INTERNATIONAL JOURNAL OF ADVANCED INNOVATIVE TECHNOLOGY IN ENGINEERING

Published by Instigative Research Private Limited Home page: www.ijaite.co.in

# AI-Powered interview preparation platform: A Comprehensive approach using NLP, adaptive question generation, and real-time feedback

[1]Prof. Leena Pimple, [2]Nilayam Jiwtode, [3]Pranay Somnathe, [4]Rajiv Ramteke, [5]Tushar Didmuthe

[1,2,3,4,5]Department of Computer Engineering, Suryodaya College of Engineering & Technology, Nagpur, Maharashtra, India

[1]sulbhewar.leena@gmail.com, [2]jiwtodenilayam@gmail.com,
[3]pranaysomnathe69@gmail.com, [4]ramtekerajiv96@gmail.com,
[5]tushardidmuthe33@gmail.com

**Article History**

Received on: 20 July 2026

Revised on: 15 Aug 2026

Accepted on: 20 Sept 2026

**Keywords:** Artificial Intelligence, Natural Language Processing, Interview Preparation, Adaptive Question Generation, Feedback System

**e-ISSN:** 2455-6491

**DOI:** 0.65809/IJAITE/26/v03sp02/025

**Production and hosted by**
www.insightiveinc.org
©2026|All right reserved.

## ABSTRACT

The escalating competitiveness of the global employment market has imposed significant challenges on students and fresh graduates seeking to secure placements in technical and non-technical roles. This paper presents the design, development, and evaluation of an AI-Powered Interview Preparation Platform (AIPP) that integrates Natural Language Processing (NLP), adaptive difficulty algorithms, and real-time feedback mechanisms to simulate authentic interview environments. The proposed system incorporates a resume-aware question generation engine, a multi-domain question repository, an automated answer evaluation module leveraging semantic similarity and keyword extraction, and a progressive performance tracking dashboard. The platform's adaptive questioning mechanism exhibited a 91.7% accuracy in difficulty calibration. These results confirm that the proposed AIPP constitutes a scalable, effective, and pedagogically sound solution to the pervasive challenge of interview preparedness among student populations.

## 1. INTRODUCTION

The transition from academic institutions to the professional workforce represents one of the most consequential junctures in a student's developmental trajectory. Despite possessing theoretical knowledge and technical competencies acquired through years of formal education, a significant proportion of students fail to demonstrate their capabilities effectively during structured employment interviews. This disconnect between academic proficiency and interview performance is well-documented in educational and occupational research, and it presents a systemic challenge that necessitate

intelligent and scalable technological intervention [1].

Traditional mechanisms of interview preparation — such as peer-to-peer mock interviews, career counselling sessions, and printed guidebooks — suffer from inherent limitations including inconsistency, lack of personalisation, unavailability at scale, and absence of objective performance metrics. Furthermore, the rapid evolution of industry-specific technical requirements demands that preparation resources be dynamic, continuously updated, and tailored to individual skill profiles. The adoption of Artificial Intelligence (AI) and Natural Language Processing (NLP) technologies has opened unprecedented avenues for addressing these shortcomings in a computationally rigorous and pedagogically sound manner [2].

AI-powered systems are increasingly being deployed across educational domains to deliver personalised learning experiences, intelligent tutoring, automated assessment, and adaptive content delivery. The application of such systems to the domain of interview preparation is both timely and strategically significant, given the scale of the problem — with millions of students globally entering competitive job markets each year. A well-designed AI interview preparation platform can democratise access to high-quality preparation resources, eliminating geographic, economic, and logistical barriers that have historically disadvantaged students from underrepresented or resource-constrained backgrounds [3].

The present work introduces the AI-Powered Interview Preparation Platform (AIPP), a full-stack web application that leverages modern AI/NLP technologies to simulate realistic interview scenarios, generate personalised questions based on user resumes, evaluate responses in real time, and provide actionable feedback aimed at continuous improvement. The system is designed with scalability, modularity, and user-centricity as core architectural principles.

*A. Key Contributions of This Work*

- Design and implementation of a resume-aware question generation engine that produces contextually relevant interview questions aligned with the candidate's declared skills and experience.
- Development of an adaptive difficulty adjustment algorithm that modulates question complexity based on ongoing assessment of user performance metrics.
- Integration of a semantic similarity-based answer evaluation module employing NLP techniques including TF-IDF vectorisation, cosine similarity, and keyword density analysis.
- Implementation of a real-time, multi-dimensional feedback framework providing granular insights across dimensions of accuracy, clarity, completeness, and communication quality.
- Construction of a scalable, RESTful full-stack architecture integrating React.js, Node.js, MongoDB, and GPT-based AI services.
- Empirical validation of platform efficacy through a structured user study involving 120 participants across multiple technical and non-technical domains.

## 2. LITERATURE REVIEW

The intersection of artificial intelligence with educational technology (EdTech) and human resource management has generated a rich body of research exploring automated interview training, intelligent tutoring systems, and NLP-driven assessment tools. A thorough review of existing literature reveals both the progress made and the gaps that the present work aims to address.

Kumar et al. [1] proposed an automated interview system employing Hidden Markov Models (HMM) for speech pattern analysis and response classification. Their system demonstrated accuracy rates of approximately 76% in evaluating spoken responses against predefined rubrics. However, the system was confined to speech-only modalities and lacked the capacity for text-based or multi-modal interaction, significantly constraining its applicability in diverse deployment contexts.

Zhang and Li [2] developed an NLP-based question answering evaluation framework that applied BERT-based transformer models to assess the semantic relevance of student responses in technical interview simulations. Their system achieved a semantic similarity score correlation of 0.83 with human evaluators on a corpus of software engineering interview responses. Nevertheless, the system did not incorporate resume-based personalisation or adaptive difficulty mechanisms, thereby limiting its ability to provide a tailored preparation experience.

Patel and Mehta [3] introduced a chatbot-based interview preparation tool built upon rule-based dialogue management and a static question repository. While the system achieved positive user satisfaction scores (mean score 3.8/5.0), its reliance on fixed question sets and deterministic response evaluation rendered it incapable of addressing the dynamic nature of real interview scenarios. The absence of machine learning components also precluded longitudinal learning or performance-based adaptation

Nguyen et al. [4] presented a comprehensive study on the use of deep reinforcement learning for adaptive educational content delivery. Their findings corroborated the effectiveness of difficulty modulation in improving learner outcomes, with participants in adaptive-difficulty conditions outperforming those in static-difficulty conditions by an average of 23.4% on post-test assessments. Although their work was situated in the broader educational context rather than interview preparation specifically, the underlying principles of adaptive learning are directly applicable to the present work.

Anand and Singh [5] explored the application of resume parsing technologies using Named Entity Recognition (NER) and dependency parsing to extract structured information from unstructured resume documents. Their parser achieved a field extraction accuracy of 89.2% across a standardised dataset, providing a foundational reference for the resume analysis component of the proposed system.

Chen et al. [6] conducted a comparative study of existing AI interview platforms including HireVue, Pymetrics, and InterviewBuddy, identifying key usability and technical limitations. Their analysis noted that most commercial platforms prioritise corporate hiring pipelines over student-centric learning, lack comprehensive feedback granularity, and provide limited transparency in evaluation criteria — creating a significant opportunity for an academically-oriented, open-architecture platform such as the one proposed herein.

Rajput and Verma [7] investigated the role of emotional intelligence simulation in interview preparation tools, demonstrating that platforms incorporating affective feedback — including sentiment analysis and confidence scoring — produced significantly higher improvement trajectories among participants compared to platforms offering only factual correctness feedback ($p < 0.05$). This finding informed the multi-dimensional feedback design adopted in the present work.

*A. Identified Gaps in Existing Research*

- Absence of end-to-end resume-to-question pipeline that personalises the interview experience based on the candidate's unique profile.
- Lack of real-time adaptive difficulty calibration in student-facing interview platforms.
- Insufficient granularity in automated feedback, with most systems providing binary correct/incorrect assessments rather than qualitative improvement guidance.
- Limited scalability and accessibility of existing solutions, many of which are proprietary, institution-specific, or require specialised hardware.
- Inadequate empirical evaluation with diverse student populations across multiple academic disciplines and interview domains.

## 3. PROBLEM STATEMENT

The contemporary employment landscape demands that candidates demonstrate not only domain knowledge but also the capacity for structured verbal articulation, logical reasoning under pressure, and adaptive communication — competencies that are rarely developed through standard academic curricula. Despite the centrality of interview performance to career outcomes, most students receive minimal formal training in interview techniques prior to entering the job market. The problem is multifaceted and encompasses the following specific challenges that the proposed system is designed to address:

*A. Inadequate Access to Practice Resources*

The majority of students — particularly those attending institutions in Tier-2 and Tier-3 cities or from economically disadvantaged backgrounds — lack access to professional career coaching, corporate mock interviews, or sophisticated interview simulation tools. Available free resources are largely fragmented, inconsistent in quality, and devoid of personalisation.

*B. Absence of Objective Feedback Mechanisms*

Peer-conducted mock interviews and self-assessment exercises suffer from inherent subjectivity and the Dunning-Kruger effect, wherein students who most need improvement are often least equipped to identify their own deficiencies. Objective, automated, and consistently applied evaluation criteria are therefore essential for meaningful skill development.

*C. Static and Non-Adaptive Preparation Content*

Existing question banks, whether in printed or digital form, present questions at uniform difficulty levels without regard for the individual student's current knowledge level. This one-size-fits-all approach is pedagogically suboptimal, as it neither challenges advanced students sufficiently nor provides appropriate scaffolding for those at earlier stages of preparation.

*D. Disconnect Between Resume Content and Practice Questions*

Students frequently encounter interview questions directly drawn from the skills, technologies, and experiences listed in their resumes. Existing preparation platforms do not leverage this highly predictive relationship, resulting in practice sessions that may be irrelevant to the specific contexts in which the student will actually be evaluated.

*E. Lack of Performance Tracking and Longitudinal Insight*

Without systematic, session-by-session performance tracking, students are unable to assess their progress over time, identify persistent weaknesses, or make informed decisions regarding the allocation of their study efforts. The absence of data-driven progress analytics represents a significant missed opportunity in conventional preparation methodologies.

## 4. PROPOSED SYSTEM

The AI-Powered Interview Preparation Platform (AIPP) is architected as a modular system comprising four primary functional units: the Resume Analyser, the Question Generator, the Interview Engine, and the Feedback System. Together, these modules form a coherent pipeline that transforms raw user inputs into personalised, adaptive, and evaluable interview experiences. The following subsections provide a detailed technical description of each module.

*A. Module 1: Resume Analyser*

The Resume Analyser constitutes the entry point of the AIPP pipeline. Upon user registration, the platform prompts the candidate to upload their curriculum vitae (CV) in PDF or DOCX format. The uploaded document is processed through a multi-stage NLP pipeline as follows:

Stage 1 — Document Parsing: The uploaded file is converted to plain text using the pdf-parse and mammoth.js libraries for PDF and DOCX formats respectively. Special handling is applied for multi-column layouts, table-formatted resumes, and non-standard fonts to ensure comprehensive text extraction.

Stage 2 — Named Entity Recognition (NER): A fine-tuned NER model identifies and categorises entities including technical skills (e.g., Python, React, SQL), educational qualifications (e.g., B.Tech, M.Sc.), organisational affiliations, project titles, and temporal expressions (e.g., internship dates, graduation year). This categorisation is performed using a spaCy-based NER pipeline augmented with a custom technology skills vocabulary derived from the BLS O*NET database.

Stage 3 — Skill Graph Construction: Extracted entities are mapped to a structured skill graph wherein nodes represent individual skills or competencies and edges represent co-occurrence relationships (e.g., 'React' and 'Node.js' frequently co-occurring in full-stack profiles). This graph serves as the semantic foundation for targeted question generation.

Stage 4 — Profile Scoring: A composite skill profile score is computed for each candidate based on the breadth and depth of declared skills, the recency and relevance of listed experiences, and any detected domain specialisation. This score is subsequently used to initialise the adaptive difficulty parameter for the Question Generator.

*B. Module 2: Question Generator*

The Question Generator receives the structured skill profile from the Resume Analyser and produces a ranked, categorised question set personalised to the candidate. Question generation operates across two complementary sub-mechanisms:

Sub-mechanism 1 — Template-Based Generation: A curated database of over 5,000 question templates, categorised by domain (e.g., Data Structures, Web Development, DBMS, OOPS, Behavioural, HR), difficulty level (L1–L5), and question type (Conceptual, Applied, Situational, Behavioural), is queried using the candidate's skill graph. Template variables (e.g., [LANGUAGE], [FRAMEWORK], [PROJECT_NAME]) are instantiated with entities extracted from the resume to produce highly personalised questions.

Sub-mechanism 2 — Generative AI Augmentation: For skill areas not covered by the template database, or when question diversity demands exceed the template supply, an OpenAI GPT-4-based generation call is invoked. The prompt construction follows a structured format that includes: (a) the target skill, (b) the desired difficulty level, (c) the preferred question type, and (d) a reference to the candidate's experience level. System-level instructions enforce formal register, technical accuracy, and consistency with IEEE interview standards.

Adaptive Difficulty Mechanism: The difficulty parameter D(t) for question at time step t is governed by the following recurrence relation:

D(t+1) = D(t) + alpha * (S(t) - theta)

Where:

D (t ) = difficulty level at time t (range: 1.0 to 5.0)

S (t ) = normalised score of response at time t (range: 0.0 to 1.0)

theta = target performance threshold (default: 0.65)

alpha = learning rate (default: 0.3)

This formulation ensures that questions become progressively harder when the candidate performs above the threshold and easier when performance falls below it, converging dynamically to the candidate's zone of proximal development.

*C. Module 3: Interview Engine*

The Interview Engine orchestrates the interactive interview session, managing state transitions, timing, user interface rendering, and response capture. The engine operates as a finite state machine (FSM) with the following states: IDLE, QUESTION_PRESENTED, AWAITING_RESPONSE, EVALUATING, FEEDBACK_DISPLAYED, and SESSION_COMPLETE.

The session configuration is parameterised by: (a) interview mode (Technical, HR, Mixed, Domain-Specific), (b) session duration (15, 30, or 60 minutes), (c) number of questions, and (d) evaluation strictness level. Users are presented with questions in a clean, distraction-free interface with optional countdown timers to simulate real interview time pressure.

Response capture supports two input modalities: (i) typed text entry via a rich text editor supporting code snippet formatting, and (ii) voice response via the Web Speech API with automatic speech-to-text transcription. Multi-modal input enables the system to be employed in preparation for both in-person and remote video interviews.

*D. Module 4: Feedback System*

The Feedback System constitutes the most critical analytical component of the AIPP, delivering real-time, multi-dimensional evaluation of each candidate response. The evaluation pipeline operates as follows:

Step 1 — Preprocessing: The captured response is normalised through lowercase conversion, punctuation removal, stopword filtering, and lemmatisation using the NLTK library.

Step 2 — Semantic Similarity Analysis: The processed response is encoded as a vector using sentence-BERT (SBERT) embeddings and compared to the reference answer embedding using cosine similarity. A similarity score S_sem is computed as follows:

$$Ssem = \frac{Er \cdot Eref}{\| Er \| \times \| Eref \|}$$

Where, $E_r$ = SBERT embedding of the candidate response, $E_{ref}$ = SBERT embedding of the reference answer, Ssem = Semantic similarity score.

Step 3 — Keyword Coverage Analysis: A predefined keyword set K = {k1, k2, ..., kn} for each question is extracted from expert-verified model answers. The keyword coverage score is computed as:

$$Skw = \frac{\sum_{i=1}^{n} I(k_i \in R)}{n}$$

Where, *K = {k1,k2,...,kn}* represents the predefined keyword set, *R* represents the candidate response, $I(k_i \in R)$ is an indicator function that returns 1 if the keyword is present in the response and 0 otherwise, n represents the total number of predefined keywords.

Step 4 — Composite Score Computation: The final response score is computed as a weighted combination:

$Score_{total}=w_1.S_{sem}+w_2.S_{kw}+w_3.S_{grammer}$

Where w1=0.50, w2=0.35, w3=0.15

Step 5 — Natural Language Feedback Generation: The computed scores, along with the original question, reference answer, and candidate response, are passed to a structured GPT-4 prompt that generates human-readable feedback. The feedback is structured into four components: (i) what was answered correctly, (ii) what was missing or incorrect, (iii) suggested improvements, and (iv) a recommended resource for further study.

E. Pseudo-code for Answer Evaluation Engine

```
FUNCTION EvaluateAnswer(question_id, candidate_response):
 question = FetchQuestion(question_id)
 reference = FetchReferenceAnswer(question_id)
 keywords  = FetchKeywordSet(question_id)

// Preprocess
 r_clean = Preprocess(candidate_response)
 ref_clean = Preprocess(reference)
 // Semantic Similarity
```

## 5. SYSTEM ARCHITECTURE

The AIPP employs a layered, service-oriented architecture that separates concerns across distinct tiers: the Presentation Layer (frontend), the Application Layer (backend), the Intelligence Layer (AI/NLP services), and the Data Persistence Layer (database). This separation ensures modularity, independent scalability, and maintainability.

*A. Architecture Overview (for Microsoft Word Diagram)*

The system architecture may be represented in Microsoft Word using the following structure of connected shapes and arrows. Draw a top-to-bottom flow diagram with the following components:

[USER] (Rounded Rectangle, blue fill) — at the top. Draw downward arrow labelled 'HTTPS Request'.

[REACT.JS FRONTEND] (Rectangle, light blue) — contains sub-boxes: (a) Login/Register UI, (b) Resume Upload, (c) Interview Interface, (d) Dashboard & Analytics. Draw downward arrow labelled 'REST API Calls (Axios)'.
[NODE.JS + EXPRESS BACKEND] (Rectangle, green fill) — contains: (a) Auth Controller, (b) Resume Controller, (c) Interview Controller, (d) Feedback Controller, (e) JWT Middleware. Draw two arrows outward: one right to AI ENGINE, one downward to DATABASE.
[AI/NLP ENGINE] (Rounded Rectangle, orange fill) — contains: (a) Resume Parser (spaCy NER), (b) Question Generator (GPT-4 API), (c) Answer Evaluator (SBERT + TF-IDF), (d) Feedback Generator (GPT-4 API). Draw arrow returning left back to Backend labelled 'Processed Results'.
[MONGODB DATABASE] (Cylinder shape, grey fill) — contains collections: (a) Users, (b) Resumes, (c) Questions, (d) Sessions, (e) Feedback. Draw upward arrow back to Backend.
[OUTPUT TO USER] (Rounded Rectangle, yellow fill) — contains: (a) Personalised Questions, (b) Evaluation Scores, (c) Detailed Feedback, (d) Progress Reports. Draw arrow from Frontend to this box.

*B. Frontend Layer*

The frontend is implemented using React.js (v18.x) with the Redux Toolkit for global state management. The application follows a component-based architecture with clearly delineated functional components for authentication, resume management, interview session execution, and analytics visualisation. Routing is managed using React Router v6, with protected routes enforcing JWT authentication. UI components are constructed using the Material-UI (MUI) component library to ensure accessibility compliance and responsive design across device form factors.

*C. Backend Layer*

The backend REST API is implemented using Node.js (v18.x) with the Express.js framework. It exposes versioned API endpoints following RESTful conventions (e.g., GET /api/v1/questions/:id, POST /api/v1/sessions). Authentication is implemented using JSON Web Tokens (JWT) with RS256 asymmetric signing. Input validation is enforced using Joi schema validators. Rate limiting and request throttling are applied using the express-rate-limit middleware to prevent abuse.

*D. AI/NLP Intelligence Layer*

The AI layer interfaces with OpenAI's GPT-4 API for both question generation and natural language feedback synthesis, and employs locally hosted SBERT models via the sentence-transformers Python library (served as a FastAPI microservice) for real-time semantic similarity computation. Resume parsing leverages spaCy v3.x with a custom NER pipeline trained on a corpus of 2,400 annotated engineering resumes. The Python-based AI microservices communicate with the Node.js backend through internal RESTful HTTP calls.

*E. Database Layer*

MongoDB Atlas (v6.x) serves as the primary data store, selected for its document-oriented data model, horizontal scalability through sharding, and native support for JSON-like BSON documents that align naturally with the application's data structures. Six principal collections are maintained: users (authentication and profile data), resumes (parsed resume objects), questions (template and generated questions), sessions (interview session metadata and states), responses (candidate responses with timestamps), and feedback (evaluated scores and generated feedback text). Indexing strategies include compound indexes on (userId, sessionId) pairs and text indexes on the questions collection for efficient full-text retrieval.

## 6. IMPLEMENTATION

The implementation of the AIPP followed an Agile development methodology with two-week sprint cycles spanning a total development period of sixteen weeks. The following subsections detail the implementation of each major system component.

*A. Frontend Implementation (React.js)*

The React frontend comprises 34 functional components organised into seven feature modules. State management uses Redux Toolkit with six slices: authSlice (user authentication state), resumeSlice (parsed resume data), questionSlice (current question set), sessionSlice (interview session state), feedbackSlice (evaluation results), and analyticsSlice (performance history). WebSocket connections via Socket.io-client enable real-time communication during active interview sessions, facilitating live question delivery and instantaneous feedback rendering without page refreshes.

The performance analytics dashboard employs Recharts (v2.x) to render interactive line charts (score trends over time), radar charts (multi-dimensional competency profiles), and bar charts (category-wise performance distributions). Chart data is derived from aggregated session history stored in MongoDB and served via dedicated analytics API endpoints.

*B. Backend Implementation (Node.js)*

The Express.js API server implements 28 RESTful endpoints across six route groups. The interview session lifecycle is managed through a server-side finite state machine that persists session state to MongoDB between client requests, enabling interruption-safe sessions. A Bull queue (backed by Redis) manages asynchronous AI processing tasks, ensuring that lengthy GPT-4 API calls do not block the main request thread. Email notifications for session completion summaries are dispatched via Nodemailer integrated with SendGrid SMTP services.

*C. Database Implementation (MongoDB)*

The MongoDB schema design employs a hybrid embedding/referencing strategy. Frequently co-accessed data (e.g., question text with its keyword set) is embedded within single documents to minimise query round-trips, while large or independently accessed collections (e.g., session records and feedback documents) are linked via ObjectId references. Database migrations are managed using the migrate-mongo library to ensure schema consistency across deployment environments.

*D. AI/NLP Integration*

The GPT-4 integration employs a structured prompt engineering approach with distinct system prompts for each use case (question generation, feedback synthesis, difficulty assessment). Temperature parameters are set to 0.4 for question generation (favouring coherence and accuracy) and 0.7 for feedback synthesis (favouring natural, varied language). Token usage is monitored and capped per session to manage API costs. The SBERT microservice is containerised using Docker and deployed as a sidecar container alongside the main Node.js application on an AWS EC2 instance, with NGINX serving as a reverse proxy and load balancer.

## 7. RESULTS AND DISCUSSION

The AIPP was evaluated through a structured empirical study involving 120 undergraduate and postgraduate students from the Department of Computer Engineering, Suryodaya College of Engineering & Technology. Participants were divided into two cohorts: an experimental group (n=60) using the AIPP for a period of four weeks, and a control group (n=60) using conventional preparation methods (textbooks, peer mock interviews). Pre- and post-assessment scores were recorded using a standardised rubric measuring Technical Accuracy (TA), Communication Clarity (CC), Conceptual Depth (CD), and Response Completeness (RC) on a 100-point scale.

*A. Quantitative Results*

Table I presents the comparative performance metrics for the experimental and control groups across all four evaluation dimensions, measured before and after the four-week preparation period.

**TABLE 1:** Comparative Performance Metrics — Experimental vs. Control Group

| Metric | EG Pre | EG Post | CG Pre | CG Post | EG Gain % | CG Gain % |
|---|---|---|---|---|---|---|
| Technical Accuracy | 54.2 | 76.8 | 53.6 | 61.3 | +41.7 % | +14.4 % |
| Communication Clarity | 49.1 | 71.4 | 48.7 | 57.2 | +45.4 % | +17.5 % |
| Conceptual Depth | 51.3 | 74.2 | 50.9 | 60.1 | +44.6 % | +18.1 % |
| Response Completeness | 47.8 | 69.5 | 48.2 | 58.4 | +45.4 % | +21.2 % |
| Overall Score | 50.6 | 73.0 | 50.4 | 59.3 | +44.3 % | +17.7 % |
| Confidence Rating | 48.3 | 68.8 | 47.9 | 56.2 | +42.4 % | +17.3 % |

EG = Experimental Group (AIPP users); CG = Control Group; Pre = pre-assessment; Post = post-assessment

**TABLE 2:** System Performance Metrics and Platform Evaluation

| System Metric | Measured Value | Target / Benchmark |
|---|---|---|
| Answer Evaluation Accuracy | 91.4% | ≥ 90% |
| Resume Parsing Accuracy | 89.7% | ≥ 85% |
| Adaptive Difficulty Calibration Accuracy | 91.7% | ≥ 88% |
| Average API Response Latency | 1.24 seconds | < 2 seconds |
| User Satisfaction Score (5-point scale) | 4.61 / 5.00 | ≥ 4.00 |
| Feedback Quality Rating (User-rated) | 94.1% positive | ≥ 90% |
| System Uptime (4-week evaluation) | 99.7% | ≥ 99% |
| Peak Concurrent Users Supported | 350 users | ≥ 300 users |

*B. Discussion*

The results presented in Tables I and II demonstrate statistically and practically significant advantages of the AIPP over conventional preparation methods. The experimental group's mean overall score improvement of 44.3% is more than 2.5 times that observed in the control group (17.7%), strongly

suggesting that the platform's adaptive, personalised, and feedback-rich environment is substantively more effective than unassisted preparation. The particularly large gains in Communication Clarity (+45.4%) and Response Completeness (+45.4%) for AIPP users may be attributed to the multi-dimensional feedback mechanism, which explicitly identifies gaps in these dimensions and provides targeted suggestions for improvement — a feature absent from conventional preparation resources. The system performance metrics in Table II confirm that the platform meets or exceeds all target benchmarks across accuracy, latency, reliability, and user satisfaction dimensions, validating the architectural decisions underlying the implementation.

## 8. Advantages

*A. Advantages*

**Personalisation at Scale:** The resume-aware question generation pipeline produces interview questions uniquely calibrated to each candidate's declared skills and experience, delivering a level of personalisation that is impractical with conventional preparation resources.

**Objective and Consistent Evaluation:** Automated scoring using SBERT semantic similarity and keyword coverage analysis eliminates evaluator subjectivity, ensuring that all candidates are assessed against consistent, reproducible criteria regardless of time or evaluator identity.

**Adaptive Learning Pathway:** The dynamic difficulty adjustment algorithm ensures that candidates are consistently challenged within their zone of proximal development, optimising the learning trajectory and preventing both cognitive underload and overload.

**Immediacy and Accessibility:** Real-time feedback delivery and 24/7 platform availability eliminate the logistical constraints of scheduling human-conducted mock interviews, democratising access to high-quality preparation resources for all students irrespective of geographic or economic circumstances.

**Comprehensive Performance Analytics:** Session-by-session performance tracking and multi-dimensional competency profiling empower candidates to make data-informed decisions about the focus and allocation of their preparation efforts.

**Multi-Domain Coverage:** The platform's question repository and generative AI capabilities span a broad range of interview domains — including Data Structures & Algorithms, Web Development, DBMS, Operating Systems, OOPS, System Design, and HR/Behavioural — enabling comprehensive preparation for diverse employment contexts.

*B. Limitations*

**Dependency on Third-Party AI Services:** The platform's reliance on the OpenAI GPT-4 API introduces an external dependency that subjects system performance to API rate limits, latency variability, and pricing changes that are outside the platform developers' direct control.

**Text-Only Evaluation in Primary Mode:** While voice input is supported, the primary evaluation engine processes text responses. Paralinguistic features that are critically important in in-person interviews — including tone, pace, eye contact, and body language — cannot currently be assessed by the system.

**Reference Answer Quality Dependency:** The accuracy of keyword-based and semantic evaluation is bounded by the quality and comprehensiveness of the reference answers stored in the system. For highly specialised, niche, or rapidly evolving technical domains, reference answer coverage may be insufficient.

**Cold Start Problem for Novel Skills:** For skills or technologies that are newly emergent and not yet represented in the training data of the underlying language models or the platform's question templates, question quality and evaluation accuracy may be reduced.

**Limited Emotional Intelligence Assessment:** The platform does not currently assess confidence markers, anxiety indicators, or emotional regulation — affective dimensions that are increasingly recognised as important determinants of interview success.

## CONCLUSION

This paper has presented the design, implementation, and empirical evaluation of the AI-Powered Interview Preparation Platform (AIPP) — a comprehensive, full-stack web application that addresses the critical and widespread challenge of interview unpreparedness among student populations. Through the integration of resume-aware question generation, adaptive difficulty calibration, semantic similarity-based answer evaluation, and multi-dimensional natural language feedback, the platform delivers a personalised, objective, and scalable interview preparation experience that substantially outperforms conventional preparation methods.

Empirical evaluation across 120 student participants demonstrated statistically significant improvements in all measured performance dimensions, with the experimental group achieving mean overall score gains of 44.3% compared to 17.7% for the control group. System performance benchmarks were met or exceeded across all measured dimensions, confirming the technical soundness and operational viability of the proposed architecture.

The AIPP represents a meaningful contribution to the intersection of artificial intelligence, natural language processing, and educational technology, offering a replicable, extensible, and open-architecture template for AI-powered skill development platforms. The research community is invited to build upon the foundations established herein, extending the platform's capabilities toward voice-based simulation, affective computing, and federated learning-based personalisation in future work.

As the global job market continues to evolve in complexity and competitiveness, intelligent preparation platforms of this nature will play an increasingly vital role in bridging the gap between academic knowledge and professional readiness — ensuring that students from all backgrounds have an equitable opportunity to demonstrate their true capabilities in the high-stakes arena of employment interviews.

## 9. Future Scope

The AIPP, in its current form, represents a foundational implementation of AI-driven interview preparation. Several high-impact extensions are envisioned for subsequent development phases:

*A. Voice-Based Interview Simulation*

Future iterations will incorporate a comprehensive spoken interview mode featuring real-time ASR (Automatic Speech Recognition), prosodic analysis (measuring speaking rate, pitch variation, and pause patterns), and pronunciation assessment. Integration with advanced speech processing models such as Whisper (OpenAI) and acoustic feature extraction frameworks will enable evaluation of spoken communication competencies — a critical yet currently under-addressed dimension of interview performance.

*B. Multimodal Emotion and Confidence Detection*

The integration of computer vision-based emotion recognition — leveraging facial action unit detection through models such as OpenFace or DeepFace, combined with galvanic skin response (GSR) data from compatible wearable devices — will enable the system to provide affective feedback regarding the candidate's projected confidence, anxiety levels, and emotional expressiveness during mock interview sessions.

*C. AI-Powered Interviewer Persona Simulation*

A major planned enhancement involves the development of dynamic, LLM-powered virtual interviewer personas that simulate the questioning styles, follow-up tendencies, and personality profiles of interviewers from specific companies or industry sectors. This will enable candidates to prepare not only for question content but also for the interpersonal dynamics of different interview contexts — a capability currently unavailable in any commercially deployed solution.

*D. Employer-Candidate Matching Integration*

Future work will explore the integration of the AIPP with job portal APIs (e.g., LinkedIn, Naukri) to enable a closed-loop system wherein the platform's assessment of a candidate's readiness is directly linked to personalised job recommendations, creating a seamless pathway from preparation to application.

*E. Federated Learning for Privacy-Preserving Model Improvement*

To address data privacy concerns associated with centralised training on candidate response data, future model updates will employ federated learning techniques that enable model fine-tuning using distributed on-device data without transmitting sensitive personal information to central servers, ensuring GDPR and PDPA compliance.

## REFERENCES

[1] R. Kumar, A. Sharma, and P. Verma, "Automated Interview Evaluation Using Hidden Markov Models for Speech Pattern Analysis," IEEE Transactions on Learning Technologies, vol. 14, no. 2, pp. 178–191, Apr. 2021.

[2] L. Zhang and H. Li, "BERT-Based Semantic Evaluation of Technical Interview Responses: A Framework for Automated Assessment," in Proc. 2022 IEEE International Conference on Artificial Intelligence in Education (AIED), Durham, UK, pp. 245–257, Jul. 2022.

[3] N. Patel and S. Mehta, "Chatbot-Based Interview Preparation: Design, Implementation, and Usability Evaluation," in Proc. 2021 International Conference on Intelligent Systems and Computer Vision (ISCV), Fez, Morocco, pp. 1–8, May 2021.

[4] T. Nguyen, C. Kim, and J. Park, "Deep Reinforcement Learning for Adaptive Educational Content Delivery: A Comparative Study," IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 7, pp. 2891–2904, Jul. 2022.

[5] R. Anand and V. Singh, "Automated Resume Parsing Using Named Entity Recognition and Dependency Parsing: An NLP Approach," in Proc. 2020 IEEE Region 10 Conference (TENCON), Osaka, Japan, pp. 678–683, Nov. 2020

[6] Y. Chen, M. Wang, and F. Liu, "A Comparative Analysis of AI-Powered Interview Platforms: Usability, Transparency, and Pedagogical Effectiveness," Computers & Education: Artificial Intelligence, vol. 3, p. 100072, 2022.
[7] A. Rajput and M. Verma, "Affective Feedback in AI Interview Simulators: The Role of Sentiment Analysis and Confidence Scoring in Learner Improvement," Journal of Educational Technology & Society, vol. 25, no. 1, pp. 102–115, Jan. 2022.
[8] A. Vaswani et al., "Attention Is All You Need," in Advances in Neural Information Processing Systems (NeurIPS), vol. 30, pp. 5998–6008, 2017.
[9] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks," in Proc. 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP), Hong Kong, China, pp. 3982–3992, Nov. 2019.
[10] OpenAI, "GPT-4 Technical Report," arXiv preprint arXiv:2303.08774,2023. [Online]. Available: https://arxiv.org/abs/2303.08774.