InterviewPulse:AI-Powered Mock Interview Platform

Aaroh Dinkar
aaroh.dinkar221@vit.edu

Aryan Dharmadhikari
aryan.dharmadhikari221@vit.edu

Aagam Kothari
aagam.kothari22@vit.edu

Abstract: This paper presents an intelligent embedded mock interview system designed to assist candidates in preparing for real job interviews through a conversational voice-based agent and automated performance evaluation. The system incorporates strict proctoring measures using a novel adaptive calibration-based computer vision approach, which relies exclusively on head pose estimation (yaw and pitch) to detect distraction, eliminating the need for heavyweight YOLO models. Google's Speech-to-Text and Text-to-Speech APIs enable seamless interaction between the candidate and the system. For semantic answer evaluation, a lightweight NLP model—all-MiniLM-L6-v2—is used in conjunction with the Claude AI API to balance speed and efficiency.The application runs on a high-speed embedded computing device and provides personalized feedback after each interview session. This end-to-end solution offers a scalable, efficient, and fair platform for autonomous interview preparation with integrated malpractice detection.

Keywords: **Mock Interviews,Semantic Evaluation, Proctoring, Embedded System**

INTRODUCTION

After the post-COVID-19 transition to remote education, many academic institutions have continued utilizing online proctoring systems for assessments. According to a recent survey, 69% of instructors strongly preferred online proctored exams. In contrast, student preferences were mixed—37% favored paper-based exams, 39% preferred online proctoring, and 24% remained neutral [1]. However, concerns over academic integrity remain prevalent; 46% of students indicated that cheating is easier in online proctored environments, thereby increasing the likelihood of peer-assisted cheating [1]. Comparing conditions before and after the pandemic, incidents of contract cheating have surged by over 200%, up from 40% [2].Simultaneously, the rising emphasis on technical skills in software engineering interviews has exacerbated disparities among underrepresented student groups, with graduates from elite institutions demonstrating significantly higher success rates [3]. To mitigate such inequalities, AI-driven platforms have emerged, enabling equitable access to interview preparation through simulated mock interviews and personalized feedback. Tools such as Q&AI Mock Interview Bot utilize models like BERT and ChatGPT to tailor questions and responses based on user performance [4].

Our proposed system follows this paradigm by offering a lightweight, embedded mock interview platform featuring a conversational voice agent, head-movement-based proctoring, and semantic answer evaluation. The system integrates Google's Speech-to-Text and Text-to-Speech APIs for real-time voice interaction. It utilizes DeepSeek, a high-performance language model, alongside all-MiniLM-L6-v2 to generate and evaluate candidate responses through cosine similarity-based semantic matching. These AI-driven systems not only provide a level playing field for learners from varied backgrounds but also offer scalability, feedback, and fairness in remote assessment settings [5]. Additionally, immersive learning solutions such as VR-based interview training have shown promise in addressing underrepresentation, with 84% of students reporting satisfaction with such platforms [6]. Together, these innovations illustrate how intelligent systems can help bridge educational and professional gaps in a post-pandemic world.

I. LITERATURE REVIEW

Computer-vision based proctored approach with human intervention implemented by Convolutional Neural Network (CNN) based algorithm, YOLOv3 for object detection like books, mobile-phones and absence of interviewee with a custom-made dataset containing 98 high-quality videos. The model had a precision of 98% detecting malpractices and a total accuracy of 87% [7]. SkillUpBot utilizes AI-driven video and audio analysis through CNN, Long Short Term Memory (LSTM) models respectively. Communication skills, emotional cues and body language are evaluated with 84% accuracy by these models. The resume builder powered by Large-Language models and Gemini-APIs is included in the platform. The assessment of the candidate is provided by using RAVDESS and FER2013 data sets for emotion recognition and integrating ATS for resume analysis [8]. CV based system evaluated candidates based on three primary factors, emotion, confidence, knowledge base and generated a report at end of mock interview, highlighting strengths, weaknesses and suggesting where person should improve. Used CNN for facial recognition, Deep Neural Networks for speech and emotion recognition and Natural Language Processing (NLP) for appropriate conversation and feedback generation [9].

The use of AI in recognizing personality traits from video interviews is analyzed. Analyzed facial expressions and audio data from asynchronous video interviews of 120 job applicants using deep learning techniques like CNN. The big five personality traits can be identified with accuracy ranging from 90.9% to 97.4% [10]. Evaluation of candidates was based on facial emotion recognition, speech confidence evaluation, and knowledge assessment. CNN algorithm for emotion classification from facial expressions and from a dataset of 2882 training and 7066 testing images for performance check had been taken across seven emotions. LSTM-based emotion recognition is an audio file for speech feature analysis. A dataset of 2800 samples were taken which included face recognition, data separation, feature extraction, and scoring based on emotion, confidence, and knowledge, conceived with details of performance evaluation [11]. A research model proposed an automated prediction and analysis of job interview performance. Users got feedback from gesture recognition, facial expression and dialogue management blocks. The performance of the proposed model was evaluated based on Tension rate, Time consumption and Practice Cost. Results showed that candidates were more tense in the physical interview rather than the online interview [12].

NLP and ML-based assessment was done by behavioral Event Interview method to reduce bias introduced by interviewers in assessing candidates. It involved: two-stage data training-tokenization, Bayesian-inference, term frequency analysis and testing, where bot evaluated the response to predict competency levels. Enhanced the accuracy of assessment, by including statistical methods that train data and NLP techniques in the processing of unstructured text [13]. Detecting fraudulent activities during online examination in real-time was proposed. It detected the candidate and multiple people for which Deep Metric Learning is used and prohibited object detection for which Region based CNN inceptionV2 architecture pre-trained on COCO dataset is used. For mobile detection, a dataset using a 2-megapixel webcam was used. Total dataset size was 2500 images. The verification accuracy of 81% and for prohibited object detection, mean average precision of 90% [14]. Introduced fairness in the automated interview process by using adversarial training and 1-Wasserstein-Distance. Multimodal datasets like Hiring Recommendation dataset and First Impression Dataset were used which contained actual interview videos. For fairness check, metrics Strongly Paired Demographic Disparity (SPDD) and Strongly Paired Equal Opportunity (SPEO) were implemented. Used library pyAudioAnalysis for audio feature extraction. Compared results with the models, Vanilla, Data Balancing and Adversarial Learning. Accuracy of 84.5% with better SPDD was achieved [15].

The Low-resource Mock-Interview Generator model, a dialogue-based application extracted essential information for generating questions with the help of user's resume. Semantic meaning of text identified using Bidirectional Encoder Representations from Transformers. Real World Dataset by 'Boss-Zipin' and Weibo Dataset were used on TensorFlow. A dialogue generator predicted the next words of a response based on the prior-sequence which employs a hierarchical encoder. Each utterance was learned by a self-attention mechanism using BERT, ReLU activation function and softmax function for response decoder, along with two blocks, knowledge selector and decoding manager. [16]. Fillers in mock job interviews were studied. Frequent fillers like 'ahm' and 'uhm' were used, especially under stress with males using more repetitive words. [17]. Using ChatGPT, Human-Computer Interaction education can be enhanced by generating user personas. A survey of 456 students identified challenges in user research with Qualitative and quantitative analysis of survey responses. [18].

Avatar-based feedback examined job interview performance through a controlled experimental approach. The respondents were split into two groups: self-feedback and avatar-based feedback groups. Initial anxiety level was determined using Measure of Anxiety in Selection Interview and Self-Assessment Manikin questionnaires. For real-time anxiety, Shimmer3 GSR+ device analyzed Skin-Conductance-Response. NeuroKit2 was implemented to process data from Galvanic Skin Response. Aimed to facilitate self-appraisal and improve stressful interview situations [19]. Emotion and confidence analysis aimed to improve feedback accuracy and efficiency. The evaluator analyzed language patterns, facial expressions, tone during mock interviews and provided feedback on candidate's confidence levels and emotional states. Deep learning and language-processing algorithms were employed. Metrics such as accuracy, precision, recall and F1-score are used [20]. An automated question classification and behavioral interventions improved interviewing. 42 employees participated in the interviews and received no intervention, feedback, or sampling training. An XGBoost model classified questions achieving a classification accuracy of 71% using resampling and additional rules to increase accuracy [21].

Open-source libraries such as dlib and OpenCV extracted video frames, detected facial landmarks, and created mirrored left-left and right-right facial composites. Structural Similarity Index was computed to quantify asymmetry, reflecting differences between controlled and genuine emotional displays. The discrepancies indicated incongruence in emotional expression, providing unbiased support for interview assessments [22]. A virtual recruiter is used to recognize and respond to users' social cues in real-time, achieving 88.64% accuracy in behavior recognition like open arms, hands behind head. User studies showed 5 out of 6 participants gave positive remarks for interview preparation [23]. The system integrated 6 modules-business, training, interview, social, pricing and evaluation to provide immersive experiences using virtual reality, NLP and sensor-based image analysis. For anxiety level reduction of the candidate during the interview, it integrated gamification, scoring and social media integrations [24]. Nonverbal cues like smiling and head movements in video job interviews, employing a framework that integrates CNN and LSTM for feature extraction. It demonstrates that these cues correlate strongly with personality traits, especially extraversion, enhancing the estimation of sales abilities and candidate hire ability [25].

III. METHODOLOGY

The proposed system offers a mock interview platform for software development engineering (SDE) roles, in building a strong foundation of fundamental subjects of SDE with proctoring measures for avoiding malpractices. A voice agent offers realistic interview experience using Google's Speech-to-Text and Text-to-Speech libraries and is evaluated using Natural Language Processing (NLP). This system enhances SDE preparation with proctoring, coding tests, Generative AI based Q/A round, semantic similarity evaluation, and detailed performance feedback. This approach improves the credibility of the mock interview system and makes users aware about the type of questions industries ask and what skillsets are required to attain that role.

System Architecture

The system is divided into two modules, the mock interview module and the proctoring module for invigilation.

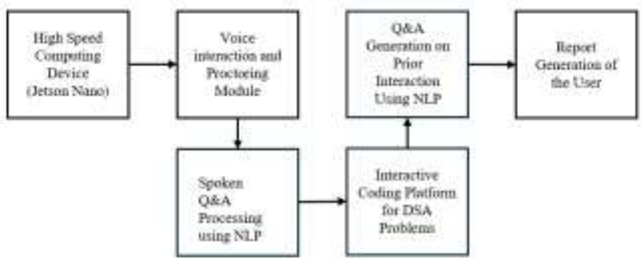
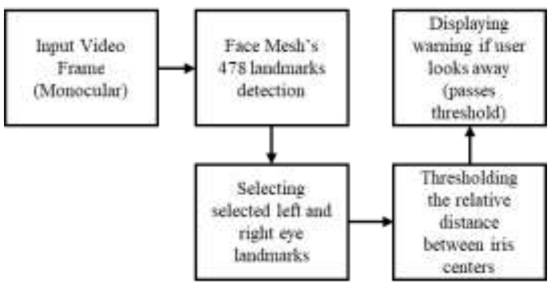


Fig. 1. Mock Interview System

In figure 1, the mock interview system comprises a high speed computing device (Jetson Nano Board) which is responsible for running the Deepseek 1.5B LLM model for inference in Stage 3. In 1st stage, ten fundamental questions on four subjects are asked, followed by a coding test of four questions, a question-and-answer round based on the coding test using Deepseek 1.5B LLM model, and a final report based on the candidate's performance after all three stages. Everything in this module is controlled by a voice agent which listens and assesses the answer given by the person and also asks the questions from the questionnaire to the person through text as mentioned in figure 1.

The Proctoring Unit, active at all stages, assesses user



eye-ball movements, alerting if user looks away for a long time. For tracking purposes, we have used the media pipe's face mesh containing 478 face landmarks. The FaceLandmarker uses 3 models, BlazeFace, a custom CNN model based on Single Shot Multibox Detector architecture, for face detection, FaceMesh-V2, CNN model based on custom MobileNetV2 architecture and Blendshape, a CNN model utilising the MLP-Mixer model. The block diagram for the same is given in Figure 2. The dataset for stage 1 and stage 2 are custom-made and were prepared from various sources such as GeeksForGeeks, GitHub repositories and platforms like

Fig. 2. Proctoring Unit

LinkedIn. The Stage 1 dataset contains 500 predefined SDE subjects' questions, while the Stage 2 dataset includes 90 coding questions on Data Structures and Algorithms, divided into easy, intermediate, and hard levels with answers available in C and C++ languages.

The essential data preprocessing steps for our questionnaire were employed such as: text normalization to ensure consistency during T2S conversion by removing extra spaces, special characters and unnecessary punctuation. Tokenization and lemmatization to improve accuracy in matching user responses with reference answers.

For stage 3, Deepseek 1.5B open source model was specifically fine-tuned over 500Qs based on data structures & algorithms. This made the question generation more accurate for stage 3 thereby testing the candidate thoroughly.

Algorithm for calculating Semantic Similarity

Algorithm 1: Semantic Similarity Calculation using Sentence Transformers

Input: Input speech converted text str_user, predefined text str_pre

Output: Similarity score sim and feedback (Accurate, Good, Needs Improvement)

Steps:

1. vec_pre ← SentenceTransformer(str_pre)
2. vec_user ← SentenceTransformer(S_user)
3. cos_sim ← $\frac{vec_{pre} \cdot vec_{user}}{|vec_{pre}| \cdot |vec_{user}|}$
4. **if** cos_sim > 0.9 **then**
5. feedback ← "Highly accurate".
6. **end if**
7. **if** 0.7 < cos_sim ≤ 0.90 **then**
8. feedback ← "Good but could be improved".
9. **end if**
10. **if** feedback < 0.7 **then**
11. result ← "Needs Improvement".
12. **end if**
13. **return** cos_sim, feedback

Sentence embeddings map sentences to high-dimensional vector spaces, ensuring similar meanings. The Sentence Transformer encodes input sentences, embedding each word into a word embedding vector. The final output is a fixed-length vector vec_user (e.g., 768 dimensions), which represents the meaning of the entire sentence. Refer algorithm 1.

For semantic similarity calculation, the primary metric is the Cosine Similarity Score, which ranges from -1 to 1, indicating how closely aligned the answers are, and is calculated as per equation (5). Feedback is categorized as: Highly accurate: score greater than 0.9, Good: score is in the range from 0.7 to 0.9, Needs improvement: score less than 0.7. The cosine similarity is calculated as given in equation (5):

$$\cos\_sim = \frac{vec_{pre} \cdot vec_{user}}{|vec_{pre}| \cdot |vec_{user}|} \quad --(5).$$

where in equation (5): vec_{pre} : predefined answer's vector representation with $|vec_{pre}|$ being its magnitude, vec_{user} : user's answer's vector representation and $|vec_{user}|$ being its magnitude.

The evaluation framework collects answers, calculates similarity scores, and analyzes results for continuous improvement of the mock interview platform, forming a robust model for our application.

Another novelty in our proposed work is in NLP tasks, instead of using traditional heavyweight models like BERT, we have used lightweight all-MiniLM-L6-v2 model for driving speed and efficiency. A novel semantic evaluation pipeline employs an approach of sentence embeddings and cosine similarity to assess candidate's performance. Using a hierarchical approach candidates's responses are categorized as "Highly Accurate", "Good but could be improved," or "Needs Improvement." This classification specifically aims to minimize false negatives, thereby aiming to increase precision and recall. Post interview, the system identifies topics and concepts which need to be improved by the candidate, providing the links to those resources in the final feedback document. This data-driven approach aims at providing constructive feedback to the candidate, streamlining the process of learning and maximizing the chances of selection.

IV. Results and Discussion

The proposed system was evaluated in two core domains: real-time behavioral monitoring using head movement detection and automated answer evaluation using semantic similarity. The proctoring module continuously monitors the candidate's head movements to detect signs of distraction or potential malpractice. Unlike hybrid systems that incorporate gaze estimation, this implementation relies solely on head pose estimation using facial landmarks extracted via MediaPipe FaceMesh. Specifically, yaw is used to track side-to-side (left-right) movements, while pitch measures up-and-down (vertical) movement. These angular deviations are computed relative to a predefined neutral head position and then classified based on fixed threshold values.

The dataset used for evaluation included 1200 observations—600 for sideways movement and 600 for vertical movement—equally distributed among directional classes (e.g., left, right, up, and down). Each observation was manually labeled and compared with model predictions. The proctoring module achieved strong performance in sideways detection, with particularly high recall, demonstrating its ability to detect lateral distractions with minimal missed events. This is a critical strength for ensuring candidate attention during remote exams or interviews. However, vertical head movement detection performed slightly below the sideways counterpart, primarily due to factors such as the natural ambiguity in nodding behavior, variability in user camera placement, and the absence of user-specific calibration, which led to slight inconsistencies in identifying intentional versus unintentional vertical motions.

Several key factors contributed to occasional misclassifications. Bright or uneven lighting conditions created shadows on the face, negatively impacting landmark detection accuracy. The use of static thresholds across all users introduced challenges due to natural variations in posture, facial structure, or seating angles, leading to both false positives and false negatives. Additionally, without dynamic calibration for each session, pitch-based detections often suffered from noise, especially when users leaned in while focusing. Face occlusions caused by hair, glasses, or partial obstructions, as well as non-frontal face angles, further contributed to landmark instability and occasional detection errors.

To enhance accuracy, several improvements are recommended. First, integrating an initial neutral position calibration step can help establish individualized yaw and pitch baselines. Replacing static angle thresholds with dynamically adapted ones—derived from short observation windows or confidence-based intervals—would enable more personalized and robust detection. Pose stabilization methods, such as applying temporal smoothing or recurrent neural networks over sequential frames, could reduce jitter and identify sustained suspicious movements rather than short-lived natural ones. Additionally, transitioning to infrared (IR) or near-infrared (NIR) cameras would eliminate lighting dependency, and retraining the FaceMesh model on real candidate data could significantly improve robustness under practical conditions.

The Table 2 below shows the results of all the calculated metrics of 500 interviews.

Metric	Value
Accuracy	88.58%
Precision	86.58%
Recall	100%
F1 Score	92.8%
Average Cosine Similarity	79.10

Table 2.Metric results for answer evaluation

From Table 2,The results showed perfect recall of 100% ensuring that no correct answer was overlooked. A precision of 86.58% indicated that some borderline or vague answers were incorrectly accepted. These false positives typically involved grammatically correct but shallow responses or those that used correct terminology without full conceptual understanding. Despite this, the F1-score of 92.8% revealed a balance between precision and recall .A cosine similarity of 79.1% demonstrated that, on the whole, candidate responses were semantically close to their respective reference answers.Several architectural choices contribute to the system's strengths. Prioritizing recall helps ensure that stylistically different yet substantively correct answers are not unfairly penalized. The use of a open-source model like DeepSeek 1.5B allows real-time scoring without human involvement, and integration with Firestore ensures seamless storage and retrieval of candidate responses, reference answers, and scores for future review.

The second core module of the system handles semantic answer evaluation. During stage 1 of each candidate's interview, answers are automatically scored based on their semantic alignment with predefined reference answers. The evaluation pipeline retrieves the question–answer pair from Firestore and sends it to DeepSeek, which generates a numeric score ranging from 0 to 10. In parallel, cosine similarity is computed between the candidate's answer and the reference answer to provide a secondary validation layer. A reference answer is considered correct if the cosine similarity exceeds 0.75. However, to maximize recall and ensure inclusiveness, the system is designed to accept answers with similarity values as low as 0.70.

For further enhancement, contextual scoring mechanisms can be integrated by adopting deeper NLP models such as BERTScore or large-scale GPT evaluators, which can assess not just word similarity but also logical flow and reasoning. Anchoring answers with critical concept keywords can prevent technically incomplete answers from being accepted. Additionally, multi-tiered feedback generation can be incorporated to provide candidates with personalized suggestions, including resources like documentation, tutorial links, or targeted review material based on their performance in specific topics.

V. Conclusion

The proposed system emphasized on proctoring based mock interview experience to make the users ready for their actual job interviews and provide them a personalised report highlighting the good and bad points. In the proctoring process, gaze estimation and head movement-based tracking is introduced on which very less amount of research was done to detect misconduct of the user. Instead of YOLO based pretrained models for proctoring, a novel adaptive calibration-based Computer Vision solution is proposed. Another novelty is in NLP tasks, instead of using traditional heavyweight models like BERT, the lightweight all-MiniLM-L6-v2 model is used for driving speed and efficiency with similar results. The system helps users in building a strong grip on Software Engineering subjects and can further be expanded to actual interviews for shortlisting in a company offering either an internship or job. Even if user faces any difficulty in answering some questions, the links of the resources of those concepts are given in the report, reducing the time wastage of the user for searching the resource online by itself.

VI. References

[1]. Nicola-Richmond, Kelli, Phillip Dawson, and Helen Partridge. 2023. "Online Proctored Exams: Rhetoric vs Reality." *Higher Education Research & Development* 43 (2): 392–405, doi: https://doi.org/10.1080/07294360.2023.2234310

[2]. Erguvan, I. D. (2021). The rise of contract cheating during the COVID-19 pandemic: a qualitative study through the eyes of academics in Kuwait. *Language Testing in Asia*, *11*(1), 34, doi: https://doi.org/10.1186/s40468-021-00149-y

[3]. Mejias, Marlon, Zef Vargas, William Ted Edwards, Gloria Washington, Legand Burge, Dale-Marie Wilson, and Luce-Melissa Kouaho. "Equity In The Preparation Of Students For Software Engineering Coding Interviews: ChatGPT as a Mock Interviewer." In *2023 Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE)*, pp. 1016-1020. IEEE, 2023, doi: 10.1109/CSCE60160.2023.00169.

[4]. J. M. C J, M. Sabi, M. Benson, G. Baburaj and S. S, "Q&AI: An AI Powered Mock Interview Bot for Enhancing the Performance of Aspiring Professionals," *2024 International Conference on Recent Advances in Electrical, Electronics, Ubiquitous Communication, and Computational Intelligence (RAEEUCCI)*, Chennai, India, 2024, pp. 1-5, doi: 10.1109/RAEEUCCI61380.2024.10547951.

[5]. Miao, Y., Huang, W., Jiang, B. (2020). Research on Interaction Design of Artificial Intelligence Mock Interview Application Based on Goal-Directed Design Theory. In: Kurosu, M. (eds) Human-Computer Interaction. Multimodal and Natural Interaction. HCII 2020. Lecture Notes in Computer Science (), vol 12182. Springer, Cham, doi: https://doi.org/10.1007/978-3-030-49062-1_15

[6]. Wilkie, LeAnn, and Joseph Rosendale. "Efficacy and Benefits of Virtual Mock Interviews: Analysing Student Perceptions of Digital Employment Preparations." *Journal of University Teaching and Learning Practice* 21, no. 1 (2024), doi: https://doi.org/10.53761/rvtxt659

[7] A. K. Srivastava, V. Tripathi and B. Pant, "Computer Vision based Online Job Interview Proctoring for Campus Placement," 2024 11th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2024, pp. 663-668, doi: 10.23919/INDIACom61295.2024.10499075

[8] Shashank Rai, Alisha Miranda, Samiya Jagirdar, Prof. Nidhi Chitalia, "Skillup Bot: An AI Driven Mock Interview Platform", 2024 International Research Journal of Engineering and Technology, Volume: 11 Issue: 04

[9] S. Sivaramakrishnan, A. Anand, B. Hemang, F. Z. Minni and A. Sahoo, "Real Time Mock Interview Evaluation using CNN," 2024 4th International Conference on Data Engineering and Communication Systems (ICDECS), Bangalore, India, 2024, pp. 1-4, doi: 10.1109/ICDECS59733.2023.10503311.

[10] H. -Y. Suen, K. -E. Hung and C. -L. Lin, "TensorFlow-Based Automatic Personality Recognition Used in Asynchronous Video Interviews," in IEEE Access, vol. 7, pp. 61018-61023, 2019, doi: 10.1109/ACCESS.2019.2902863

[11] Mandal, Rubi, Pranav Lohar, Dhiraj Patil, Apurva Patil, and Suvarna Wagh. "AI-Based mock interview evaluator: An emotion and confidence classifier model." In 2023 International Conference on Intelligent Systems for Communication, IoT and Security (ICISCoIS), pp. 521-526. IEEE, 2023, doi: 10.1109/ICISCoIS56541.2023.10100589.

[12] P. V. Chintalapati, S. S. Paluri, S. S. Nikhitha, T. S. V. Nanditha, S. T. V. L. Daneswari and P. K. Sree, "A Research Model For Automated Prediction And Analysis Of Job Interview Performance," *2024 Sixth International Conference on Computational Intelligence and Communication Technologies (CCICT)*, Sonepat, India, 2024, pp. 284-291, doi: 10.1109/CCICT62777.2024.00055

[13] Siswanto, Joko, Sinung Suakanto, Made Andriani, Margareta Hardiyanti, and Tien Febriyanti Kusumasari. "Interview bot development with natural language processing and machine learning." International Journal of Technology 13, no. 2 (2022): 274-285, doi: https://doi.org/10.14716/ijtech.v13i2.5018

[14] Kamble, K. P., & Ghorpade, V. R. (2021). Video Interpretation for cost-effective remote proctoring to prevent cheating. In *Proceeding of First Doctoral Symposium on Natural Computing Research: DSNCR 2020* (pp. 259-269). Springer Singapore, doi: https://doi.org/10.1007/978-981-33-4073-2_25

[15] C. Kim, J. Choi, J. Yoon, D. Yoo and W. Lee, "Fairness-Aware Multimodal Learning in Automatic Video Interview Assessment," in IEEE Access, vol. 11, pp. 122677-122693, 2023, doi: 10.1109/ACCESS.2023.3325891

[16] Li, Mingzhe, Xiuying Chen, Weiheng Liao, Yang Song, Tao Zhang, Dongyan Zhao, and Rui Yan. "Ezinterviewer: To improve job interview performance with mock interview generator." In Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, pp. 1102-1110. 2023, doi: https://doi.org/10.1145/3539597.3570476

[17] Evangelista–Pelaez, Cindy, Charmy Tutor-Del Mundo, and Kevin Christopher Dayuta. "Fillers Used in a Mock Job Interview: A Cognitive Analysis."

[18] J. Barambones, C. Moral, A. de Antonio, R. Imbert, L. Martínez-Normand and E. Villalba-Mora, "ChatGPT for Learning HCI

Techniques: A Case Study on Interviews for Personas," in IEEE Transactions on Learning Technologies, vol. 17, pp. 1486-1501, 2024, doi: 10.1109/TLT.2024.3386095.

[19] S. Hosseini, J. Quan, X. Deng, Y. Miyake and T. Nozawa, "Avatar-Based Feedback in Job Interview Training Impacts Action Identities and Anxiety," in *IEEE Transactions on Affective Computing*, doi: 10.1109/TAFFC.2024.3363835.

[20] Rai, Mrs & R, Abhiram & Padthe, Adithya & R, Hrithik. (2024). AI Based Interview Evaluator: An Emotion and Confidence Classifier. IARJSET. 11. 10.17148/IARJSET.2024.11442.

[21] Haginoya, Shumpei, Tatsuro Ibe, Shota Yamamoto, Naruyo Yoshimoto, Hazuki Mizushi, and Pekka Santtila. "AI avatar tells you what happened: The first test of using AI-operated children in simulated interviews to train investigative interviewers." *Frontiers in Psychology* 14 (2023): 1133621, doi: https://doi.org/10.3389/fpsyg.2023.1133621

[22] Keshari, S., Dutta, T., Mullick, R., Rathor, A., & Patnaik, P. (2023). Facial asymmetry: A Computer Vision based behaviometric index for assessment during a face-to-face interview. *arXiv preprint arXiv:2310.20083*, doi: https://doi.org/10.48550/arXiv.2310.20083

[23] T. Baur, I. Damian, P. Gebhard, K. Porayska-Pomsta and E. André, "A Job Interview Simulation: Social Cue-Based Interaction with a Virtual Character," 2013 International Conference on Social Computing, Alexandria, VA, USA, 2013, pp. 220-227, doi: 10.1109/SocialCom.2013.39.

[24] Vardarlıer, P. (2023). A System That Allows Users to Have a Job Interview Experience. *Sustainability*, *15*(22), 16031. https://doi.org/10.3390/su152216031

[25] Kassab, K., & Kashevnik, A. (2024). Personality Traits Estimation Based on Job Interview Video Analysis: Importance of Human Nonverbal Cues Detection. *Big Data and Cognitive Computing*, *8*(12), 173. https://doi.org/10.3390/bdcc8120173