



Automated Interview Assessment System

¹ Shivaji Vasekar, ² Sujal Kadam, ³ Ameya Kamble, ⁴ Suraj Belhekar, ⁵ Fazal S

¹Professor, ²Student, ³Student, ⁴Student, ⁵Student

¹Computer Engineering

¹Dr. D. Y. Patil Institute of Engineering Management and Research, Akurdi, Pune, India

Abstract: This paper details the design and implementation of a lightweight, web-based Automated Interview Assessment System (AIAS) aimed at enhancing candidate mock interview performance through objective and comprehensive analysis. Addressing the limitations of traditional interview methods, which often suffer from subjectivity and lack standardized evaluation, the AIAS leverages the Flask framework to create a dynamic and user-friendly platform. The system begins with the secure processing of uploaded resumes, pulling key information about the candidate such as educational history, technical skills, and work experience. This pulled information is used to create a list of customized interview questions so that the mock interview would be highly applicable to the candidate's particular profile. The AIAS performs actual interview simulation in real-time, recording both video and audio of the answers of the candidate. This multimodal data acquisition allows a comprehensive assessment of the candidate's performance. A lightweight Facial Expression Recognition (FER) model is used to examine the candidate's non-verbal behavior, offering insights into his or her confidence, emotional state, and pressure poise. In addition, voice analysis is conducted by transcribing the audio through speech-to-text technology and then a careful evaluation of speech fluency such as marking pauses, hesitations, and filler words and also marking grammatical accuracy and sentence formation. The AIAS is intended to deliver real-time and actionable feedback to the candidate to raise self-awareness and enable specific improvement in interviewing skills. The system focuses on a privacy-oriented paradigm, providing safe handling of data and reduced storage of sensitive data. Interestingly, the AIAS accomplishes real-time feedback without the use of computationally intensive deep networks like Convolutional Neural Networks or Recurrent Neural Networks. This design makes the system more efficient and deployable in resource-limited setups. This report clarifies the workflow of the system in operation, outlines the specific metrics used for evaluating facial expressions, fluency of voice, and grammatical correctness, and offers future directions as possible areas of improvement, such as incorporating more non-verbal features and investigating more advanced yet lightweight natural language processing strategies.

Keywords—Machine learning, automated interview evaluation system (AIAS), facial expression recognition (FER), application programming interface (API), prediction models, speech-to-text, Flask, video capture, emotion analysis, voice confidence.

INTRODUCTION

The conventional interview process, as a cornerstone of candidate assessment, is itself vulnerable to subjectivity through interviewer prejudices, unequal application of assessment criteria, and the unreliability of measuring subtle facets of candidate performance. These weaknesses can generate inequitable judgments, block the identification of the best candidates, and create inefficiencies in hiring. Identifying the necessity for more unbiased, uniform, and effective assessment tools, the industry of recruitment, human resources, and education has increasingly looked towards automated systems and artificial intelligence. This paper presents a novel Automated Interview Assessment System (AIAS), a web-based application created using the Python Flask microframework. The AIAS realistically mimics actual interview situations, giving candidates instant and objective feedback regarding key areas of their performance, thus remedying the flaws in conventional approaches.

The AIAS is a complete solution for improving mock interview practice and preparation. The workflow of the system starts with the candidate submitting their resume via a secure web portal. This file, a treasure trove of knowledge regarding the background of the candidate, is astutely analyzed with natural language processing methods to identify essential facts regarding their educational backgrounds, technical skill and know-how, project work and achievement, and work experience.

This data that is unearthed serves as the basis for the forthcoming phases of the interview simulation. The mock role simulation is accompanied by a fully automated mock interview that takes into consideration the applicant's specific resume and biographical details obtained prior to the interview. They can collect all relevant data, which can include past achievements, their personality type, work style preferences and even hobbies. Such data may be collected from work portfolios or social networking platforms like LinkedIn. Such information is important since in the case of an actual interview, the interviewer would have very limited time to conduct profiling and question formulation. Therefore, having this information readily available to the system will greatly enhance the accuracy and relevancy of the questions posed to the interviewee for maximum usability and experiential learning. In line with the above, the system is capable of high precision with regard to the information fed into it regarding the resume and biography of the candidate. Such a profound level of accuracy ensures that the questions crafted for the mock interview shall remain pertinent and adaptable to

developments regarding social context and expectations. The system will also have the ability to dynamically adjust in real-time the follow up questions based on answers the candidate provides. Voice fluency and clarity are established by transcribing the audio using speech-to-text software and then scanning the output text for evidence of anxiety, such as pausing, hesitation, and filler words. Finally, the system verifies grammatical accuracy and sentence information to assess the verbal communication skills of the candidate. The second distinguishing characteristic of this system is its emphasis on practicality and efficiency. The AIAS is founded on efficient and interpretable models, such as a highly efficient FER model and rule-based grammar checkers, rather than computationally intensive deep learning models such as Convolutional Neural Networks or Recurrent Neural Networks. This is done by the design choice of having instant delivery of feedback, decreasing time for processing, and simplicity in deployment, thus making the system accessible to more users educational organizations, and training institutions with varying amounts of computational capabilities.

In addition, the system has a privacy-led design that focuses on secure management of candidate data and reduces storage of sensitive information. This paper explains the workflow of the system in great detail, specifying the particular metrics used to evaluate facial expressions, voice fluency, and grammatical correctness. It further explains why the light models were chosen and how they are appropriate for real-time feedback. Lastly, the paper proposes future directions for improvements, such as incorporating other non-verbal responses, investigating more advanced, yet computationally minimal, natural language processing methods, and exploring adaptive feedback processes to tailor the training experience for every candidate... detection systems will play a key role in global initiatives for the optimization of crop management practices. These advances have the ability to reduce pesticide dependence while encouraging sustainable agricultural practices. Therefore, as these technologies evolve further, they will offer solid support for the advancement of the agricultural industry in addressing existing challenges.

II. RESEARCH AREA

2.1 Problem statement

AIAS specifically responds to the growing need for scale-up and unbiased mock interview sites in education, hiring, and career advancement. One major challenge for students, job applicants, and career changers is the absence of timely, unbiased, and constructive critiques of their interviewing style, grammar skills, and emotional display within mock interview settings. Traditional mock interviews often rely on subjective feedback from career counselors, mentors, or peers, which can be inconsistent, time-consuming, and limited in scope. The quality of feedback can vary significantly depending on the experience and expertise of the evaluator, and the process is often not standardized, making it difficult to compare performance across different individuals. Furthermore, the rising popularity of online and remote interviews has further emphasized the necessity for effective tools that can measure candidate performance in virtual environments. In virtual environments, non-verbal information, including facial expressions and body language, which are essential for good communication, might be more difficult to understand correctly. This calls for the creation of computerized systems that can offer objective and dependable measurements in such environments.

Although advanced AI models are available for many test tasks, such as facial expression recognition, speech processing, and natural language analysis, their high computational demands make it often impossible to deploy them in light, web-based applications. Such models, especially deep learning frameworks like Convolutional Neural Networks and Recurrent Neural Networks, usually demand much processing power, vast amounts of training data, and specific equipment, such as GPUs, to perform well. This constraint prevents accessibility and the provision of instant feedback, which is essential for optimum learning and self-improvement. Most current systems demand dedicated hardware, high-end compute, or large cloud infrastructure, thus rendering them infeasible for single users, budget-constrained educational institutions, or organizations looking to make such tools a part of their current workflows. The time gap between the mock interview and feedback provided can also lower the efficacy of the learning experience, as it may make candidates less likely to remember certain details of their performance.

This system seeks to fill this gap by providing an effective, easy-to-use, and low-cost platform that offers important insights into candidate performance without requiring vast computational resources. The AIAS concentrates on creating and combining light weight AI models and algorithms that can perform satisfactorily in the limitations of a typical web browser and an average personal computer. More people can use the system in this fashion, especially those with little technological expertise or resources. The one that, because AIAS is scalable and deployable, it can be used in e-learning platforms, career centers, and educational institutions. Applicants can enhance their prospects of success in live interview situations by harnessing the AIAS's automated, objective, and real-time feedback in order to determine their strengths and weaknesses, practice interviewing techniques, and build confidence. Moreover, the system provides educators and career counselors with an informative tool that allows them to provide more comprehensive and personalized advice to their customers and students. Data acquired by the system can also be used to track candidate improvement over a time period as well as decide where additional training or support would be worth it.

2.2 Data Input and Interview Context

The present study employs a tailored and dynamic method of inputting data, unlike most AI-based assessment tools which operate under static, pre-established sets of data to train and test against. The résumé uploaded by the candidate is the AIAS's singular and primary source of data. Every candidate's CV is an individualized representation of his/her singular background, experiences, and career goals.

This approach enables it to be able to construct context-specific and highly relevant interview simulations that can reproduce the individualized nature of in-person interviews.

Their résumé is automatically processed in real-time through natural language processing (NLP) algorithms when they submit it via a secure online application portal. This smart parsing action retrieves important information related to the candidate's academic credentials, such as degrees, institutions attended, and dates attended; technical skills and abilities, covering programming languages,

software tools, and technical certifications; project experience and achievements, emphasizing their role, contribution, and outcomes; and work experiences, outlining their work history, roles played, and accomplishments. The system uses a combination of methods such as text extraction, named entity recognition, and dependency parsing to properly extract and retrieve relevant information from the resume. The application of NLP allows the system to deal with differences in resume formats and to extract information even when it is presented in an unstructured format.

This organized data, being a complete description of the candidate's background and qualifications, is then standardized into a representation, for example, a JSON object. This standardization allows for compatibility and easy transfer to the question generation API. The data is then sent as a request to a API. This API is an independent web service, merely for generating interview questions from given resume information. This API utilizes algorithms, rule-based systems, and possibly machine learning models to prepare a well-balanced list of technical questions, inquiring the applicant's domain knowledge, problem-solving skills, and technical skills, and non-technical questions, evaluating their soft skills, behavior, communication skills, teamwork skills, and fit for a prospective job. The API is created to produce questions that are specifically applicable to the contents of the resume and develop a custom and interactive interview experience.

Having a distinct API for generating questions provides flexibility in maintaining and enhancing the logic of question generation without the need to change the core AIAS application.

The whole mock interview setup is replicated within the comfortable web browser, using the candidate's own web cam and microphone. This browser-based model allows remote and interactive interview sessions without needing customized software, hardware installations, or specific operating system compatibility. Candidates can be accessed anywhere with an internet connection, ensuring flexibility and ease of use. The system records two major data streams: video, which is then utilized by the FER model to analyze facial emotions, and audio, which is converted to speech-to-text for voice and grammar analysis. Multimodal data recording creates an enriched set of data to evaluate the candidate's interview performance completely. The web-based platform also makes it convenient for integration with other web-based learning management platforms or career services portals, making the process of adding mock interviews to current education or training programs seamless. The system is made to be easy to use and intuitive, with minimal technical skills needed from the candidate.

III. METHODOLOGY

3.1 System Architecture Overview

The Automated Interview Assessment System (AIAS) is designed as a web application using the Flask micro framework, a light-weight and modular Python-based web framework, for effective server-side management, efficient routing, and smooth integration of its different functional modules. The modular nature of the AIAS encourages maintainability, scalability, and simplicity of future development. Each module is intended to carry out a discrete function, and the modules interact with one another using clearly defined interfaces. The modularity enables the development and testing of each component independently, which simplifies the overall system. The major modules of the AIAS, which interact cooperatively to generate a complete interview assessment, are:

Resume Parsing and API-based Question Generation: This module takes care of the secure upload and smart parsing of the resume of the candidate. It thereafter interacts with an external API, a different service meant to come up with a list of customized interview questions based on the parsed data from the resume. Independent development and updating of the question generation logic are made feasible by the modular structure. By using an API, the system can also leverage existing question generation services or integrate with other platforms. **Real-time Webcam and Audio Input Capture at the Time of Interview:** This module records the concurrent webcam and microphone input of the candidate while he/she responds to the questions designed in the web interface in real time. This module uses the browser's own media capture capabilities, i.e., the `getUserMedia()` API, to record the responses of the candidate in real time. After capture, the audio and video streams are passed on to the corresponding modules for processing.

Emotion Analysis (FER): This module uses a simple Facial Expression Recognition (FER) model to the video recording stream. The FER model reads the candidate's facial expressions and detects and classifies their emotional states, reporting on their confidence, anxiousness, and overall emotional presentation. A key aspect of interview performance, the candidate's non-verbal communication skills are evaluated in this module. **Evaluation of Grammar and Confidence in Voice:** This module processes the audio stream that was recorded. Initially, it applies a speech-to-text (STT) engine to transform the audio into a textual transcript processor. Once it processes the audio signal, the STT engine transcribes the words of the candidate when he or she speaks into text. Then the text is analyzed for confidence and speaking fluency hints such as the use of filler words, pauses, and hesitations. A grammar checking software then examines the text to review the candidate's verbal answers' overall consistency, sentence structure, and grammatical correctness. The module evaluates the candidate's verbal communication abilities, such as expressing themselves briefly and understandably.

In order to achieve seamless and efficient interview simulation and assessment, the data exchange and interactions between such modules are also properly managed. The solution has low latency adequately and offers immediate feedback to the candidate.

3.2 Resume Processing and question generation

The candidate sends over their resume via a secure online application interface to initiate the interview process. For document uploads in the most commonly used formats, including PDF, DOC, and DOCX, the AIAS offers a user-friendly interface. Once the document is uploaded, the resume parsing library of the company is specially invoked to search for and extract salient information from the uploaded document in an intelligent way. The library employs a harmony of procedures, such as text retrieval, OCR, pattern recognition, and semantic analysis, to effectively extract useful information. The specific skills employed will vary depending on the uploaded document type.



The information yielded is:

Educational Qualifications: Degrees obtained, institutions studied at, dates attended, courses studied, academic honors.

Technical Skills and Proficiencies: Programming languages, software tools, hardware platforms, technical certification, domain-specific knowledge, technical expertise.

Project Descriptions: Description of relevant projects, covering project names, descriptions, role, responsibility, contribution, outcome, and technologies applied.

Work History: Last employment information, including titles at the workplace, names of companies, dates of work, tasks performed, accomplishments, and skills applied.

The resume parsing library is implemented to manage differences in resume formats and styles so that the system can properly extract data from a broad spectrum of documents. The system can also include error handling controls to manage situations wherein the process of resume parsing is not smooth, such as when the document is of bad quality or has non-standard content.

This organized data, a complete summary of the candidate's history and credentials, is then put into a normalized representation, i.e., a JSON object. This normalized form ensures interoperability and allows for easy transfer to the question generation API. The data is then sent as a request to a RESTful API. This is a standalone web service API specifically tasked with creating interview questions based on the resume input provided. The API uses algorithms, rule-based systems, and possibly machine learning algorithms to produce a balanced set of technical questions, testing the candidate's knowledge in their domain area, problem-solving skills, and technical skills, and non-technical questions, evaluating their interpersonal skills, behavioral aspects, communication style, teamwork skills, and general fit for the probable job. The API is made to make questions directly related to the information in the resume, providing a customized and interactive interview experience. Having a standalone API for question creation gives flexibility in the updating and refining of question generation logic without impacting the underlying AIAS application. The API can further include methods to ensure the generated questions are varied, tough, and suitable for the experience level of the candidate. Most importantly, to protect data privacy and security, after the resume has been parsed successfully and the generated questions have been returned by the API, the original resume document and the temporarily extracted data are deleted instantly and securely from the server of the system. This helps prevent sensitive candidate data from being stored persistently and reduces the threat of unauthorized access. Appropriate authentication and encryption mechanisms, e.g., HTTPS, are used to secure communication between the AIAS and the question generation API in order to protect data in transit. The system also complies with applicable data privacy laws, e.g., GDPR, to ensure candidate data handling according to norms.

3.3 Interview Execution

The web interface shows the candidate the generated interview questions one at a time in sequential order, showing a reasonable amount of questions at a time to ensure a natural flow of conversation and avoid overwhelming the candidate. In this view, five questions are shown at a time. The questions are clearly and succinctly displayed, and the candidate is provided enough time to prepare answers. For every question, the candidate is asked to answer aloud in front of their webcam.

The system makes use of the browser's native media capture feature, namely the `getUserMedia()` API, to record both the video and audio of the candidate's answer at the same time. The API grants access to the user's camera and microphone, so the system can record the data it needs. The facial expressions, body language, and eye contact of the candidate are captured by video recording, while their speech, tone of voice, and speech patterns are captured through audio recording. A comprehensive evaluation of the candidate's performance requires multimodal data gathering since it allows the system to evaluate both the verbal and nonverbal aspects of their communication. The candidate receives appropriate instructions from the system on how to position themselves in front of the webcam and how to clear their audio every recorded response, together with the corresponding query, is stored temporarily in either a secure temporary storage location or the browser's memory. This temporary storage enables subsequent processing and analysis of the recorded data by the emotion analysis, voice confidence, and grammar evaluation modules. The recorded data is not stored on the server permanently until the candidate finishes the entire mock interview session. The system uses proper memory management techniques so that the recorded data does not eat up too much of the resources and the browser stays responsive.

3.4 Emotion and Confidence Analysis

The AIAS uses a multi-modal strategy to evaluate the emotional state and level of confidence of the candidate during the interview from visual and audio inputs. This method gives a stronger and richer evaluation compared to using one modality in isolation.

Facial Confidence: Each response's recorded video is inspected with a lightweight Facial Expression Recognition (FER) model. This model

is learned to recognize and classify faint facial expressions corresponding to various emotional states, such as anxiety expressions (e.g., furrowed brows, tense lips, and fast eye movements), neutral expressions, and confidence expressions (e.g., relaxed facial muscles, direct stare, and faint smile). The FER model inspects the video frames to extract and follow these facial action units across time. Facial action units refer to individual muscle movements that are related to various emotions. Through examining the frequency, strength, and length of such identified expressions during the interview, the system delivers a rating of the overall facial confidence, emotional stability, and pressure proofness of the candidate.

The FER model utilized here is more focused on interpretability and computational speed. Instead of utilizing deep convolutional neural networks that require a high amount of computing, it identifies meaningful face features and their movement based on a combination of pre-defined rules and feature extraction methods. The model is able to utilize methods such as Local Binary Patterns (LBP) for extracting features and the Viola-Jones method for detecting faces. The system can be implemented in an environment with low processing capability due to its design, which facilitates rapid processing and immediate feedback. To make the FER model less sensitive to environmental variations, the system also includes provisions for dealing with variations in lighting and head position.

Voice Confidence A functioning speech-to-text (STT) engine initially transforms each response's audio recording into text. Following audio signal transcription, the STT engine produces a text-based representation of the candidate's verbal responses. The system employs a pre-trained STT model, which could be developed by implementing the theory of language and acoustic modeling. The candidate's voice confidence and fluency scores are then analyzed in the textual encoding of their speech. The system recognizes instances of speech irregularities like repetition of words or phrases (e.g., "like," "you know," "um"), the frequency and length of pauses or hesitations within sentences, and any point where the candidate interrupts himself/herself or suddenly switches train of thought. Such irregularities might signal nervousness, poor preparation, or inability to articulate thoughts. Greater frequency and duration of such ungrammaticalities are signs of lower confidence in the voice, possible nervousness, and lack of fluency. The system measures statistics like the frequency of filler words per sentence, the average pause time, the proportion of speech time to overall response time, and the frequency of self-interruptions.

Grammar Confidence: The same transcript of text produced from the audio is then processed by a specific grammar checking component. This component examines the grammaticality, sentence structure, and overall coherency of the candidate's verbal answers. It flags up errors in grammar (i.e., misconjugated verbs, subject-verb agreement, misusage of tenses), syntax (i.e., unidiomatic phrasing, misordering of words, sentence fragmentation), and punctuation (based on pauses, intonation, sentence structure). The result of this module is a score for the grammatical correctness and naturalness of the candidate's oral speech. The software utilizes well-established and effective grammar checking libraries, which use rule-based systems and statistical models, to conduct this examination without the computational expense of sophisticated natural language processing models. Techniques like part-of-speech tagging, syntactic parsing, and error detection rules can be employed by the grammar checking module.



The system derives individual scores for the following important metrics:

Facial confidence score: The overall prevalence of confident facial expressions over anxious or uncomfortable ones. This score comes from the computation of facial action units and the corresponding emotional states. Voice fluency score by quantity and extent of speech abnormalities identified in the transcript, which reflects how fluid and logical the candidate's speech was. Measures such as speaking time ratio, pause time, and frequency of filler words are employed to compute it. The quantity and severity of grammatical errors found establish the candidate's grammar and sentence structure score, a measure of the clarity and coherence of their spoken discourse. This is computed from the output of the grammar checking module.

A pre-defined weighting equation is then utilized to blend these discrete results together to produce the final composite interview confidence measure. Depending on the particular needs of the interview situation or the users' personal preferences, the weighting method can be altered. The online interface presents the user with this aggregate score along with complete question-by-question comments based upon each analysis element. The feedback includes exact suggestions for the improvement of the candidate's performance in future interviews by pointing out the emotions, speech defects, and grammatical mistakes that were detected. To make user understanding more effective and stimulate self-reflection, the system also provides visual displays of the results of the analysis, including graphs and charts.

IV. RESULT AND DISCUSSION

The AIAS offers an extensive and impartial assessment of a candidate's performance in simulated interviews. The ability of the system to create customized interview questions from resumes provides assurance that the simulation is realistic and realistic. Recording and analyzing audio and video data in real time provides an exhaustive review of both verbal and nonverbal communication skills.

The Facial Expression Recognition (FER) model offers intelligent feedback regarding a candidate's self-confidence, emotional balance, and pressure composure by correctly identifying and classifying a spectrum of emotional states. The system is appropriate for web-

based implementation due to its focus on light models, which ensure efficient information processing and timely feedback. The voice analysis module accurately gauges speech fluency by tagging pauses, hesitations, and filler words in the transcribed voice. Such an analysis accompanied by the grading of grammatical correctness and sentence structure provides an extensive assessment of the candidate's verbal communication ability. The ability of the system to give instant and actionable comments allows candidates to realize their strengths and weaknesses and better their interview skills.

The AIAS has several advantages over other forms of interview evaluation. The objective and standardized nature of the system eliminates subjectivity and prejudice from human evaluation. The real-time feedback process allows candidates to learn and remedy their mistakes and do better in subsequent mock interviews. The scalability and ease of use of the system allow it to be accessed by schools, career services centers, and individuals who wish to prepare themselves for interviews.

V. Conclusion

The Automated Interview Assessment System (AIAS) is a significant breakthrough in simulated interview technology. Leveraging lightweight AI models and user-friendly web interface, the AIAS provides respondents with objective, personalized, and actionable behavioral feedback regarding their performance in interviews. The efficiency, practicability, and privacy-oriented nature of the system render it a valuable

resource for a wide range of users and institutions.

VI. ACKNOWLEDGMENT

All authors declare that they have no conflicts of interest.

All participants provided informed consent before taking part in this study. Personal data was anonymized to ensure confidentiality.

VII. REFERENCES

- [1] P. Ekman, "Facial Action Coding System (FACS): The Manual," Consulting Psychologists Press, Palo Alto, CA, 2002.
- [2] Y. Lu and Y. Zhang, "Speech Emotion Recognition Using CNN and RNN Techniques," *IEEE Access*, vol. 7, pp. 123, April 2019.
- [3] S. Huang and L. Wang, "Automated Knowledge Assessment in Interview Simulations Using NLP," *Proceedings of the IEEE International Conference on Data Science and AI*, pp. 45-52, 2021.
- [4] G. Schuller, "Real-Time Emotion Recognition for Interview Assessments," *IEEE Transactions on Affective Computing*, vol. 12, no. 4, pp. 300-309, July 2022.
- [5] J. Smith and A. Brown, "Natural Language Processing Techniques for Automated Interview Evaluation," *2023 International Conference on Computational Intelligence and NLP (CINLP)*, IEEE Xplore, June 2023.
- [6] M. Johnson and R. Lee, "AI-Driven Voice Analysis for Confidence Evaluation in Interviews," *Journal of Artificial Intelligence Research*, vol. 48, pp. 12-19, 2020.
- [7] T. Clark, "Improving Job Interview Outcomes with Emotion Detection Systems," *Proceedings of the 2020 IEEE International Conference on Human-Centered Computing*, IEEE Xplore, August 2020.
- [8] S. Gupta, "Challenges in Emotion Recognition Using Machine Learning Models," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 5, pp. 1555-1565, May 2021.
- [9] K. Wu and J. Li, "Advanced Techniques for Real-Time Voice Confidence Analysis in Interview Simulations," *Proceedings of the 2022 International Conference on Speech and Language Processing*, IEEE Xplore, February 2022.