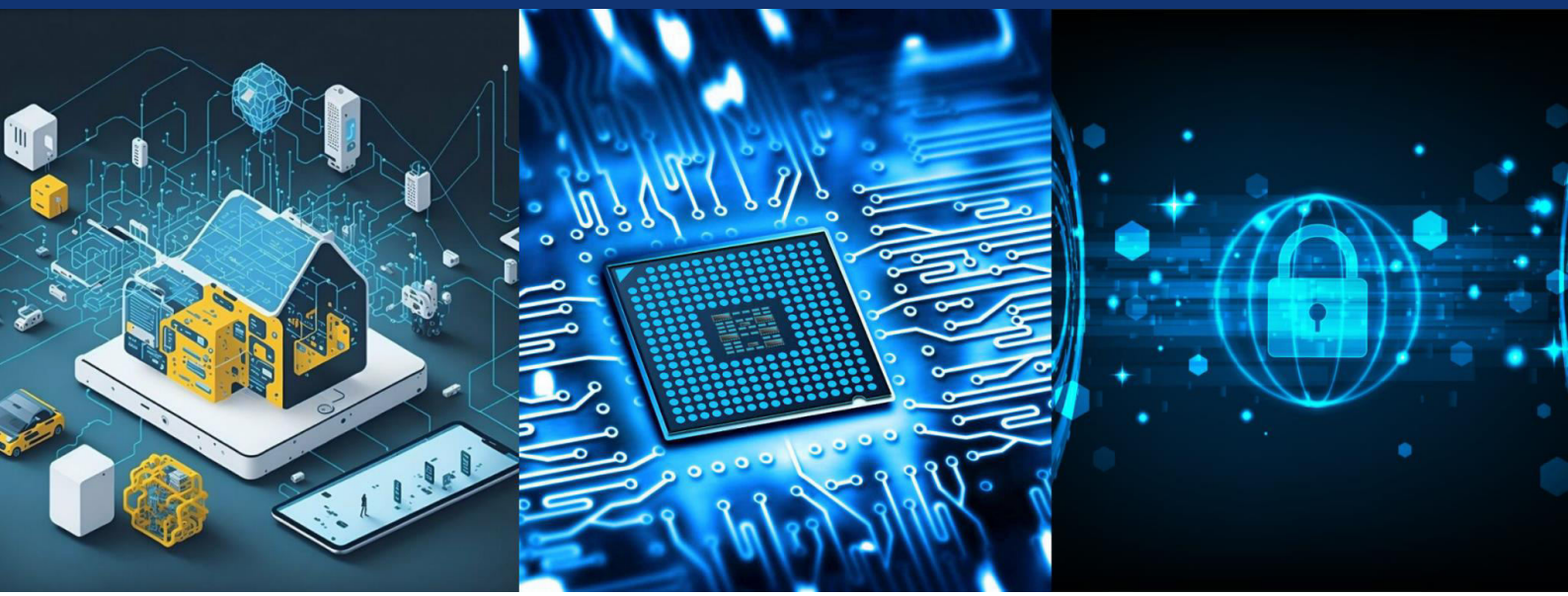
IJIRCCE

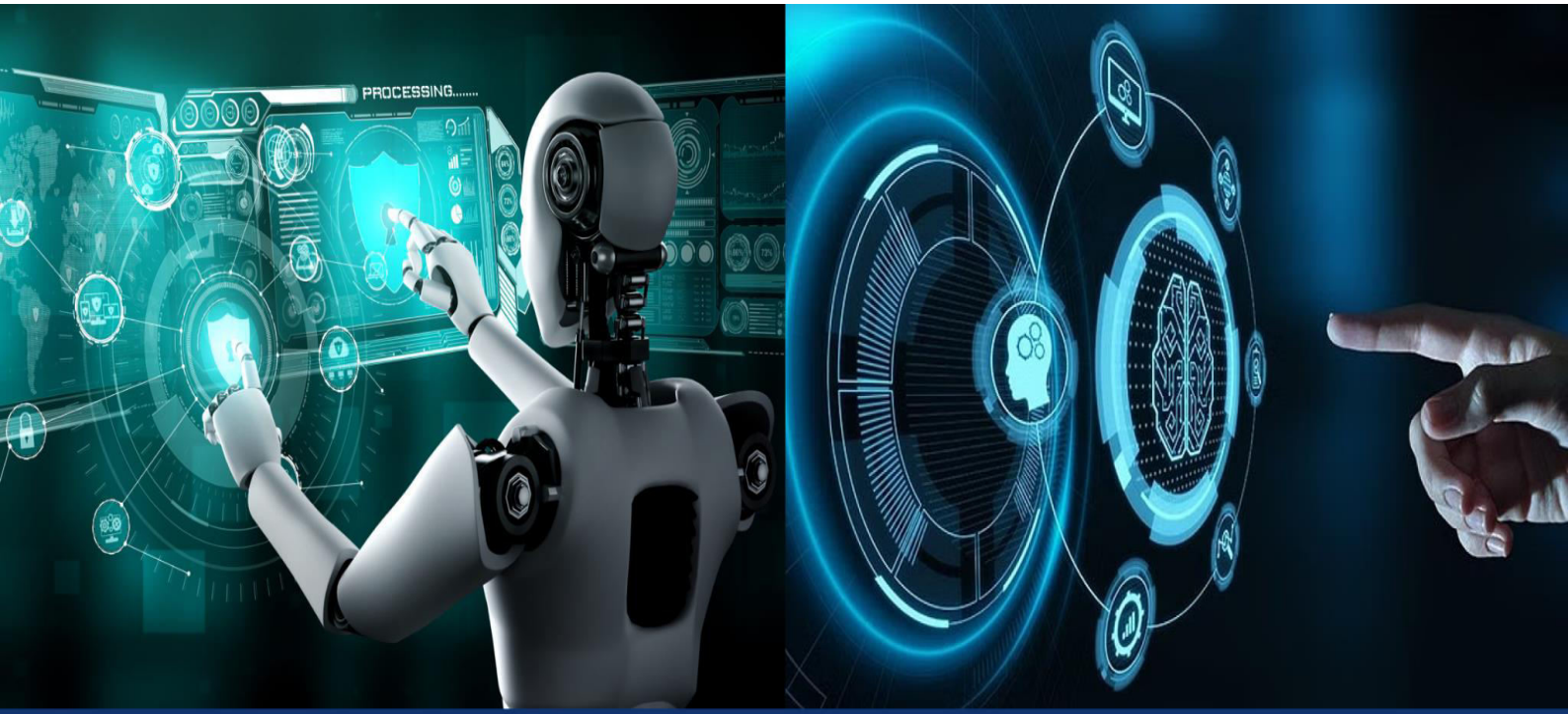
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.927

Volume 14, Issue 10, October 2026

© 2026 IJIRCCE | Volume 14, Issue 10, October 2026 DOI: 10.15680/IJIRCCE.2026.1410020

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Talent Gauge: An AI-Assisted Career-Readiness Platform for Resume Analysis, Job Matching and Mock Interview Practice

Imran Saikalgar[1], Karthik Shankarappa Shivalli[1], Prajwal D N[1], Vivekanand[1],

Er. Rameshkumar H K[2]

B.E. Students, Department of Computer Science & Engineering, STJIT, Ranebennur, Karnataka, India[1]

Assistant Professor, M.Tech, (Ph.D), Department of Computer Science & Engineering, STJIT, Ranebennur, Karnataka, India[2]

ABSTRACT: Job seekers and students frequently submit resumes without knowing whether those documents communicate the skills a target role demands, and the resources they use for interview practice are usually separate from resume feedback. This paper presents Talent Gauge, an AI-assisted web platform that brings resume analysis, job-description matching, skill-gap awareness, mock-interview practice and progress tracking into one workflow. The system extracts text from an uploaded resume, identifies skills with a configurable vocabulary, applies transparent rule-based quality checks, and can compare the resume with a job description using TF-IDF vectors and cosine similarity as a baseline. The mock-interview module presents role-specific behavioural, technical and case-study questions and gives formative feedback on speech (confidence, fluency, filler words) and on facial-expression and emotion indicators. The application uses a React front end, a FastAPI back end and a relational database. Results are presented in dashboards, downloadable reports and an interview history. The platform is intended to guide users and not to guarantee employment or reproduce any employer's applicant tracking system.

KEYWORDS: Resume analysis, Applicant Tracking System, TF-IDF, cosine similarity, mock interview, skill gap, FastAPI, React, emotion analysis.

I. INTRODUCTION

A resume summarises a candidate's education, technical skills, projects, certifications and experience, and recruiters use it to judge relevance to a role. When organisations receive large numbers of applications, digital recruitment tools such as Applicant Tracking Systems (ATS) help store, search and organise them [1][4]. These systems do not share a single scoring formula, and their behaviour depends on how readable a document is and on the criteria an employer configures. A resume that is hard to parse, omits role-related terms or describes projects vaguely may therefore fail to communicate a candidate's strengths.

Students and early-career applicants often lack access to professional resume review or interview coaching. Generic resume checkers tend to return a single score without explaining it, while interview-preparation material is normally disconnected from resume analysis. Talent Gauge addresses this gap with one integrated workflow: resume upload and parsing, skill identification, resume-quality checks, comparison with a job description, skill-gap feedback, interview practice and progress tracking. The system deliberately avoids claiming to replicate any particular employer's ATS; it offers practical, understandable checks and states their limitations.

The main contributions of this work are: (i) an integrated platform combining resume analysis, job matching and interview practice; (ii) an explainable evaluation design that keeps a rule-based checklist separate from a text-similarity estimate; (iii) a mock-interview module that gives real-time speech and expression feedback; and (iv) a modular, web-based implementation with dashboards, reports and history.

© 2026 IJIRCCE | Volume 14, Issue 10, October 2026 DOI: 10.15680/IJIRCCE.2026.1410020

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

II. RELATED WORK

Term weighting and vector-space retrieval, introduced by Salton and Buckley [1] and described comprehensively by Manning et al. [4], remain a dependable and interpretable baseline for comparing documents, and are available in scikit-learn through TfidfVectorizer [5]. Transformer-based language models such as BERT [2] can capture semantic similarity beyond exact word overlap and are a natural extension, while support vector machines have long been used for text categorisation [3].

Several studies address automated interview assessment. Sharma et al. [8] built a mock-interview system that uses NLP to evaluate technical knowledge and communication, but it analyses text only and ignores non-verbal behaviour. Reddy et al. [9] applied convolutional networks and speech processing to facial expressions and voice, improving accuracy at the cost of high computational requirements. Iyer et al. [10] used NLP to evaluate grammar, structure and relevance of answers with real-time feedback, again without non-verbal cues. Mehta et al. [11] used video and audio to estimate confidence, stress and emotional stability, noting that interpreting emotions across different environments is difficult. Taken together, existing work either concentrates on interviews or on documents; few systems link resume analysis with interview preparation in a single, explainable application, which is the gap Talent Gauge targets.

III. PROPOSED SYSTEM

Existing approaches comprise manual review, template-based resume writing, keyword checkers and stand-alone interview resources. Their main limitations are the time needed to tailor a resume to each job description, a fragmented workflow, single scores that do not identify what to fix, dependence on exact keyword matching that misses synonyms and abbreviations, inconsistent scores across tools, and privacy risks when resumes are retained unnecessarily.

Talent Gauge is an integrated web application that accepts a resume and, optionally, a target job description. The proposed workflow is: (1) upload a supported file and optionally supply a job description; (2) parse the file and verify that enough readable text was obtained; (3) detect skills using a vocabulary with aliases; (4) apply documented resume checks and, when a job description is supplied, compute a TF-IDF/cosine similarity; (5) show detected matches, potential gaps and suggestions; (6) offer interview practice for a chosen role and interview type; and (7) present results and keep history only for authenticated users who have a stored record.

IV. SYSTEM ARCHITECTURE

The architecture (Fig. 1) has five layers. The user layer comprises students and job seekers. The React front end provides the dashboard, resume analyser, resume gallery, mock-interview screens and reports. The FastAPI back end handles authentication, interview sessions, resume processing, AI requests and report generation, and acts as the bridge between the interface and the processing modules. The AI/ML layer contains a Resume Analyzer (parsing, keyword matching, ATS-style scoring, suggestions) and a Mock Interview AI (speech-to-text, response analysis, emotion detection, eye tracking, confidence and stress estimation). The database layer stores user information, resume records, interview data, reports and scores. Optional external services include speech-recognition and NLP APIs, e-mail and cloud storage.

© 2026 IJIRCCE | Volume 14, Issue 10, October 2026 DOI: 10.15680/IJIRCCE.2026.1410020

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

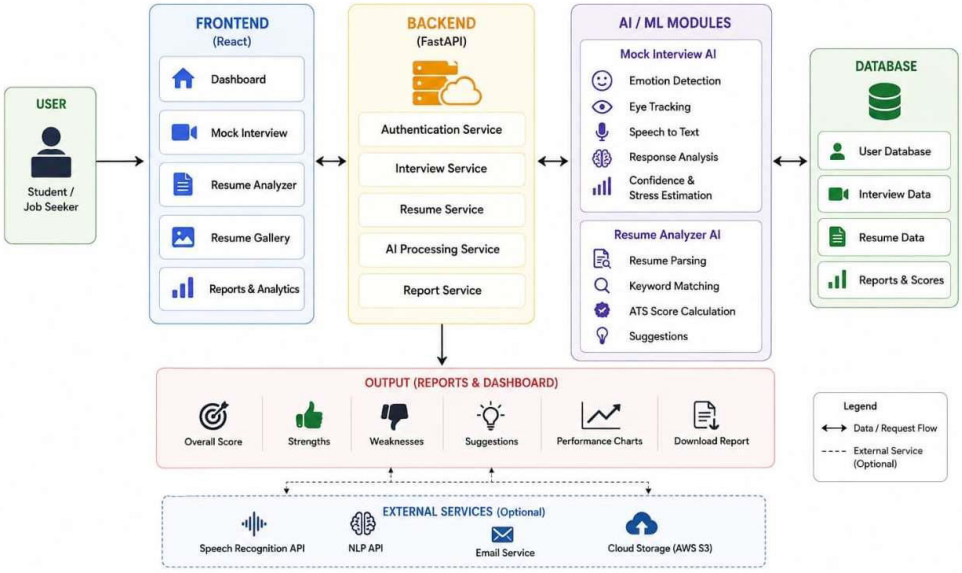


Fig. 1: System architecture of Talent Gauge

A request flows from the browser (over HTTPS) to the React front end, then to the FastAPI back end, which routes it to the resume-analysis or interview-analysis module, reads or writes the database where required, and returns results for presentation. Separating client interaction, application logic, data storage and AI processing keeps the parsing, matching and presentation components independently maintainable and allows future scaling.

V. METHODOLOGY AND IMPLEMENTATION

A. Resume parsing and preprocessing

The user uploads a PDF or DOCX file. The back end validates the file type, extracts the text and checks that the output is not empty; unsupported, empty or corrupted files produce a clear message instead of an analysis. Scanned image-only PDFs need OCR, which is not assumed. Extracted text is normalised (case folding, whitespace and punctuation clean-up) while meaningful technical tokens and section headings are preserved.

B. Skill identification and resume checks

Skills are detected by matching tokens and phrases against a configured vocabulary with aliases that map to canonical names (for example, a short form of a framework name resolves to the full name). The system then checks for key sections such as Education, Skills and Experience, looks for dates that indicate experience duration, and reports each finding together with a suggestion, so that every deduction is explainable.

C. Job-description matching

When a job description is provided, both documents are converted to TF-IDF vectors. The weight of term t in document d is $w(t,d) = tf(t,d) \times \log(N / df(t))$, and the similarity between resume vector r and job vector j is the cosine similarity:

$$\cos(r, j) = (r \cdot j) / (\|r\| \|j\|)$$

The result is reported as a similarity estimate, separate from the checklist, together with matched terms and terms not found. Because exact matching can miss synonyms, the output reminds users to verify the findings; a semantic embedding model [2] is identified as future work.

D. ATS-style scoring

The analyser reports an overall ATS-style score with five component metrics: keywords, formatting, readability, experience relevance and extras. It also lists missing sections, keyword coverage (found, missing critical, missing nice-

© 2026 IJIRCCE | Volume 14, Issue 10, October 2026 DOI: 10.15680/IJIRCCE.2026.1410020

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

to-have) and readability indicators such as Flesch reading score and grammar issues. The score is labelled as an estimate based on general ATS practice and not as any employer's actual score.

E. Mock-interview module

Users choose a job role and an interview type (behavioural, technical or case study), or a five-minute quick-practice mode. Questions are fetched from a question bank and displayed one at a time with tips such as the STAR method. While the user answers, the browser captures audio and video; speech is transcribed live and analysed for confidence, fluency and filler words, while an in-browser face-analysis model estimates emotional indicators (joy, confidence, focus, stress, engagement, frustration, surprise and boredom) against a baseline calibrated at the start of the session. The back end combines the available results to score each answer; after the last question the system calculates an overall score and generates feedback.

F. Dashboard, reports and history

Authenticated users can view a dashboard of resume and interview results, download a PDF interview report, and browse an interview history with filters by interview type and analytics such as average score, minutes practised, total sessions and improvement. A profile page stores skills, experience level and target roles to personalise practice. A resume gallery of sample resumes organised by category and experience level offers inspiration, and a learning hub lists courses with enrolment and progress tracking.

VI. RESULTS AND DISCUSSION

The implemented application was run end to end; representative outputs are shown below. Fig. 2 shows the ATS Resume Analyzer for a sample resume: the overall score is accompanied by component metrics (keywords 50 %, formatting 60 %, readability 0 %, experience relevance 50 % and extras 0 %) and by recommendations such as adding clear Education, Skills and Experience headers. The explanation of each component lets the user see why the score is low, which a single number would not.

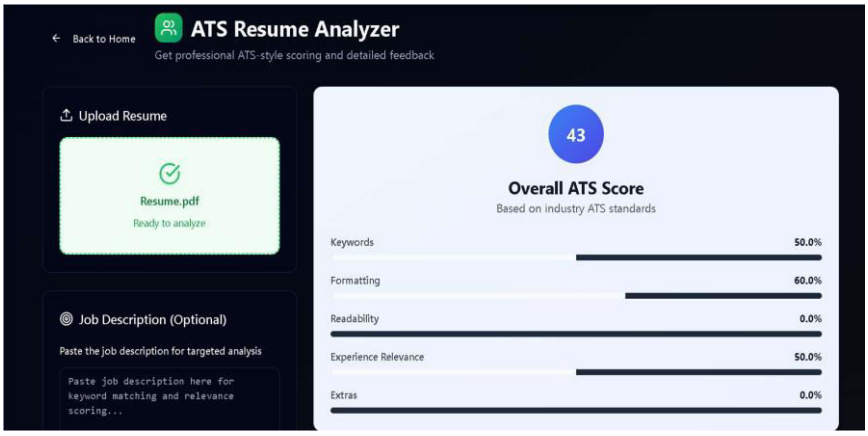


Fig. 2: ATS Resume Analyzer with overall score and component metrics

© 2026 IJIRCCE | Volume 14, Issue 10, October 2026 DOI: 10.15680/IJIRCCE.2026.1410020

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Fig. 3 shows the live interview screen with the question, timer, real-time feedback, emotional-state indicators and live speech analysis. Fig. 4 shows the post-session report with the confidence-over-time curve and emotion distribution, followed by strengths and areas for improvement.

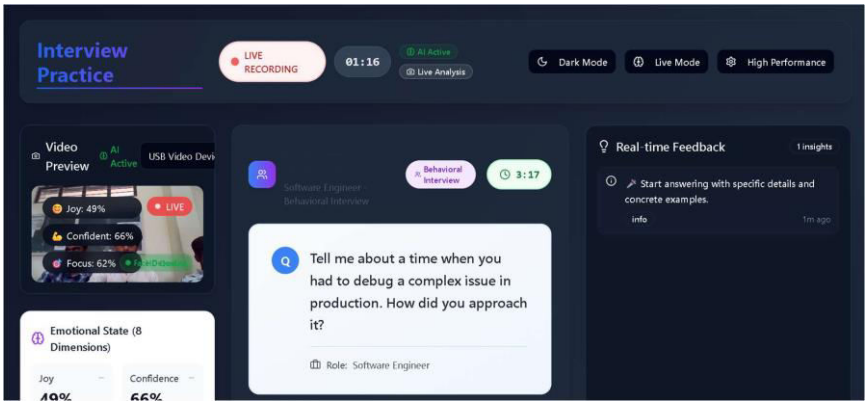


Fig. 3: Mock-interview session with live feedback



Fig. 4: Interview report showing confidence trend, emotion distribution, strengths and improvement areas

Table I summarises a sample session. These values are illustrative outputs of one run and should not be read as accuracy or performance claims for the underlying models.

Output	Sample value	Source module
Overall ATS-style score	43 / 100	Resume Analyzer
Overall interview performance	76 / 100	Mock Interview AI
Communication / Clarity	4.4 / 5 and 4.3 / 5	Speech and NLP analysis
Confidence / Engagement	3.0 / 5 and 3.5 / 5	Speech and expression analysis
Emotion distribution	Focused 33 %, Confident 28 %, Joyful 24 %, Stressed 14 %	Face / emotion analysis

Table I: Sample outputs from one analysis run

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

VII. TESTING

A test plan of fifteen cases covers the principal workflows: valid PDF and DOCX upload, unsupported, corrupted or empty files, submission without a file, valid and blank job descriptions, alias detection, omitted skill terms, changed job descriptions, repeated identical inputs (deterministic output), long documents, opening the interview page, access control between users, and the distinction between checklist and similarity in the score explanation. Execution results for these cases are still to be recorded; the authors will report observed pass and fail outcomes, and any trained model will be evaluated with suitable metrics on a defined test set before accuracy is claimed.

VIII. LIMITATIONS AND ETHICAL CONSIDERATIONS

Skill vocabularies need continuing maintenance and job descriptions use varied terminology, so similarity depends on preprocessing. Image-only PDFs may not be parsed. Facial-expression and emotion indicators are approximate and may be affected by lighting, camera quality and cultural differences; they should be treated as practice feedback and not as a judgement of the candidate. Resumes contain personal data, so uploaded documents must be protected from unauthorised access, retained only where necessary and transmitted over encrypted connections, and the analysis should concentrate on job-related content and not on protected personal attributes. Talent Gauge is a career-support tool, not a hiring decision-maker.

IX. CONCLUSION AND FUTURE SCOPE

Talent Gauge integrates resume parsing, skill identification, job-description matching, ATS-style scoring, mock-interview practice and progress tracking in a single web application built on React and FastAPI. By keeping the rule-based checklist distinct from the similarity estimate and by showing which terms and checks contribute to each result, it gives students understandable and actionable feedback. Future work includes completing and publishing the systematic test results, adding OCR for scanned resumes, using semantic embeddings for synonym-aware matching, evaluating the speech and emotion models on labelled data, and adding secure export and data-retention controls.

X. ACKNOWLEDGEMENT

The authors thank the Department of Computer Science & Engineering, STJIT, Ranebennur, for the facilities and guidance provided during this project.

REFERENCES

[1] G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” Information Processing & Management, vol. 24, no. 5, pp. 513–523, 1988.
[2] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. NAACL-HLT, vol. 1, pp. 4171–4186, 2019.
[3] T. Joachims, “Text categorization with support vector machines: Learning with many relevant features,” in Proc. European Conf. on Machine Learning (ECML), 1998.
[4] C. D. Manning, P. Raghavan and H. Schütze, Introduction to Information Retrieval. Cambridge, U.K.: Cambridge University Press, 2008.
[5] Scikit-learn Developers, “TfidfVectorizer,” scikit-learn documentation. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
[6] FastAPI, “FastAPI Documentation.” [Online]. Available: https://fastapi.tiangolo.com/
[7] Meta Platforms, “React Documentation.” [Online]. Available: https://react.dev/
[8] R. Sharma, P. Verma, S. Gupta and A. Singh, “AI-based mock interview system for skill assessment,” IEEE Int. Conf. on Artificial Intelligence and Applications, 2023.
[9] K. Reddy, M. Kumar, S. Rao and V. Patel, “Automated interview evaluation using deep learning,” Int. J. of Computer Science and Engineering, 2022.
[10] S. Iyer, R. Nair and P. Deshmukh, “Natural language processing for interview response analysis,” Springer Int. Conf. on Data Science, 2021.
[11] T. Mehta, R. Joshi and K. Shah, “AI-based personality and behavior analysis system,” Elsevier J. of Artificial Intelligence Research, 2020.



INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details