved slightly higher BERTScore F1 values.

For reference, the **Average BERTScore F1** results were:

- **BART-Large**: 0.9110
- **Gemini**: 0.9079

Despite the minor numerical difference, the qualitative difference observed through human evaluation led us to conclude that **Gemini was the most reliable model** for our summarization task. As a result, we chose Gemini as the preferred model for generating final candidate summary reports.

Sample outputs for each model:

Transcript:

Hello, my name is Nabil Samaham. I am 24 years old. I am from Syria. I am a first year master student in robotics and artificial intelligence. I finished my bachelor degree when I was in Syria and then I moved here to Russia to study my master degree. Actually I have many skills that can help me to get new work in particular companies such as I don't have any problem work under stress, I like to work in teams and I always meet deadlines. I do not like to exceed deadlines and I always on time. I do not like doing sales that much because I think I am not a good person in convincing others for selling, for buying this product from me. If I want to do sales maybe because I want to get money, maybe because if I created or invented a new product and I want to have some investors to take my prototype and start working on it to sell it to as many people as we can for getting money and benefit from this product. Over the past three years the most important achievement I am proud of that I finished my bachelor degree in mechatronics engineering and I became an

engineer and I got the opportunity to travel to Russia and continue my master degree here in Russia.

Gemini 2.0 Flash:

Nabil is a 24-year-old first-year master's student in Robotics and Artificial Intelligence, originally from Syria. He completed his bachelor's degree in Mechatronics Engineering before relocating to Russia for his master's studies. He highlights his ability to work well under pressure, collaborate effectively in teams, and consistently meet deadlines. While he is not particularly drawn to sales roles due to a perceived lack of persuasiveness, he would consider it if it involved securing investment for a product he developed. Nabil considers completing his bachelor's degree and pursuing his master's in Russia as his most significant achievements over the past three years.

BART-Large-cnn:

Nabil Samaham is a first year master student in robotics and artificial intelligence. He says he has many skills that can help him to get new work in particular companies. He doesn't like doing sales because he thinks he is not a good person.

Flan T5:

Nabil Samaham is a first year master student in robotics and artificial intelligence. He is from Syria and he is studying his master degree in Russia

---

Transcript:

Hi, I'm Mario Meza. I've been working as a Field seismologist for around 5 years now. In my last role at Mccarthy LLC, I led several successful projects

and collaborated with cross-functional teams. I really enjoy problem-solving and creating efficient workflows. One challenge I faced was transitioning to a remote setup, but I quickly adapted by organizing regular check-ins and streamlining communication tools. I'm currently looking for opportunities where I can grow and contribute meaningfully.

Gemini 2.0 Flash:

Mario Meza is a Field Seismologist with approximately five years of experience, most recently at Mccarthy LLC, where he led successful projects and fostered collaboration within cross-functional teams. He enjoys problem-solving and workflow optimization. He demonstrated adaptability by successfully transitioning to a remote work environment through proactive communication strategies. Mr. Meza is seeking an opportunity for professional growth and meaningful contribution.

BART-Large-CNN :

Mario Meza has been working as a Field seismologist for around 5 years. In his last role at Mccarthy LLC, he led several successful projects. He is currently looking for opportunities where he can grow and contribute meaningfully.

FLAN T5:

Mario Meza is a Field seismologist. He's looking for a new job

4.3 Video Module

4.3.1 Gaze-Based Cheating Detection

This system detects potential cheating behavior based on gaze direction inferred from facial landmarks using **MediaPipe Face Mesh**. The idea is that a subject is likely cheating if they look away from the screen (i.e., their gaze is not centered) for a sustained duration.

The gaze-based cheating detection system performs the following core functions to infer gaze direction and identify prolonged off-center behavior using facial landmarks extracted from video input:

- **Face Mesh Detection and Landmark Localization:**
  Video frames are processed using the **MediaPipe Face Mesh** model, a lightweight neural network pipeline capable of detecting and localizing 468 facial landmarks per face in real-time. For each frame, the model identifies key facial regions including the eyes and irises. The detection operates directly on RGB input frames and supports real-time inference, optimized for CPU performance.
- **Iris Position and Eye Boundary Extraction:**
  From the predicted landmarks, specific indices corresponding to the left and right eye boundaries (`[33, 133]` and `[362, 263]` respectively) and iris centers (`[468, 473]`) are extracted. The horizontal iris position is then calculated relative to the eye boundaries to estimate gaze direction. This calculation is performed separately for each eye.
- **Gaze Estimation and Center Validation:**
  For each frame, the system computes the **relative horizontal position of the iris** using the equation:
  relative_position = (iris_x - eye_left) / (eye_right - eye_left)
  If the relative position falls within the empirically derived threshold

range of `0.35 < x < 0.65`, the gaze is considered **centered**. Otherwise, it is classified as **non-centered**. A temporal logic mechanism confirms whether the gaze remains centered for at least **0.5 seconds**, or deviates for over **3 seconds**, to reduce false positives.

- **Cheating Behavior Decision Module:**
  A counter tracks the cumulative duration during which gaze is not centered. If this **non-center time** exceeds **3.0 seconds**, the system flags the user as potentially cheating. This decision threshold balances sensitivity and specificity by ignoring brief gaze shifts while responding to sustained gaze deviation.

A) Techniques and Algorithms

The Gaze-Based Cheating Detection Module integrates real-time facial analysis with lightweight geometric computations to infer user attention from eye movements:

- **MediaPipe Face Mesh Model:**
  The system utilizes the **MediaPipe Face Mesh**, a real-time 3D face landmark estimation framework that outputs 468 facial landmarks from RGB input. It employs a two-stage pipeline: first, a modified BlazeFace detector localizes the face, followed by a regression-based mesh model that predicts high-resolution facial geometry. With `refine_landmarks=True`, it additionally estimates five iris landmarks per eye, enhancing the precision of gaze estimation.

- **Landmark-Based Eye and Iris Localization:**
  Specific landmark indices are used to extract the horizontal eye boundary and iris position for both eyes:
  - Left Eye Corners: 33 (left), 133 (right)

- Right Eye Corners: 362 (left), 263 (right)
- Left Iris Center: 468
- Right Iris Center: 473
- These points are used to determine the iris position relative to the eye region, enabling robust gaze detection without deep learning inference.

- **Iris Position Estimation Equation:**
  Gaze direction is inferred by computing the **normalized iris position** within the eye width:
  relative_position = (iris_x - eye_left_x) / (eye_right_x - eye_left_x)
  If the result lies within the calibrated threshold $0.35 < \text{relative\_position} < 0.65$, the gaze is considered **centered**. This threshold range was empirically selected to account for typical pupil movement within natural gaze bounds.
- **Temporal Logic for Gaze Validation:**
  To mitigate frame-level noise and ensure consistent predictions, a temporal gating mechanism is applied:
  - **CENTER_CONFIRMATION_TIME**: A centered gaze must persist for ≥ 0.5 seconds before resetting the non-centered timer.
  - **NON_CENTER_THRESHOLD**: A non-centered gaze persisting for ≥ 3.0 seconds triggers a **cheating flag**.

B) Technologies

- **MediaPipe Framework:**
  Developed by Google, MediaPipe is a cross-platform machine learning pipeline framework that enables real-time face analysis directly from RGB frames. In this system, MediaPipe's Face Mesh solution is used,

which provides 468 high-resolution 3D facial landmarks, including precise iris coordinates when refine_landmarks=True is enabled.

- **BlazeFace – Face Detection Model:**
  The first stage of the pipeline uses BlazeFace, a lightweight and efficient face detector optimized for real-time applications. It performs bounding box regression to locate faces in the input frame. Designed for mobile and edge devices, BlazeFace maintains a high frame rate (>200 FPS on CPU) and robustly handles multiple faces with minimal latency.
- **Face Mesh Landmark Model:**
  Once a face is detected, MediaPipe passes the region to a dedicated Face Mesh Landmark Model, a regression neural network trained to predict 468 3D landmarks. This model reconstructs a dense facial mesh structure, enabling detailed localization of eyes, nose, mouth, jawline, and other key regions necessary for accurate eye-tracking and expression analysis.
- **Iris Landmark Model – Refinement Network:**
  With the refine_landmarks=True setting, an additional Iris Landmark Model is invoked. This specialized CNN predicts 5 refined iris landmarks per eye (landmarks 468–473), providing subpixel-accurate localization of the pupil center. These refined predictions significantly enhance the reliability of gaze direction estimation, particularly in low-resolution or noisy inputs.

4.3.2 Personality Traits Assessment

The Video Module implements a deep learning-based system for the automatic assessment of Big Five personality traits from pre-recorded interview videos. Designed to enhance hiring decisions with scalable and objective personality profiling, this module exclusively utilizes visual signals—particularly facial expressions—to infer trait scores. The system transforms raw video inputs

into continuous personality trait values using spatiotemporal modeling, optimized feature extraction, and regression-based prediction. This approach eliminates manual biases inherent in self-report or interviewer-judged methods, enabling a fairer, data-driven candidate evaluation pipeline.

A) System Functions

The Video Module performs the following core functions to derive Big Five trait predictions from interview videos:

- **Face Detection and Frame Extraction:**
  Videos are processed using the MTCNN (Multi-task Cascaded Convolutional Networks) framework to detect and align faces. A total of 60 evenly spaced frames are extracted per video and resized to 160×160 pixels, with zero-padding applied where needed for consistency.
- **Visual Feature Extraction:**
  The extracted face-aligned frames are passed to a pre-trained X3D model to obtain a spatiotemporal summary of facial behaviors. The model internally captures short-term expressions and motion cues that contribute to personality perception.
- **Trait Prediction via Regression Heads:**
  The final feature vector from the X3D model is passed to five independent multi-layer perceptrons (MLPs). Each MLP maps features to a specific Big Five trait: Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism.
- **Output Generation:**
  The system returns a normalized vector of trait scores ranging between 0 and 1. These values form the candidate's personality profile, enabling objective comparison across applicants. According to predefined thresholds, each trait score is also mapped to a descriptive label—such

as “Friendly” or “Reserved” for Extraversion—providing interpretable insights that align with HR evaluation practices.

B) Techniques and Algorithms

This module leverages recent advances in video-based personality inference by combining spatiotemporal deep learning, lightweight temporal modeling, and supervised regression.

- **X3D Pre-trained Model:**
  The X3D architecture, optimized for video classification, uses 3D convolutions to capture both spatial facial patterns and short-term temporal changes. In our pipeline, 60 face-aligned frames are passed through the X3D model, producing a 400-dimensional vector per video that encodes expressive and temporal features relevant to personality trait prediction.
- **MTCNN Face Detection:**
  Facial regions are reliably detected using the Multi-task Cascaded Convolutional Networks (MTCNN) algorithm. This ensures clean and aligned face crops even in videos with head movement or lighting variation, improving downstream feature quality and inference accuracy.
- **Temporal Modeling via Feature Flattening:**
  Instead of using sequential models like RNNs or Transformers, temporal modeling is handled internally by X3D's 3D convolutions and global average pooling. The final output of the X3D model is a fixed-size 400-dimensional vector per video. This strategy is efficient and reduces the risk of overfitting, especially for smaller datasets.
- **Regression Head Architecture:**
  Five independent regression heads are implemented as dedicated multi-layer perceptrons (MLPs), each mapping the 400-dimensional

video features to a scalar trait score (in the range [0, 1]). Each MLP has the following architecture:
$400 \rightarrow 128 \rightarrow 64 \rightarrow 1$, with GELU activations and a dropout rate of 0.2 after each layer. The five heads predict the Big Five traits: **Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism.**

- **Loss Function and Optimization:**
  The system is trained using the Mean Squared Error (MSE) loss function for continuous trait regression. Optimization is done using the AdamW optimizer with a learning rate of **0.001**, weight decay of **0.0001**, and a cosine learning rate scheduler. This configuration ensures stable convergence and generalization.
- **Training Strategy:**
  The model is trained progressively on the **ChaLearn First Impressions V2 dataset** by dividing it into chunks of 2,000 videos. Each chunk is used for training sequentially, using the checkpoint from the previous chunk to resume training. Training is conducted for **50 epochs per chunk** with a batch size of **16**, and GPU acceleration is employed to handle the computational demands.
- **Evaluation Metrics:**
  Model performance is evaluated using:
- **Implementation and Tools:**
  The system is implemented in PyTorch, utilizing GPU acceleration and frame-level batching to ensure efficient memory usage. Frame extraction is conducted using the `av` library, which provides fast and direct access to video frames. Face alignment is managed by MTCNN to ensure consistent facial region detection across all samples.

C) Technologies

- **X3D-S Pre-trained Video Model:**
  X3D-S, a lightweight spatiotemporal convolutional neural network

developed by Facebook AI, is optimized for efficient video understanding tasks. Pre-trained on the Kinetics-400 dataset (400 human action classes), it extracts meaningful temporal and spatial patterns from face-aligned video frames. Its compact architecture enables fast inference and low memory usage, making it well-suited for scalable personality trait analysis.

- **MTCNN for Face Detection and Alignment:**
  MTCNN (Multi-task Cascaded Convolutional Networks) is utilized to detect and align facial regions across all frames. It provides robustness to variations in pose, lighting, and background noise. This ensures that each frame fed into the X3D model is high quality, centered on the face, and standardized for feature consistency.
- **Temporal Pooling Mechanism:**
  Instead of recurrent networks or Transformers, a lightweight temporal modeling strategy is applied. The X3D model handles spatiotemporal encoding internally, and global average pooling is applied to the feature sequence, summarizing each video into a fixed-length descriptor. This technique improves scalability and avoids overfitting on small datasets.
- **PyTorch Framework:**
  The model is implemented entirely in PyTorch, leveraging its dynamic computation graph capabilities for rapid prototyping and training. PyTorch also ensures compatibility with pre-trained models (e.g., X3D) and provides efficient GPU-based computation.
- **Checkpointing and Frame-Level Batching:**
  Training was conducted using periodic checkpointing to retain the best-performing model states. Frame-level batching was implemented to ensure balanced GPU memory usage and consistent gradient updates, enabling stable training over large video datasets.
- **GPU Acceleration:**
  All training was performed on NVIDIA GPU hardware, significantly

reducing training time. The ChaLearn First Impressions V2 dataset (10,000 annotated videos) was processed efficiently in batches of 2,000 using staged training. Full training was completed in several hours thanks to GPU acceleration.

- **MoviePy for Frame Extraction:**
  The MoviePy library was used to extract and preprocess video frames. It enabled conversion of full-length videos into fixed-frame (e.g., 60 frames) clips, standardized to 160×160 pixels, ensuring consistent input for the face detection and feature extraction stages.

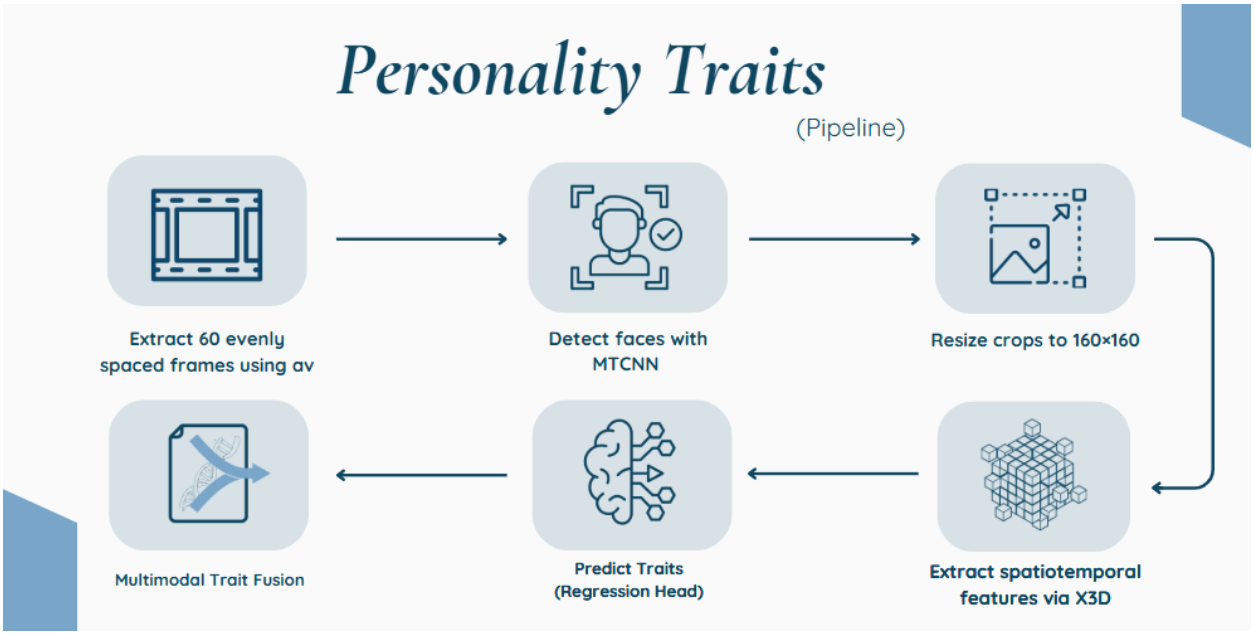


Figure 4.3 Personality Traits Pipeline

D) Dataset:

The **ChaLearn First Impressions V2 Dataset** is a large-scale, publicly available dataset designed for personality analysis based on video data. It is widely used in research and competitions involving social signal processing and psychological profiling. Key features:

- **Size and Scope:** The dataset contains **10,000 short video clips**, each approximately 15 seconds long. All videos feature individuals speaking directly to the camera in a job interview-like setting.
- **Labeling:** Each video is **annotated with continuous scores** for the **Big Five personality traits—Extraversion, Agreeableness, Conscientiousness, Neuroticism, and Openness**—on a scale from 0 to 1. These annotations are based on **crowdsourced human judgments collected via Amazon Mechanical Turk (MTurk)** and were aggregated to reduce individual bias.
- **Demographics and Diversity:** The dataset includes speakers of diverse ethnicities, genders, and ages. This diversity ensures generalizability of the trained models across real-world hiring scenarios.
- **Labeling Process:** Each video was rated by multiple annotators to ensure label consistency. The scores are normalized and averaged across raters to provide continuous ground-truth values for each personality dimension.
- **Focus on Visual Cues:** All annotations are based **solely on visual and behavioral cues**, such as facial expressions, gaze, posture, and gestures, rather than audio or textual information. This makes it ideal for training models based on visual personality inference.

Dataset Usage in Our Model:

During training, we focused on the **visual modality**, specifically facial expressions, to predict the Big Five traits. From each video, **60 evenly spaced face-aligned frames** were extracted using **MTCNN**, resized to 160×160 pixels, and passed through the **pretrained X3D model** for feature extraction.

Due to computational constraints, we adopted an **incremental training strategy**:

- The dataset was split into **five chunks of 2,000 videos**.
- The model was trained on each chunk sequentially, saving checkpoints at each stage and resuming from the previous model state for the next chunk.
- This approach allowed for efficient memory management and improved model generalization.

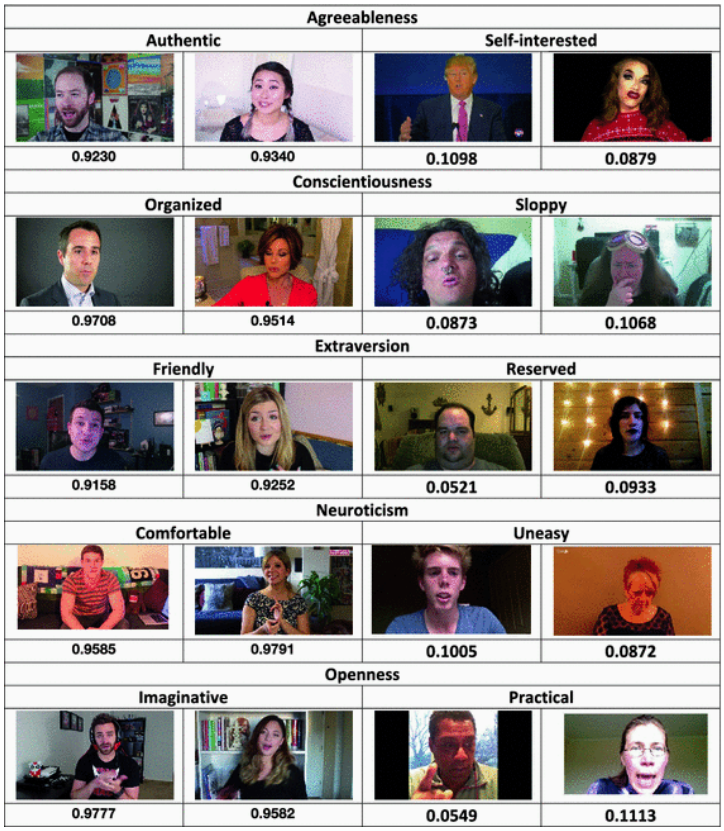


Figure 4.4: Visual Personality Trait Assessment Output

E) Experimental Results:

The performance of the Video Module was evaluated using two key metrics: **Mean Squared Error (MSE)** as the primary loss function and **Mean Accuracy (MA)** as the primary evaluation metric. MA was used to align with the ChaLearn First Impressions Challenge and other related benchmarks. The model was trained and evaluated across multiple training stages, with results summarized in Table 4.3.2.5

Table 4.6: Personality Traits Video Model Results

Trait	Extraversion	Agreeableness	Conscientiousness	Neuroticism	Openness	Mean Acc
Accuracy	**91.3**	89.7	90.8	89.9	91.7	**90.52**

- **Mean Squared Error (MSE):**
  to assess prediction precision on continuous trait values.

$$\text{MSE} = \sum_{j=1}^{5} \frac{1}{N} \sum_{i=1}^{N} \left(y_i^j - \hat{y}_i^j \right)^2 \tag{4.6}$$

N: The number of samples in the current batch.

i: Index over samples in the batch (from 1 to N).

j: Index over the five personality traits

y: The **ground truth** (target) value of trait j for sample i.

$\widehat{y}_i$: The **predicted** value of trait j for sample i.

- **Mean Accuracy (MA):**
  Mean Accuracy reflects how closely the predicted trait scores approximate the ground truth. It is computed by subtracting the absolute error from 1 for each trait, then averaging:

$$\text{MeanAccuracy} = \frac{1}{N \times 5} \sum_{i=1}^{N} \sum_{j=1}^{5} \left(1 - \left|\hat{y}_i^j - y_i^j\right|\right) \tag{4.7}$$

  N: Number of samples in the batch.

  j∈{1,2,3,4,5}:Index for the five personality traits.

  y: The **ground truth** (target) value of trait j for sample i.

  $\widehat{y}_i$: The **predicted** value of trait j for sample i.

  |$\widehat{y}_i$-y|: Absolute error per trait per sample

  1-|$\widehat{y}_i$-y|:Accuracy for trait j of sample i.

  This formulation provides a bounded, intuitive score in the range [0,1], where 1.0 represents perfect agreement. The MA metric allows direct comparison with other works evaluated on the same dataset and benchmark.

Trait Score Fusion Strategy

To generate a **unified personality profile**, predictions from both the **Video** and **Text modules** are integrated using a **weighted average**. Since the Video

module consistently outperforms the Text module (e.g., 90.52% Mean Accuracy), the fusion strategy gives it greater weight:

$$\text{Combined Trait} = 0.7 \times \text{Video Prediction} + 0.3 \times \text{Text Prediction} \tag{4.8}$$

This **weighting strategy** ensures that while both modalities contribute to the final assessment, the more reliable visual signals are emphasized. The final personality profile is obtained by averaging across all samples and converting the results into a five-dimensional trait vector: **Agreeableness, Conscientiousness, Extraversion, Neuroticism, and Openness.**

4.3.3 Emotion Recognition and Engagement Analysis Module

The Emotion Recognition Module leverages deep learning to automatically analyze facial expressions from pre-recorded interview videos. Its primary goal is to assess the emotional state and engagement of candidates during asynchronous or virtual interviews. Built on the DeepFace framework and enhanced with MediaPipe for efficient detection, this module estimates frame-level emotion distributions, detects dominant emotional states, and provides interpretable feedback on candidate affect and engagement levels.

A) System Functions:

The module performs the following key operations:

- **Frame Sampling & Preprocessing:** Using OpenCV (cv2), each video is opened and decoded frame-by-frame. To ensure efficient temporal coverage, five evenly spaced frames per second are sampled. These frames are then resized and standardized to match DeepFace input

specifications, ensuring consistent processing throughout the emotion recognition pipeline.

- **Face Detection & Emotion Classification:** Each sampled frame is passed through the MediaPipe face detector. If a face is detected, it is cropped, aligned, and classified by DeepFace into one of the seven canonical emotions: *Angry, Disgust, Fear, Happy, Sad, Surprise, or Neutral*.

- **Emotion Aggregation:** Emotion predictions are aggregated across all valid frames. The mean emotion probability is computed, and the emotion with the highest score is designated as the dominant emotion for the video.

- **Engagement Assessment:**
  Emotions are grouped by **valence polarity**:
  - *Positive*: Happy, Surprise, Neutral
  - *Negative*: Angry, Sad, Fear, Disgust
    A higher average for positive emotions indicates **confidence or composure**, while a negative trend may indicate **nervousness or discomfort**.

B) Techniques and Algorithms:

CV2-Based Frame Sampling: Video frames are sampled using **OpenCV (cv2)**, a widely adopted computer vision library. Each interview video is loaded and decoded using cv2.VideoCapture, and **five evenly spaced frames per second** are extracted. This ensures efficient temporal coverage while keeping computational overhead low.

Frames are then preprocessed and resized before being passed to the face detection and emotion classification pipeline.

MediaPipe for Face Detection: Chosen for its **speed, accuracy**, and **cross-platform compatibility**, MediaPipe consistently detects faces in high-FPS scenarios and works reliably on edge devices or in CPU-only environments.

DeepFace Emotion Classifier: A pre-trained convolutional neural network trained on **FER-2013** (and optionally fine-tuned on AffectNet) is used to classify emotions from aligned face crops.

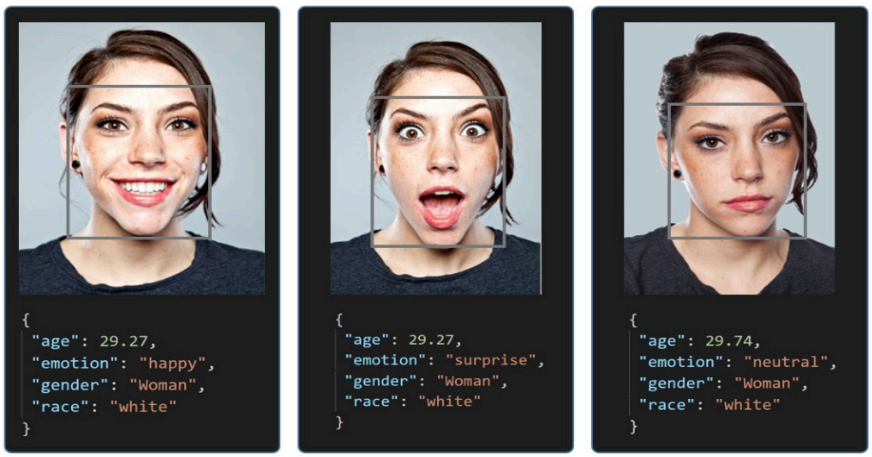


Figure 4.5: Emotion Detection Output

Valence-Based Emotion Mapping: Predictions are grouped into broader categories (*positive, negative*) to improve interpretability for non-technical users (e.g., recruiters or educators).

Fault Tolerance: Frames without valid face detections are automatically skipped. This ensures robustness in the presence of motion blur, occlusion, or lighting inconsistencies.

C) Technologies:

DeepFace Emotion Engine: A plug-and-play, pre-trained deep learning model requiring no fine-tuning. This makes it ideal for large-scale deployment across diverse user environments.

MediaPipe Face Detection: Offers high-performance face tracking with low computational demand, outperforming MTCNN in real-time scenarios.

Valence-Based Engagement Inference: Emotion labels are abstracted into high-level engagement descriptors like *"confident," "neutral,"* or *"anxious"*, providing recruiters with intuitive insights.

Inference-Only Architecture: The module requires no retraining. All predictions are generated using pre-trained models, optimizing runtime efficiency and deployment speed.

D) Dataset: CAER (Context-Aware Emotion Recognition)

As the model was pre-trained, we validated it using 350 videos from the CAER dataset—a benchmark composed of emotion-labeled clips from real-world TV shows. Although designed for entertainment, CAER provides diverse facial expressions and realistic variability that resemble conditions found in remote interviews. Each video includes annotations for seven emotions (*angry, disgust, fear, happy, sad, surprise, neutral*), allowing us to assess emotion prediction and engagement classification accuracy. Frames were sampled

using `cv2` at 5 FPS, enabling consistent temporal coverage and robust validation of the model in real-world-like settings.

E) Experimental Results:

The module was validated using a sample of 350 labeled videos from the CAER dataset. Evaluation focused on three performance indicators:

Table 4.7: DeepFace Results

Frame-Level Face Detection Rate	Dominant Emotion Agreement
>90% (MediaPipe)	85%+ with ground truth

F) Trait Mapping:

Based on the dominant emotion and aggregated valence score, an **engagement label** is generated:

- Confident, Neutral, or Nervous
  This aligns with the observed emotion distribution and provides human-readable insight into the candidate's demeanor.

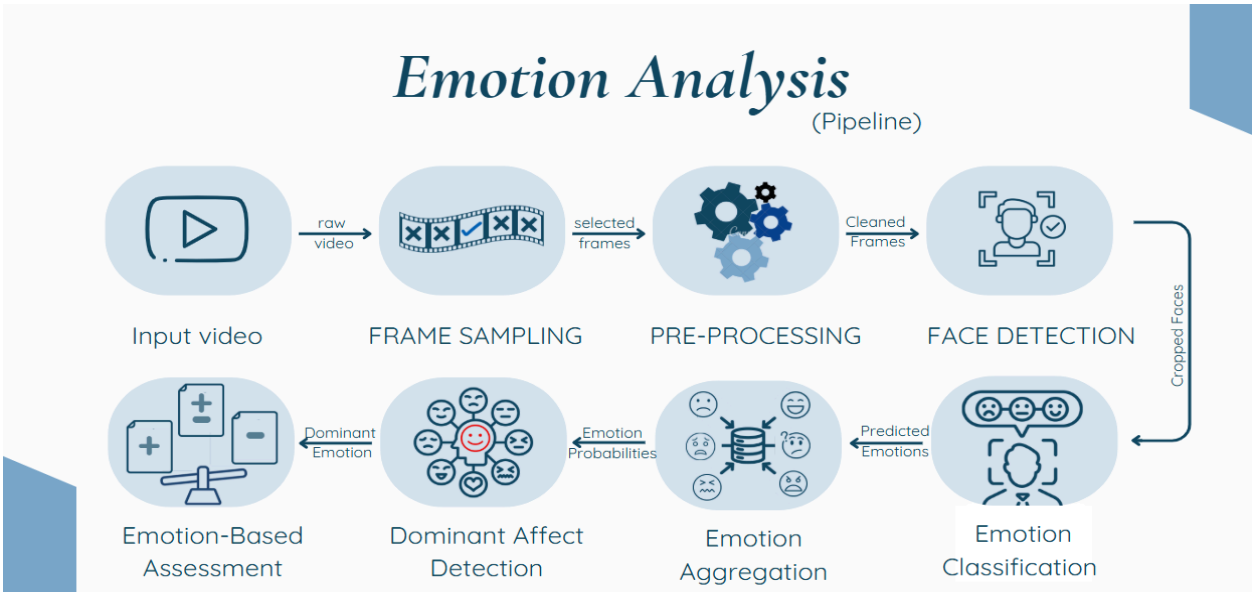


Figure 4.6: Emotion Analysis Pipeline

Chapter 5

User Manual

PRVIA is a web-based job platform developed to streamline the recruitment process for both **candidate users** and **HR administrators**. The system allows candidates to explore available job opportunities, submit applications, and complete video interviews, while enabling HR admins to post jobs, manage applicants, and review AI-analyzed interview responses. The application was developed using **FastAPI** for the backend due to its speed, simplicity, and support for asynchronous processing—ideal for handling tasks like video upload and AI-based scoring efficiently. The frontend is built with **React.js** to provide a responsive and user-friendly interface.

The user interface is designed using **blue shades and gradient backgrounds**, chosen intentionally to evoke a sense of professionalism, trust, and

calm—qualities that are essential in a hiring environment and appealing to both job seekers and recruiters.

This chapter provides a complete guide to installing, configuring, and operating the PRVIA system. It includes all necessary third-party tools and dependencies, instructions for running both backend and frontend components, and visual walkthroughs (with screenshots) of the system's main features from the perspectives of both users and administrators.

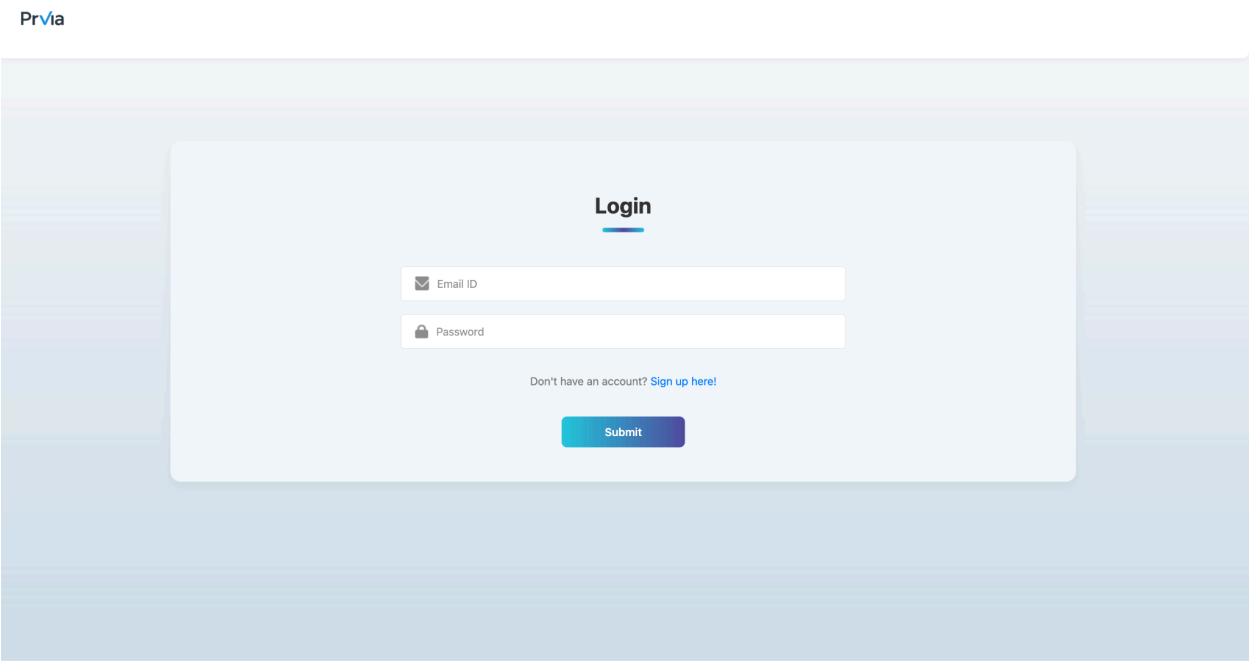
5.1 Operation Guide

Below is a step-by-step illustration of how the system works:

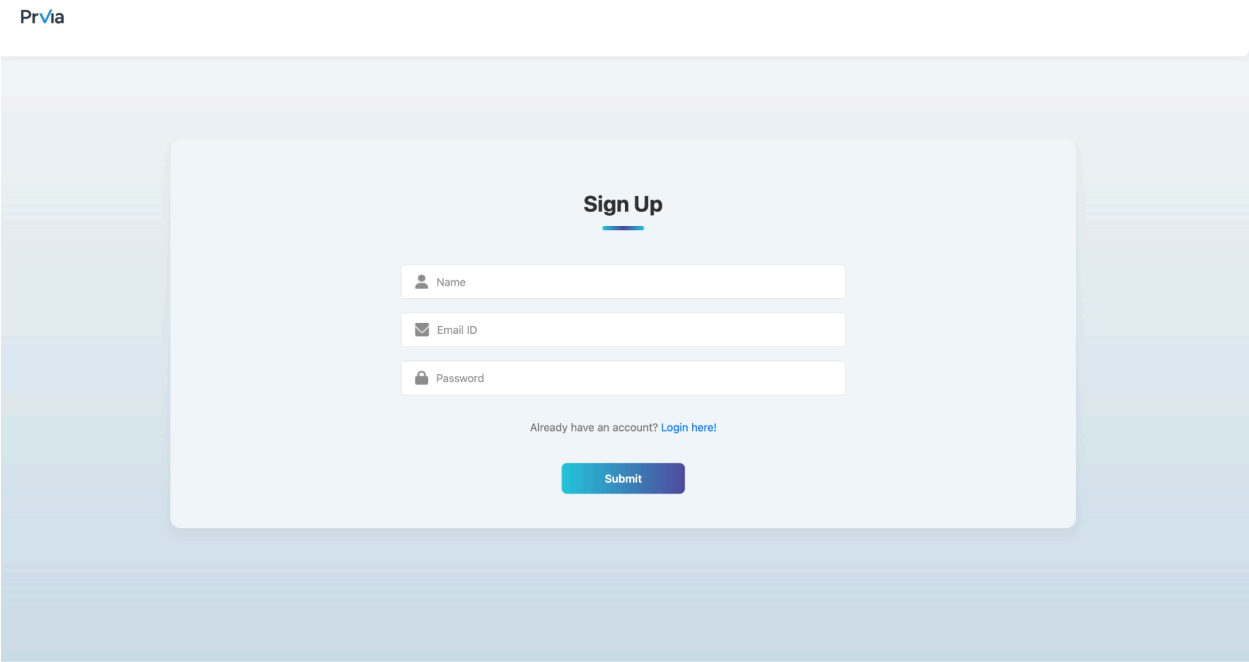
5.1.1 Hr side of the Application

HR admins require authentication and accounts in order to use PRVIA's website.Any Hr needs to post and create jobs of their companies, shall create an account on the website, then log in using the created accounts any time to view the jobs they have created,who have applied on and see the interview analysis of each user

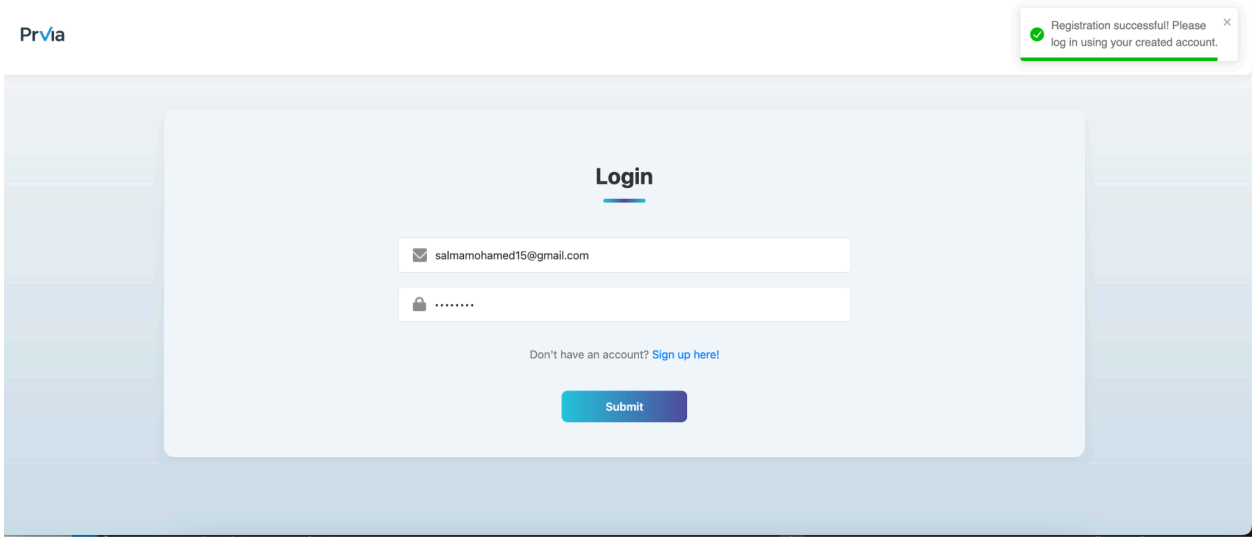
- **Login and Registration:** In order to login or register as an Hr, You shall navigate to the domain/hr, by default /hr navigates directly to the login page if the hr is not authenticated
    - If an hr have already created an account before, he shall use the email and the password to login:



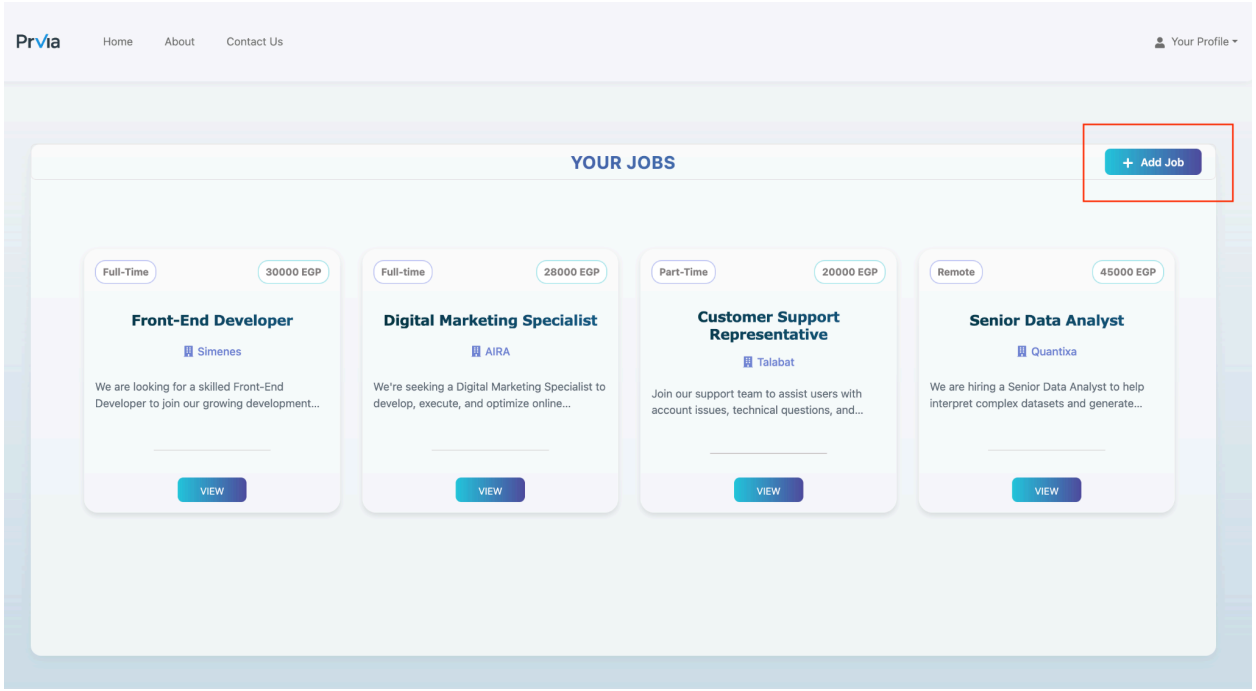
- If he has no account yet,and the first time to use PRVIA, he shall click on ***Sign up here***, and he will be navigated to the registration page:



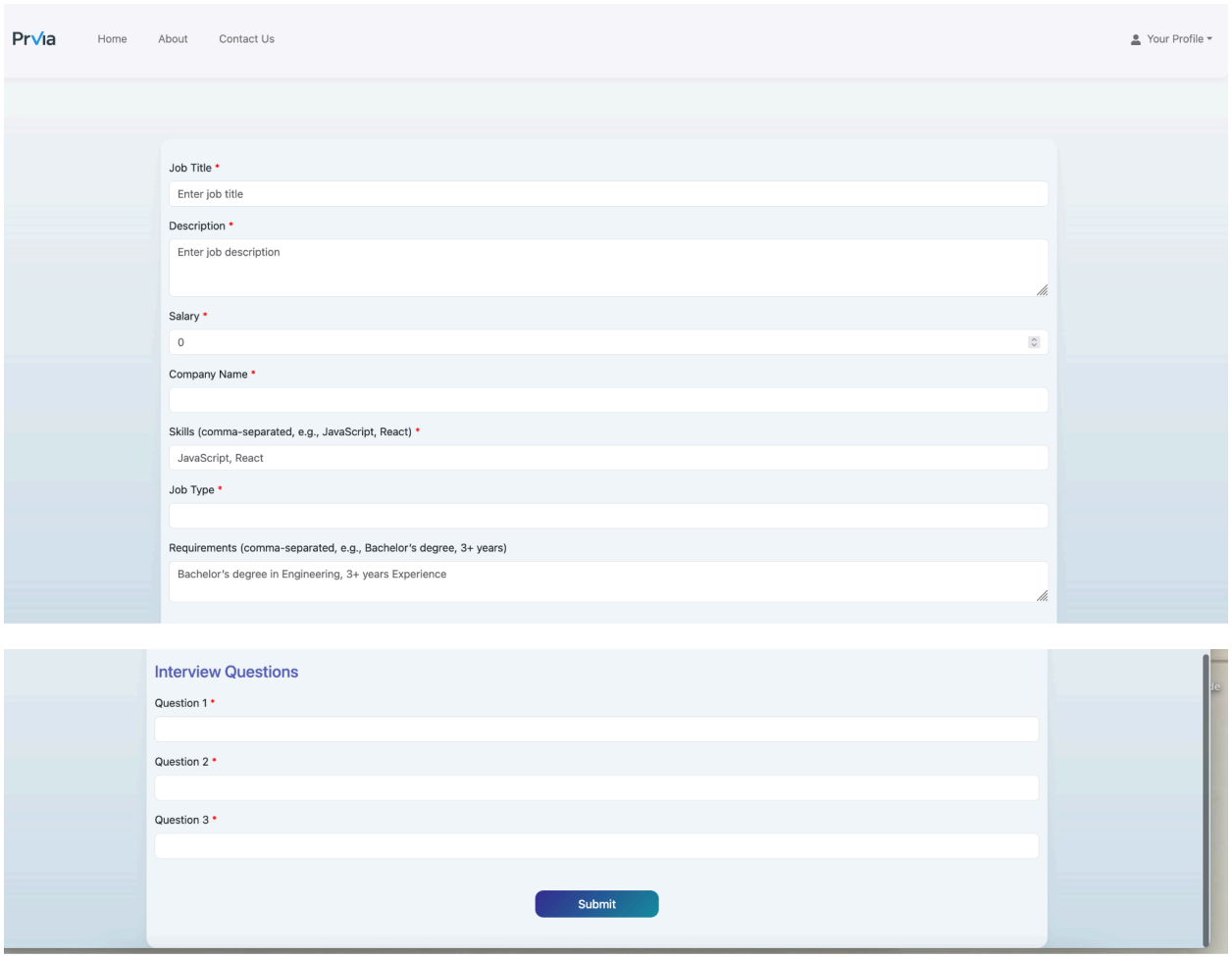
- Then if the registration is successful, you will be redirected to login with your created account:



- Once logged in, he can be able to see the list of jobs he has created only:

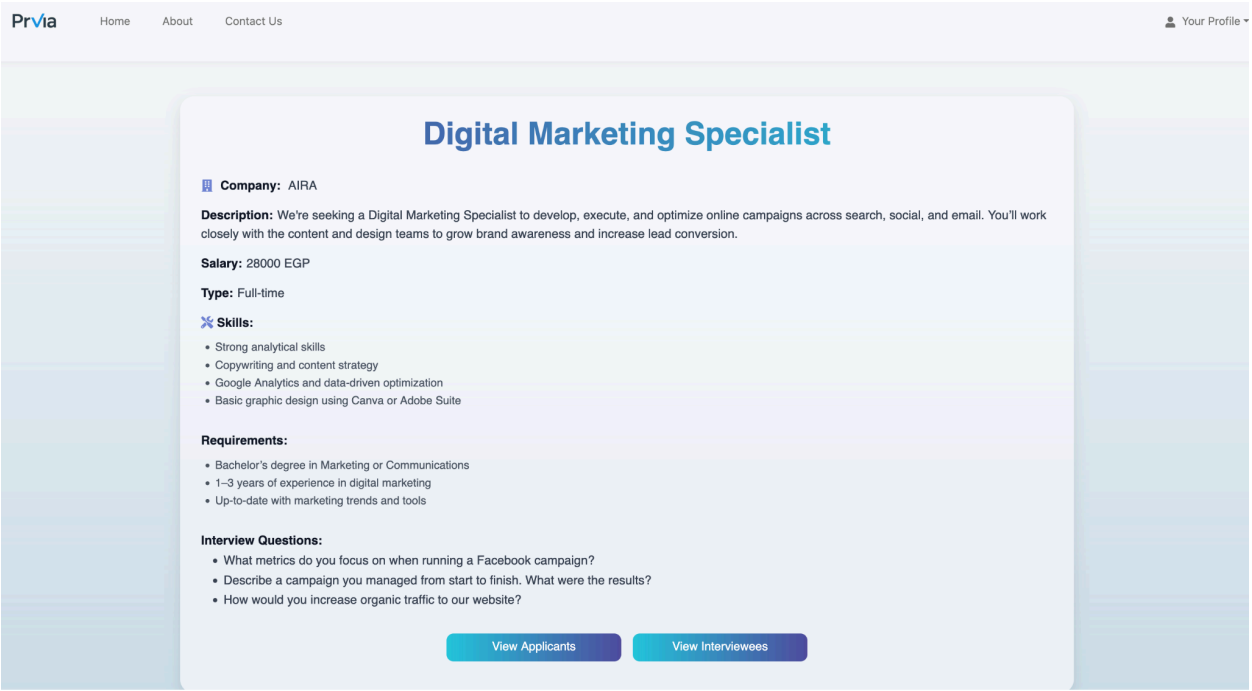


- He can add any new job to his home page through the ***Add Job*** button,If he clicks it , a form will be opened to add the job details along with the questions associated for it for any candidate to make an interview:

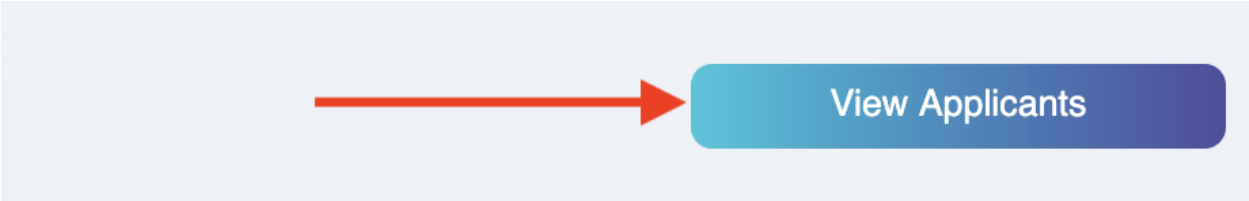


All the fields are required and must be filled to create a job

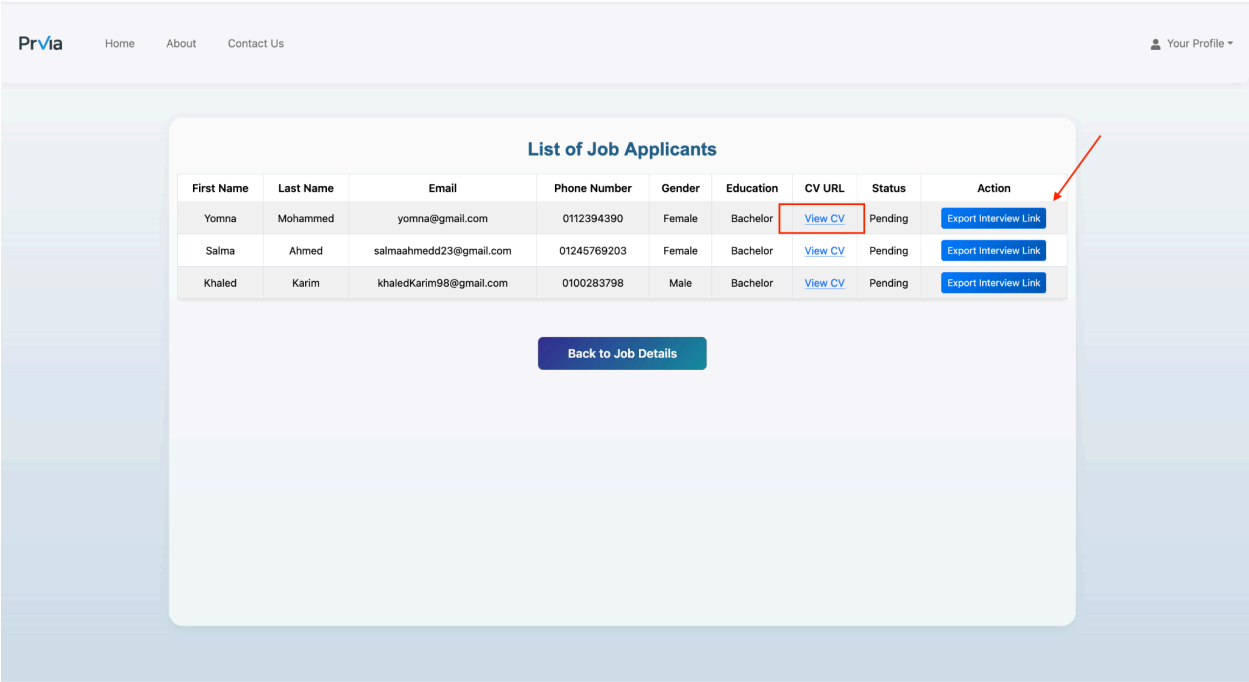
- If he clicked the ***View*** button on any job, he will be able to see all the job details along with 2 buttons:



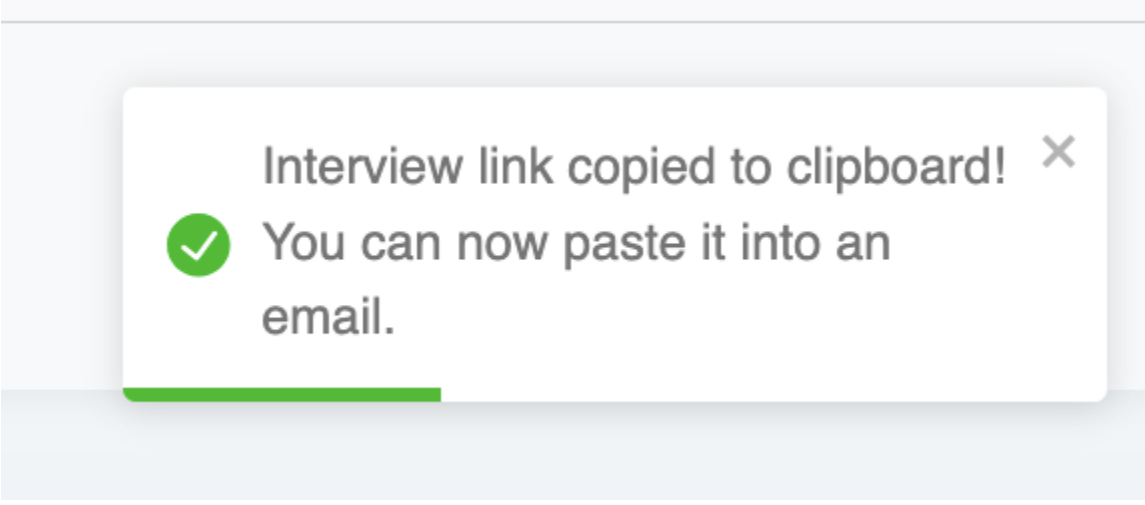
- By clicking ***View Applicants button:***



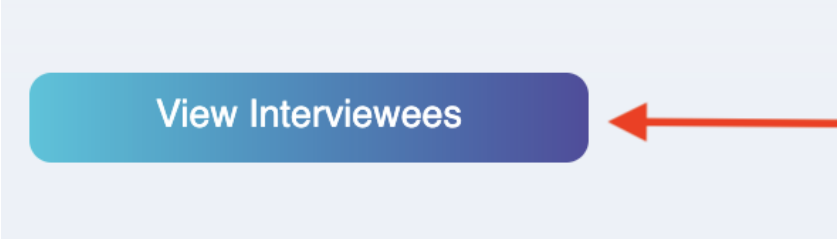
- He will be redirected to the page to see all candidates who have applied on this job post, and view their CVs, and either proceed with a video interview or not:

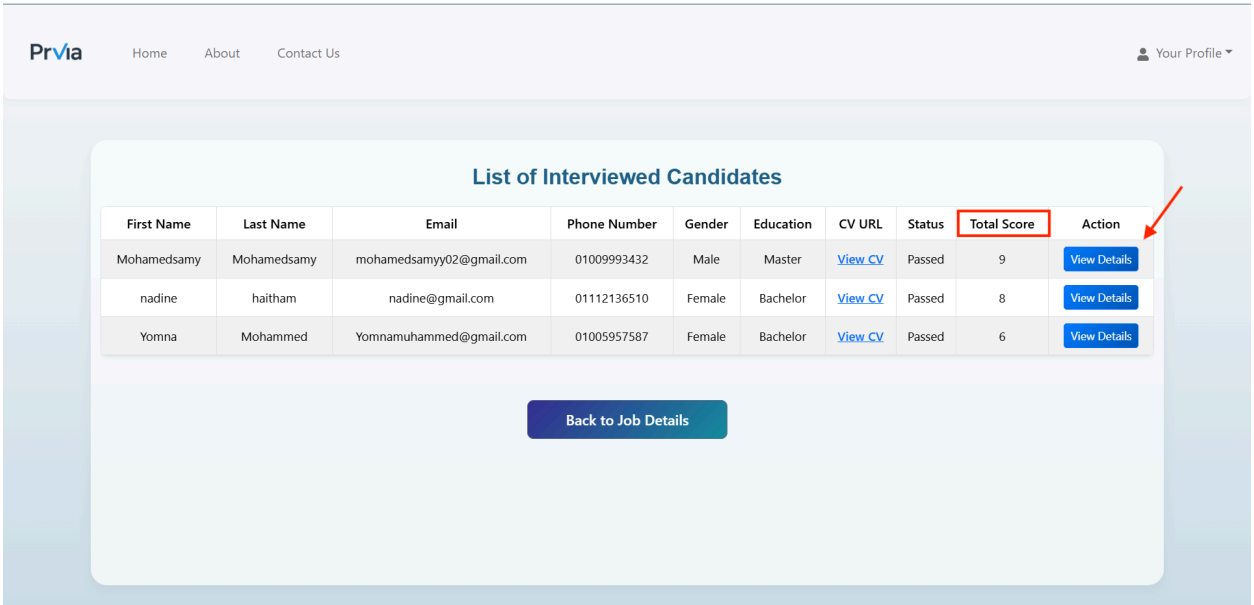


- This will be the list of applicants applied for this job, he can view their CVs from the ***View CV link,*** and if accepted he shall use the ***export interview link*** button and send it in an email the contact of this user telling him the instructions of the submission of the videos and this user status will automatically change from **Pending** to **Interview-Process**:

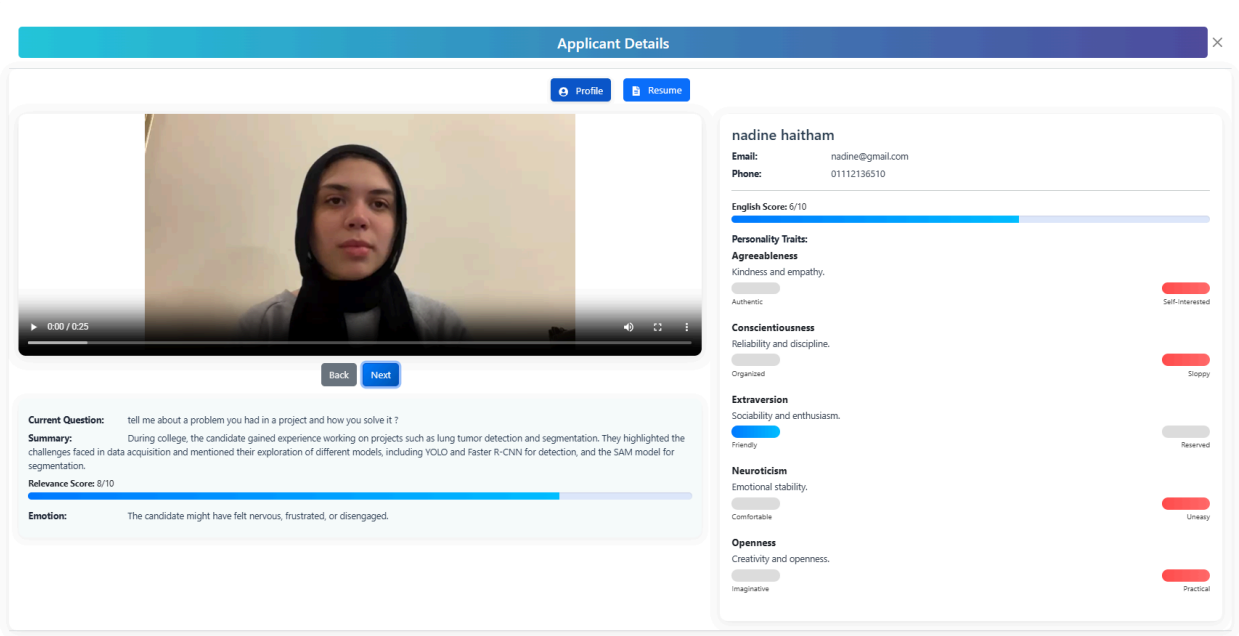


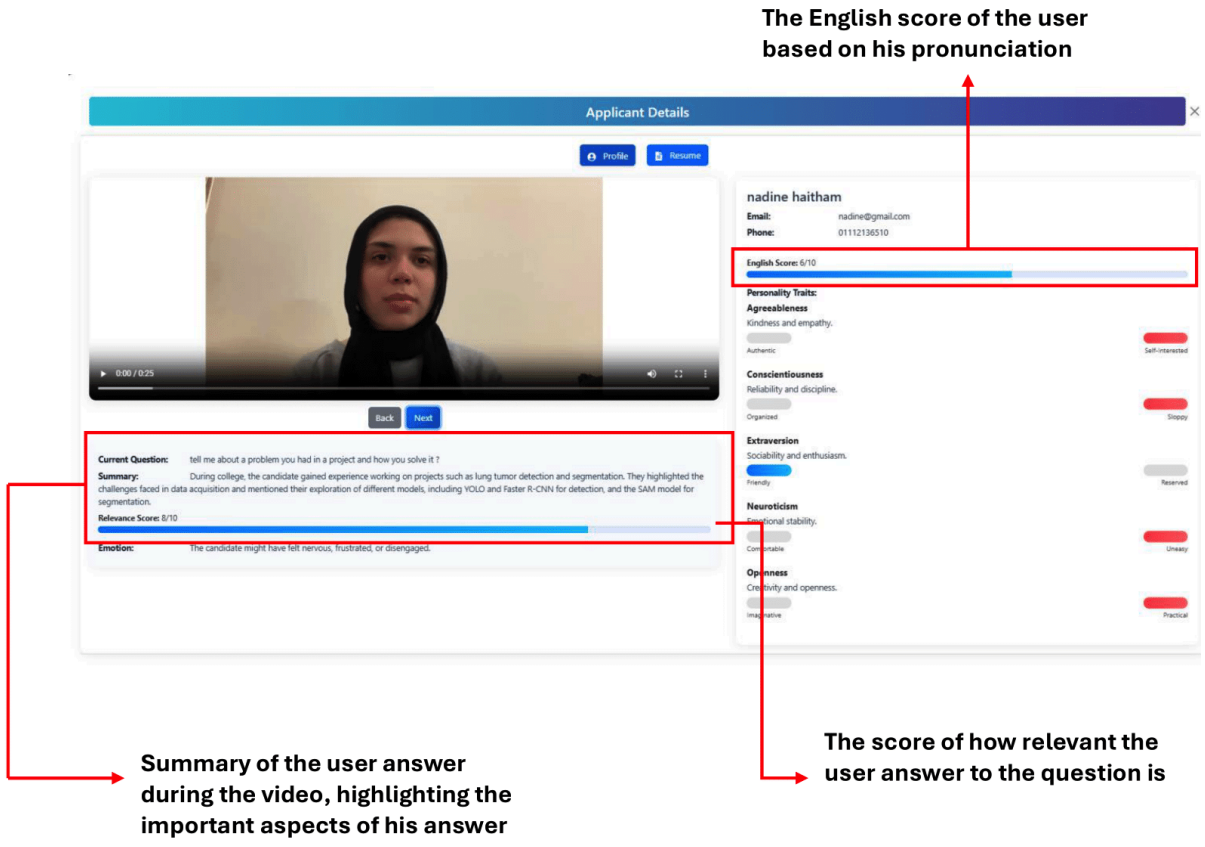
- ***By*** clicking ***View Interviewees button:***

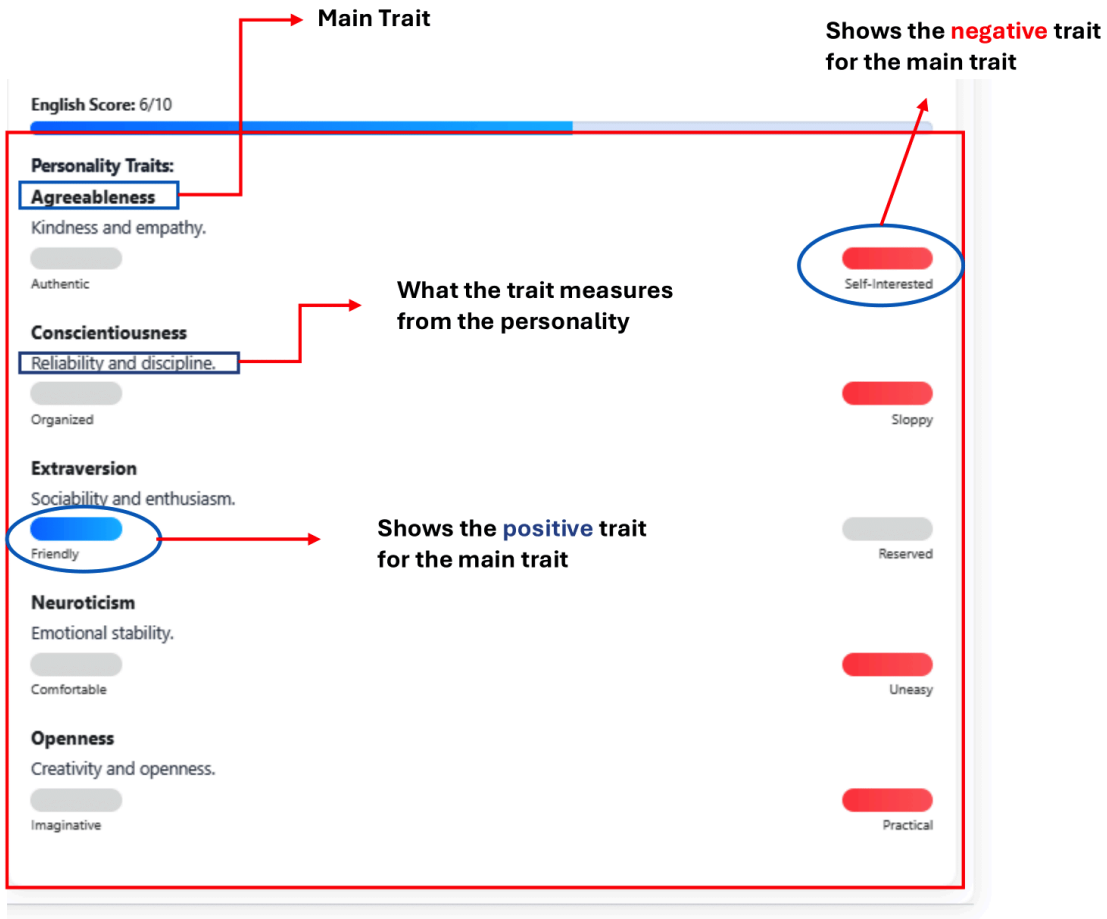




- The ***total score*** field in the table corresponds to the total score generated by PRVIA's models for this user after submitting the videos
- The ***View Details*** button allows the HR to see the detailed analysis of all models, and how the total score is generated, By clicking it opens a User Modal with details for each submitted video like this:


Applicant Details
Profile
Resume
0:00 / 0:25
Back
Next
Current Question: tell me about a problem you had in a project and how you solve it ?
Summary: During college, the candidate gained experience working on projects such as lung tumor detection and segmentation. They highlighted the challenges faced in data acquisition and mentioned their exploration of different models, including YOLO and Faster R-CNN for detection, and the SAM model for segmentation.
Relevance Score: 8/10
Emotion: The candidate might have felt nervous, frustrated, or disengaged.
nadine haitham
Email: nadine@gmail.com
Phone: 01112136510
English Score: 6/10
Personality Traits:
Agreeableness
Kindness and empathy.
Authentic
Self-interested
Conscientiousness
Reliability and discipline.
Organized
Sloppy
Extraversion
Sociability and enthusiasm.
Friendly
Reserved
Neuroticism
Emotional stability.
Comfortable
Uneasy
Openness
Creativity and openness.
Imaginative
Practical

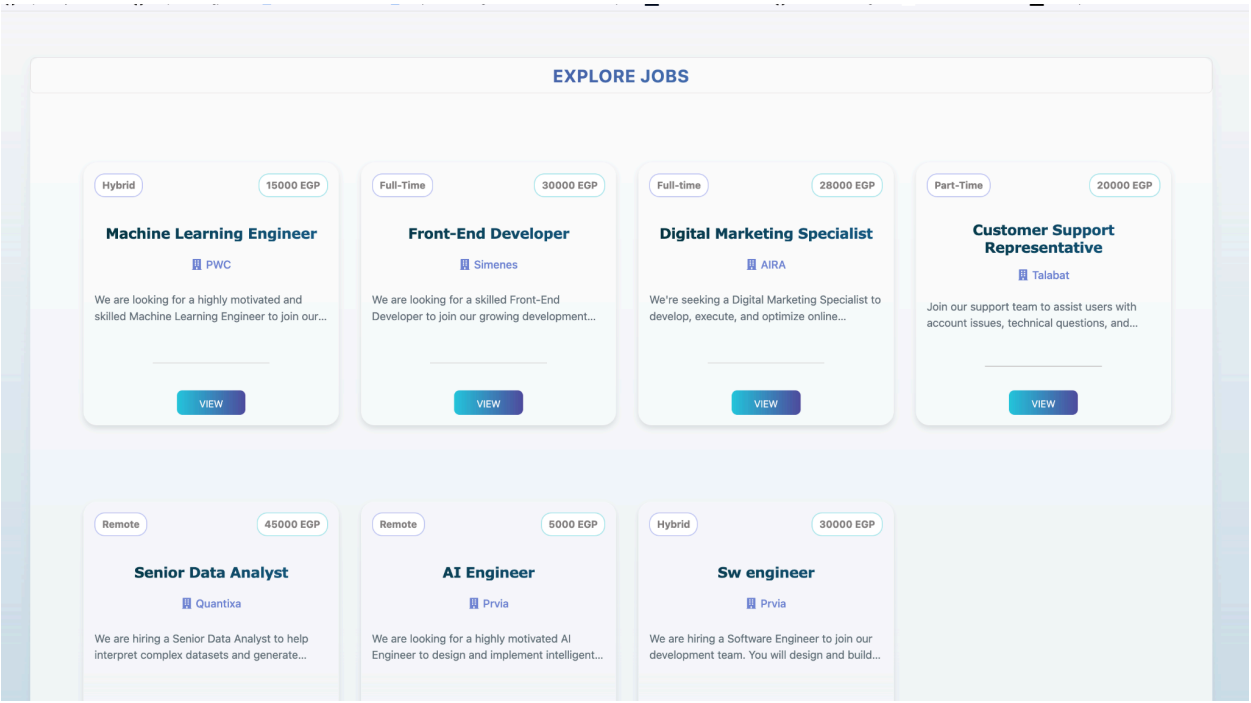

The English score of the user based on his pronunciation
Applicant Details
Profile
Resume
0:00 / 0:25
Back
Next
Current Question: tell me about a problem you had in a project and how you solve it ?
Summary: During college, the candidate gained experience working on projects such as lung tumor detection and segmentation. They highlighted the challenges faced in data acquisition and mentioned their exploration of different models, including YOLO and Faster R-CNN for detection, and the SAM model for segmentation.
Relevance Score: 8/10
Emotion: The candidate might have felt nervous, frustrated, or disengaged.
nadine haitham
Email: nadine@gmail.com
Phone: 01112136510
English Score: 6/10
Personality Traits:
Agreeableness
Kindness and empathy.
Conscientiousness
Reliability and discipline.
Extraversion
Sociability and enthusiasm.
Neuroticism
Emotional stability.
Openness
Creativity and openness.
Summary of the user answer during the video, highlighting the important aspects of his answer
The score of how relevant the user answer to the question is


Main Trait
Shows the negative trait for the main trait
English Score: 6/10
Personality Traits:
Agreeableness
Kindness and empathy.
Authentic
Self-Interested
What the trait measures from the personality
Conscientiousness
Reliability and discipline.
Organized
Sloppy
Extraversion
Sociability and enthusiasm.
Friendly
Shows the positive trait for the main trait
Reserved
Neuroticism
Emotional stability.
Comfortable
Uneasy
Openness
Creativity and openness.
Imaginative
Practical

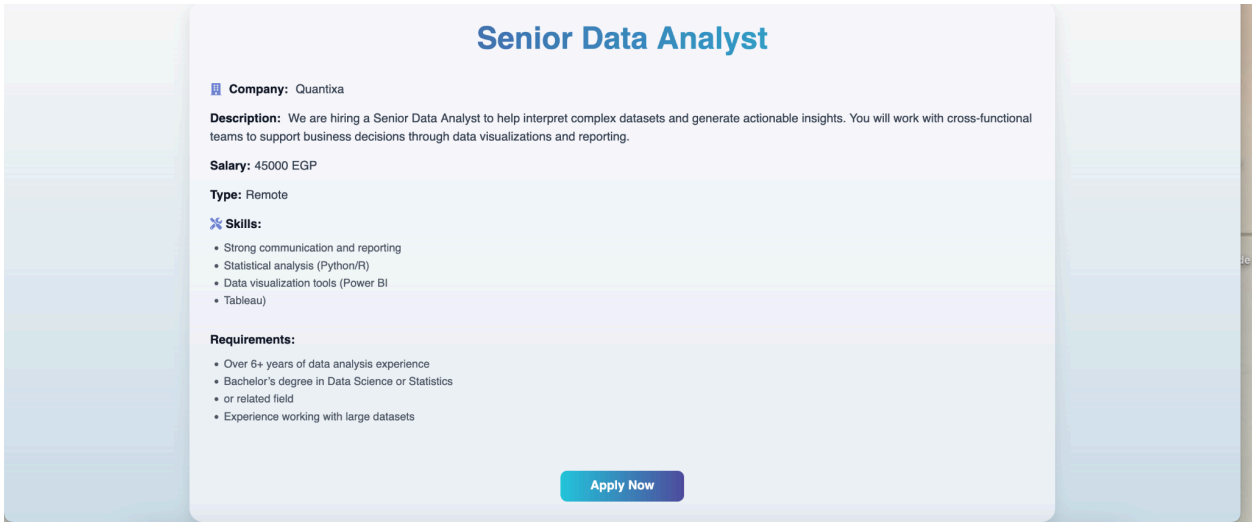
5.1.2 User side of the Application

The user in the system is treated as a guest and can explore the available jobs and apply for any one,no registration or login is required for him , once applying for a job we already collect his personal information for contacting if accepted

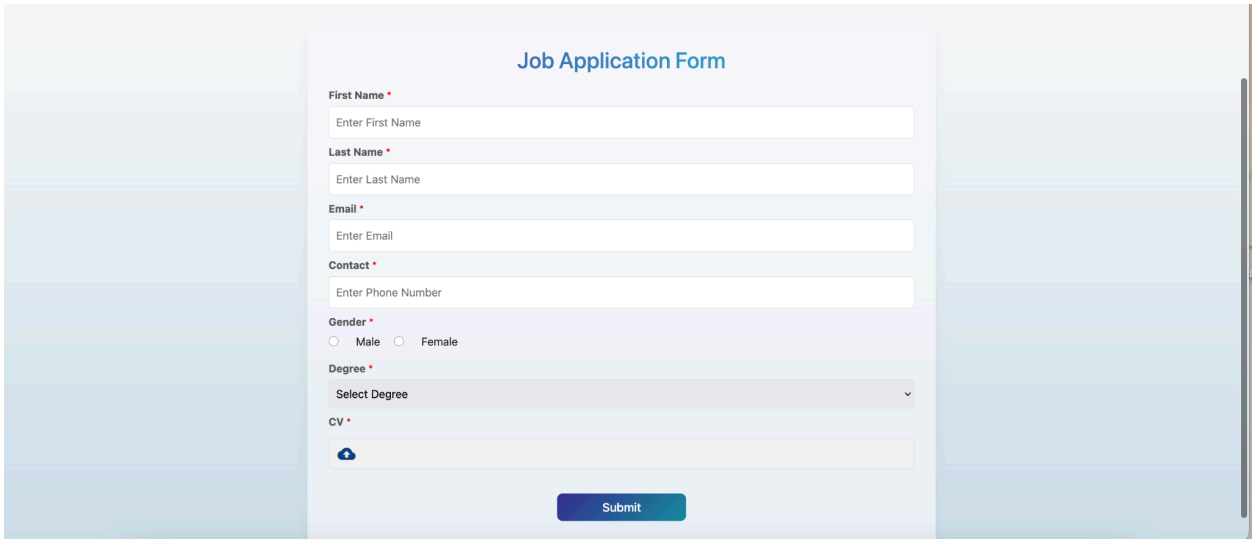
- The User (candidate) will be able to see all the job postings by all HRs on the home page:



- By clicking view on any Job post, he will be redirected to the details of the job similar to the HR page but without seeing the interview questions:

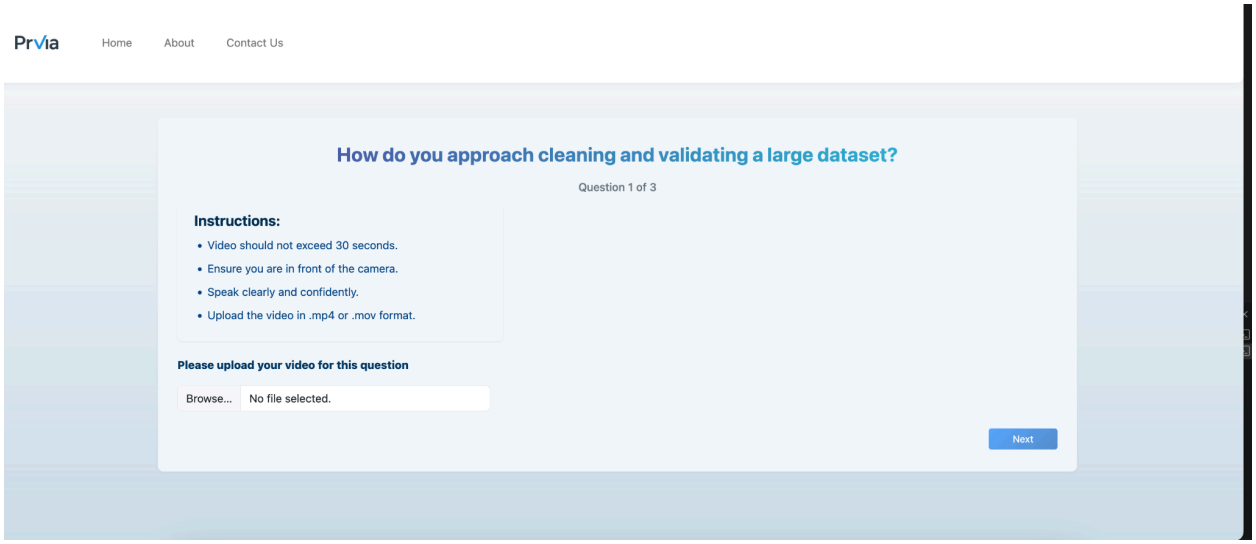


- By clicking ***Apply Now,*** he will be redirected to the Application form to submit his data and wait for an email:

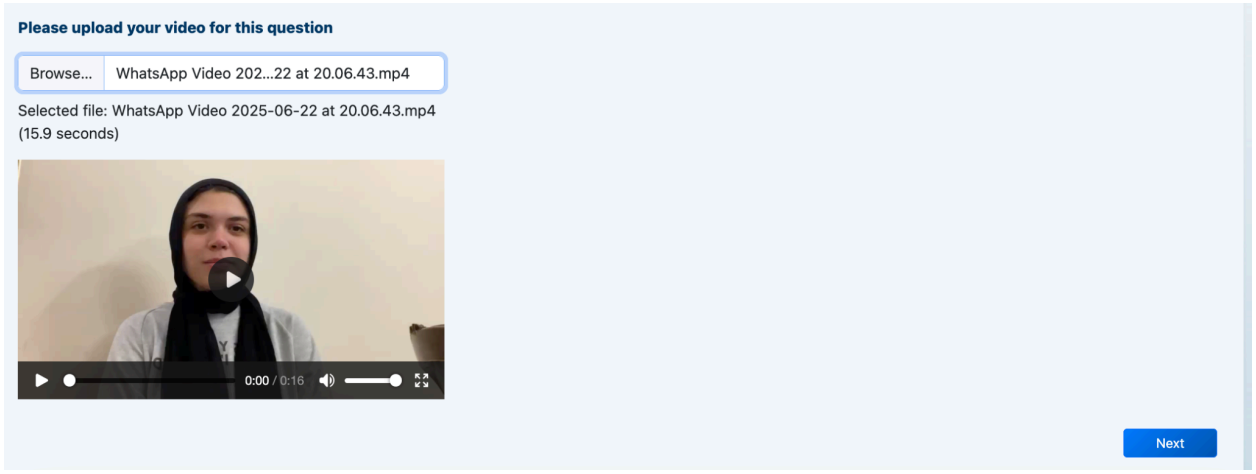


All the fields are required and must be submitted by the candidate to finish the application process.

- If the HR viewed this candidate application and decided to proceed with him to next phase (Video Interview) he will send the interview link to his email and the user will be redirected to the Question pages with instructions:



- The ***Next*** button will not be enabled until a video is submitted:



5.2 Installation Guide:

This section provides a detailed explanation of how to install, configure, and run the project, including the setup of third-party tools, backend services, and the frontend interface. This guide is intended for both technical evaluators and users who wish to test the system locally.

5.2.1 System Requirements:

Before installing the project, ensure that the following dependencies are installed on your machine:

- **Backend Requirements:**
  - Python 3.8+
  - PostgreSQL (v13 or later)
  - pip (Python package manager)
- **Frontend Requirements:**
  - Node.js (v16 or later)
  - npm (Node package manager)

You can download the required tools from the following official websites:

- PostgreSQL: https://www.postgresql.org/download/
- Node.js and npm: https://nodejs.org/
- Python: https://www.python.org/downloads/

5.2.2 Backend Setup

- Navigate to the backend directory and create a `.env` file inside the **Backend** folder. This file will contain your database configuration using:
  - `cd Video-Interview-Analysis/Backend`
  - `cp .env.example .env`
- Add the following variables to your `.env` file and fill in the appropriate values:
  - `user = your_postgres_username`
  - `password = your_postgres_password`
  - `host = your_localhost`
  - `port = 5432`
  - `database=Video-Interview-Analysis`
- Install Python Dependencies:
  - `cd src`
  - `pip install -r requirements.txt`
- Start the Backend Server:
  - `uvicorn main:app`

5.2.3 FrontEnd Setup:

In a new terminal window, follow these steps:

- `cd Video-Interview-Analysis/prvia-frontend-app`
- `npm install`
- `npm start`

Chapter 6

Conclusions and Future Work

6.1 Conclusions

In this project, we developed **Prvia**, an AI-powered solution designed to assist Human Resources departments in optimizing and accelerating the recruitment process through comprehensive video interview analysis. The core objective was to automate key aspects of candidate evaluation from saving time, enhancing decision-making, and reducing the burden of manual video screening.

PRVIA integrates several advanced AI modules targeting key assessment areas: **English language proficiency**, **personality traits**, **emotion detection based on facial expressions**, **text summarization, relevance check** and **cheating detection**. Each candidate-submitted video is automatically analyzed using a combination of **natural language processing**, **computer vision**, and **deep learning** techniques.

The system evaluates English proficiency, identifies key personality traits, and detects emotions through facial analysis. Additionally, each video is paired with a concise and informative summary of the spoken content, enabling HR professionals to quickly grasp the core of the candidate's responses without rewatching the full interview again. A grammar correction module ensures the linguistic quality of transcribed responses, while a cheating detection module flags potential cases of dishonesty for people who might cheat while recording their interview video.

To evaluate the validity and competitiveness of our system, **we compared each module with recent academic work in the field.**

Overall, **Prvia** delivers HR teams a comprehensive report for every candidate, significantly controlling the screening process and improving the quality of recruitment decisions. This project demonstrates the transformative potential of AI in modern hiring workflows and lays the groundwork for future enhancements and real-world deployment.

6.2 Future Work

While **PRVIA** provides a strong foundation for automating the video interview evaluation process, there are several areas where the system can be further enhanced and expanded in future iterations:

1. Model Improvements

Future versions of the system can incorporate **more advanced and fine-tuned models** that are specifically trained on interview datasets. Additionally, exploring larger language models or fine-tuning existing models on domain-specific data could improve both the accuracy and naturalness of the generated summaries.

2. Scalability

Future work should focus on improving the **scalability** of the platform to accommodate a larger number of simultaneous users, making it suitable for enterprise-level deployment and supporting multiple organizations at once without performance degradation.

3. Faster Model Inference

Future improvements could aim to **optimize model latency and response time** by integrating faster, lightweight models or using model distillation techniques to enable quicker analysis, especially in high-traffic environments.

4. Integration with Applicant Tracking Systems (ATS)

A valuable next step is to **integrate the PRVIA system with existing Applicant Tracking Systems (ATS)** to provide a seamless experience for HR professionals. This would allow automatic syncing of candidate reports, video submissions, and summaries directly into the company's recruitment management platform.

5. Development of an HR Virtual Agent

Building a **smart HR agent** that can autonomously interact with candidates during the interview process is another promising direction. This agent could ask pre-defined or dynamic interview questions, follow up with candidates in real-time, and guide them through the video submission process without human supervision.

6. Real-Time Video Analysis

Future versions can aim to support **real-time video analysis** and live interview processing, providing instant feedback to both candidates and HR teams. This would significantly reduce waiting times and enable more interactive recruitment experiences.

7. Chatbot Integration

Developing an **AI-powered chatbot** to assist candidates throughout the application and interview process would enhance user experience. The chatbot could answer questions, provide interview instructions, and offer feedback in real-time, making the system more user-friendly and accessible.

References

[1] Wijerathne, H.M.C.N., Wasana, P.R.E.C., Kugathasan, B. and Weerasiri, H.A.K.D., 2023, July. Smart recruitment tool with AI Technology. In *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-8). IEEE.

[2] Suen, H.Y., Hung, K.E. and Lin, C.L., 2020. Intelligent video interview agent used to predict communication skill and perceived personality traits. *Human-centric Computing and Information Sciences*, *10*(1), p.3.

[3] Kassab, K., Kashevnik, A., Glekler, E. and Mayatin, A., 2023, May. Human sales ability estimation based on interview video analysis. In *2023 33rd Conference of Open Innovations Association (FRUCT)* (pp. 132-138). IEEE.

[4] Kassab, K. and Kashevnik, A., 2024, April. Novel Framework for Job Interview Processing Automation Based on Intelligent Video Processing. In *2024 35th Conference of Open Innovations Association (FRUCT)* (pp. 336-342). IEEE.

[5] Hemamou, L., Felhi, G., Vandenbussche, V., Martin, J.C. and Clavel, C., 2019, July. Hirenet: A hierarchical attention model for the automatic analysis of asynchronous video job interviews. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 33, No. 01, pp. 573-581).

[6] Li, Y., Wan, J., Miao, Q., Escalera, S., Fang, H., Chen, H., Qi, X. and Guo, G., 2020. Cr-net: A deep classification-regression network for multimodal apparent personality analysis. *International Journal of Computer Vision*, *128*, pp.2763-2780.

[7] Lee, B.C. and Kim, B.Y., 2021. Development of an AI-based interview system for remote hiring. *International Journal of Advanced Research in Engineering and Technology*, *12*(3), pp.654-663.

[8] Naim, I., Tanveer, M.I., Gildea, D. and Hoque, M.E., 2016. Automated analysis and prediction of job interview performance. *IEEE Transactions on Affective Computing*, *9*(2), pp.191-204.

[9] Hanani, A., Abusara, Y., Maher, B. and Musleh, I., 2022. English speaking proficiency assessment using speech and electroencephalography signals.

[10] Xenos, A., Foteinopoulou, N.M., Ntinou, I., Patras, I. and Tzimiropoulos, G., 2024. Vllms provide better context for emotion understanding through common sense reasoning. *arXiv preprint arXiv:2404.07078*.

[11] Lee, J., Kim, S., Kim, S., Park, J. and Sohn, K., 2019. Context-aware emotion recognition networks. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 10143-10152).

[12] Chen, L., Zhao, R., Leong, C.W., Lehman, B., Feng, G. and Hoque, M.E., 2017, October. Automated video interview judgment on a large-sized corpus collected online. In 2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII) (pp. 504-509). IEEE.

[13] Anglekar, S., Chaudhari, U., Chitanvis, A. and Shankarmani, R., 2021, June. A deep learning based self-assessment tool for personality traits and interview preparations. In 2021 International Conference on Communication information and Computing Technology (ICCICT) (pp. 1-3). IEEE.

[14] Chakraborty, I., Chiong, K., Dover, H. and Sudhir, K., 2023. AI and AI-Human Based Salesforce Hiring using Conversational Interview Videos. Available at SSRN, 4137872.

[15] Gong, Y., Chen, Z., Chu, I.H., Chang, P. and Glass, J., 2022, May. Transformer-based multi-aspect multi-granularity non-native english speaker pronunciation assessment. In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 7262-7266). IEEE.

[16] Liu, W., Fu, K., Tian, X., Shi, S., Li, W., Ma, Z. and Lee, T., 2023, June. An ASR-free fluency scoring approach with self-supervised learning. In ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1-5). IEEE.

[17] Fu, K., Peng, L., Yang, N. and Zhou, S., 2024. Pronunciation Assessment with Multi-modal Large Language Models. arXiv preprint arXiv:2407.09209.

[18] Witt, S.M. and Young, S.J., 2000. Phone-level pronunciation scoring and assessment for interactive language learning. *Speech communication*, *30*(2-3), pp.95-108.

[19] Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11), 613–620. https://doi.org/10.1145/361219.361220

[20] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*. https://doi.org/10.48550/arXiv.1301.3781

[21] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. https://doi.org/10.48550/arXiv.1810.04805

[22] Jones, K. S. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1), 11–21. https://doi.org/10.1108/eb026526

[23] Vaswani, A., Shazeer, N., Parmar, N., Uszoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 5998–6008. https://papers.nips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[24] Pennebaker, J. W., Francis, M. E., & Booth, R. J. (2001). *Linguistic Inquiry and Word Count: LIWC 2001*. Mahwah, NJ: Lawrence Erlbaum Associates.

[25] Costa, P. T., Jr., & McCrae, R. R. (1992). *Revised NEO Personality Inventory (NEO-PI-R) and NEO Five-Factor Inventory (NEO-FFI) professional manual*. Psychological Assessment Resources.

[26] Mairesse, F., Walker, M. A., Mehl, M. R., & Moore, R. K. (2007). Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research*, 30, 457–500. https://doi.org/10.1613/jair.2349