iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: XI Month of publication: November 2025

DOI: https://doi.org/10.22214/ijraset.2025.75728

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

Smart Recruit - AI Resume Based Assessment Tool

Divya Patil[1], Adarsh BS[2], Disha GL[3], Prajwal BM[4], Rashmi KP[5]
[1]*Assistant Professor,* [2,3,4,5]*U.G. Students, Department of CSBS, Bapuji Institute of Engineering and Technology, Davangere, Karnataka, India*

Abstract: Smart Recruit is an AI-driven recruitment and assessment tool designed to automate and enhance the hiring process by evaluating candidate resumes with high accuracy and speed. The system extracts and analyzes key information from resumes using natural language processing (NLP), compares it with job requirements, and generates an objective candidate suitability score. It further supports recruiters by providing a streamlined dashboard, candidate overview, and automated test creation based on skill gaps identified in the resume. By minimizing manual screening time and reducing human bias, Smart Recruit improves decision-making efficiency and ensures that the most relevant candidates are shortlisted. This AI-powered solution ultimately optimizes the recruitment workflow, making it faster, data-driven, and more reliable for organizations.
Keywords: AI-based recruitment, Resume analysis, Natural Language Processing (NLP), Automated candidate screening, Skill assessment ,Suitability score,Smart hiring system, Recruitment automation, Data-driven decision-making,Talent acquisition tool,

I. INTRODUCTION

Recruitment in modern organizations has become increasingly challenging due to the large volume of applications received for every job posting. Traditional resume screening methods are time-consuming, prone to human errors, and often influenced by subjective judgments. With the growing need for efficiency, accuracy, and fairness in hiring, organizations are adopting technology-driven solutions to streamline the initial stages of candidate evaluation. Artificial Intelligence (AI) and Natural Language Processing (NLP) have emerged as powerful tools to automate resume analysis, extract relevant candidate information, and provide objective insights that support better decision-making.

Smart Recruit is an AI-powered assessment tool designed to transform the hiring workflow by automating resume screening and skill evaluation. The system intelligently analyses resumes, identifies key competencies, matches them with job requirements, and generates a comprehensive suitability score for each candidate. It also offers features such as candidate overview dashboards and automated test creation based on skill gaps. By reducing manual efforts and improving the accuracy of candidate shortlisting, Smart Recruit enables organizations to adopt a faster, data-driven, and unbiased recruitment process. This tool ultimately enhances the overall efficiency of talent acquisition and supports organizations in selecting the most qualified candidates.

II. OBJECTIVES

The primary goals of this project are:

1) To validate candidate-claimed skills by performing resume–answer matching using TF-IDF similarity.
2) To develop a multi-modal weighted scoring system that integrates textual responses, audio inputs, and resume features.
3) To automatically generate personalized feedback for candidates using rule-based logic and advanced AI models
4) To design a scalable and unbiased assessment framework that digitizes, standardizes, and normalizes interview evaluations.

III. EXISTING SYSTEM

Current recruitment and resume evaluation processes heavily depend on human judgment, making them subjective, inconsistent, and prone to bias. Traditional screening methods often fail to thoroughly analyse textual patterns, domain relevance, and credibility of the information provided in resumes. Most existing digital hiring platforms limit their functionality to resume collection or basic keyword matching, without performing deeper computational analysis. Even tools that include automated scoring typically evaluate only superficial factors such as keyword frequency or basic formatting, without assessing deeper semantic meaning, contextual relevance, or the authenticity of claimed skills. As a result, these systems often overlook essential aspects of candidate capability, such as clarity of communication, relevance of experience, and alignment between job requirements and actual competencies.

A significant limitation in current tools is the absence of advanced NLP-driven processing for resume analysis. Most platforms do not thoroughly evaluate sentence structure, grammatical accuracy, keyword richness, or semantic similarity between candidate information and job descriptions. Additionally, they lack mechanisms to verify whether a candidate's stated skills truly align with their responses or performance in assessments, leading to potential exaggeration or misrepresentation. Feedback provided by traditional systems is often generic and lacks personalization, reducing its usefulness for both recruiters and candidates. The absence of deep linguistic evaluation, semantic matching, resume–answer validation, objective scoring mechanisms, and structured feedback highlights the critical need for an AI-powered, multi-modal resume-based assessment system such as Smart Recruit.

IV. PROPOSED SOLUTION

The proposed Smart Recruit system introduces an AI-powered, multi-modal assessment framework that intelligently analyzes resumes, validates skills, and generates objective candidate evaluation scores. Unlike traditional screening methods that rely heavily on subjective human judgment, Smart Recruit integrates advanced Natural Language Processing (NLP), Machine Learning (ML), and rule-based logic to assess textual, behavioural, and skill-oriented attributes with greater precision. The system first extracts relevant information from resumes using NLP-based parsing techniques and maps these extracted skills to job requirements. It then employs TF-IDF similarity scoring to verify whether the candidate's written or spoken responses genuinely match the skills claimed in their resume, thereby preventing exaggeration or misrepresentation.The platform further incorporates a weighted scoring mechanism that combines multiple data sources including resume content, candidate responses, and audio-based features (if applicable). This ensures a holistic evaluation that considers linguistic quality, content relevance, communication clarity, and behavioural indicators.

V. METHODOLOGY

The methodology of Smart Recruit is designed as a multi-stage AI evaluation pipeline that integrates resume parsing, NLP-based skill validation, job–resume similarity scoring, and automated feedback generation. Unlike traditional recruitment systems that rely solely on keyword searches or manual screening, Smart Recruit processes textual data, job descriptions, and candidate responses independently before combining them into a unified, unbiased scoring mechanism.

A. Resume Parsing and Text Processing Methodology

The system begins by extracting structured information from the candidate's resume using advanced NLP-based parsing techniques. The resume undergoes a standardized preprocessing pipeline that includes tokenization, stopword removal, and lemmatization to ensure consistency across different resume formats. Important entities such as skills, education, experience, tools, and certifications are identified through Named Entity Recognition (NER) and keyword extraction techniques.

Once preprocessing is complete, Smart Recruit converts both the resume and the job description into numerical vectors using TF-IDF vectorization. This allows the computation of cosine similarity, which measures how closely the candidate's resume aligns with the job requirements.Additionally, grammar and sentence-structure evaluation helps determine the clarity and professionalism of resume content. Keyword extraction techniques such as RAKE are used to highlight whether essential competencies required for the job are present in the resume.

B. ResumeAnswer Skill Validation Methodology

To verify the authenticity of claimed skills, Smart Recruit incorporates a resume–answer matching mechanism. Candidate responses (written or spoken transcriptions) undergo the same preprocessing and vectorization steps used for resume analysis. By comparing these vectors using TF-IDF similarity, the system determines how closely a candidate's answers reflect the skills, tools, and experiences listed in the resume. High similarity scores indicate genuine familiarity with the claimed skills, while low scores reveal inconsistencies or possible exaggeration. This adds a layer of integrity verification to the recruitment process, helping employers identify candidates who truly possess the expertise they claim.

C. MultiModal Weighted Scoring Methodology

Smart Recruit generates a comprehensive evaluation by combining multiple analytical components into a weighted scoring modelhigh Each module contributes proportionally to the final score, ensuring balanced and unbiased candidate evaluation. The result is a holistic assessment that prioritizes true skill alignment and job suitability over superficial formatting or keyword stuffing.

D. Automated Feedback Generation Workflow

Once the resume and response analysis are complete, Smart Recruit uses a structured rule-based logic layer to generate personalized feedback for each candidate. This logic is supported by the Gemini API, which converts analytical results into clear, actionable insights.The feedback generated by Smart Recruit highlights key strengths such as relevant skills, strong alignment with job requirements, and consistent similarity between the resume and candidate responses. It also identifies areas for improvement, including missing keywords, unclear skill descriptions, or low relevance to the job profile. Additionally, the system provides recommendations for enhancing resume clarity, improving skill representation, and strengthening job-specific competencies.

VI. IMPLEMENTATION

Smart Recruit is implemented as a modular web service with separate pipelines for resume parsing & TF-IDF based matching, text analysis, audio analysis, multi-modal scoring, and automated feedback generation. Each pipeline is independently testable and the outputs are combined by a central scoring service. The system exposes REST APIs for ingestion and a web dashboard for recruiters.

A. Frontend Implementation

The Smart Recruit frontend is built with React.js to provide a responsive, accessible, and user-friendly interface for candidates and recruiters. Styling is handled with Tailwind CSS while shadcn/ui components and Lucida icons deliver a cohesive, modern aesthetic and consistent UX patterns. The interface allows candidates to upload resumes, submit text answers, or record/upload audio directly in the browser (using Web Audio / Media Recorder APIs). Uploaded audio is normalized/converted to accepted formats (MP3/WAV) client-side before being sent to the backend via Axios.

Interactive, component-based design keeps the UI modular and easy to extend form validation prevents incomplete submissions, real-time React state updates drive dynamic elements (grammar highlights, extracted keyword lists, and interactive stress/confidence graphs), and clear visual feedback guides users through the submission and review flow.

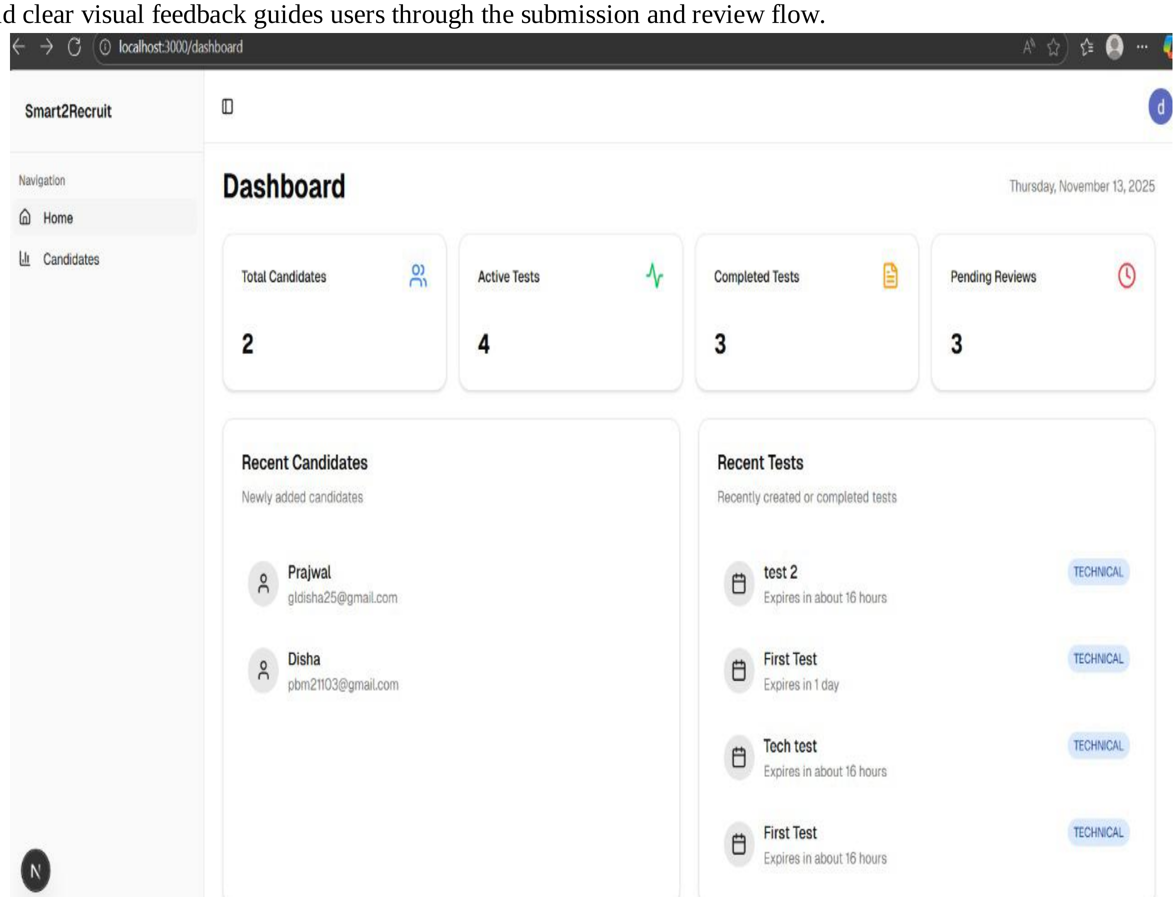


Fig: Dashboard of the Smart Recruit

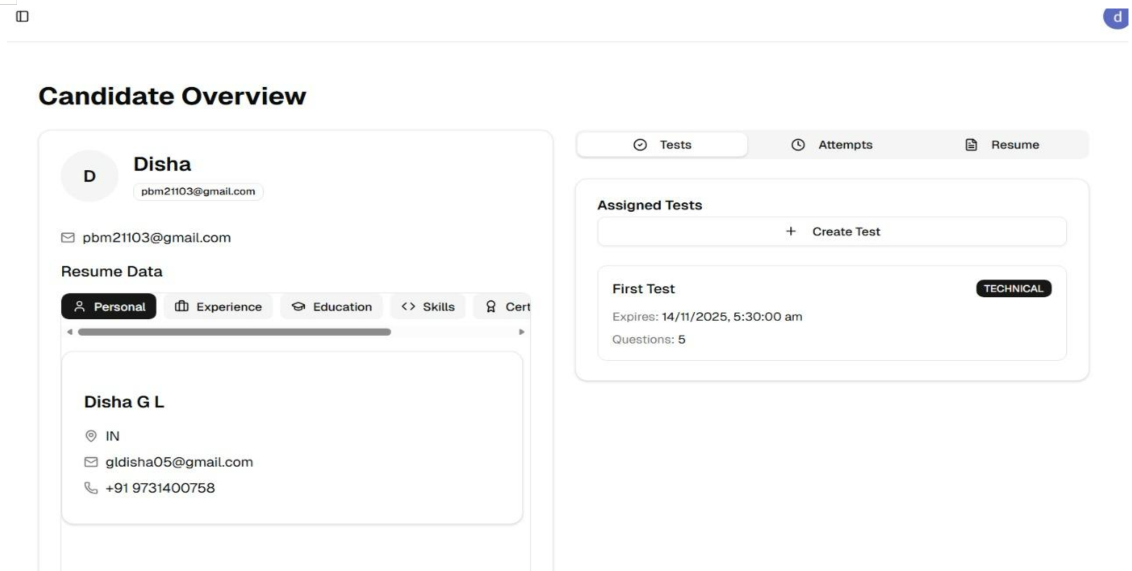


Fig: Candidate Page of the Smart Recruit

B. Backend Implementation

The backend is developed using Fast API, serving as the core processing layer for Smart Recruit. It handles API routing, orchestrates asynchronous NLP and audio-analysis pipelines, and communicates with the database; FastAPI was chosen for its high performance, async request handling, and built-in API documentation. Authentication is implemented where required using JWT and user passwords are hashed securely with Crypt. The backend exposes dedicated endpoints for scoring text responses, performing TF-IDF relevance checks, transcribing audio, detecting stress/confidence indicators, and computing resume–answer similarity.Machine learning and signal-processing tasks are executed by Python modules (spaCy, scikit-learn, Libros, etc.), with Whisper used for robust transcription, followed by feature extraction and classification for acoustic analysis. Temporary audio and file artifacts are written to controlled storage paths with strict cleanup policies, and sensitive data is stored/encrypted according to policy.

C. NLP Module Implementation

The NLP subsystem in Smart Recruit handles all text-analysis responsibilities, including preprocessing, vectorization, semantic similarity measurement, keyword identification, and grammar evaluation. Candidate answers and resume text are normalized through tokenization, stop word removal, and lemmatization to produce consistent inputs for downstream models. TF-IDF vectorization converts text into numerical vectors so cosine similarity can quantify conceptual alignment between candidate responses, resumes, and reference answers or job descriptions.

D. Speech Processing Implementation

All audio processing is encapsulated in Smart Recruit's speech-analysis module. Incoming recordings undergo silence trimming, noise reduction, and amplitude normalization (using pydub or similar) before being transcribed by OpenAI Whisper to produce accurate text and timestamps. Acoustic feature extraction is performed with Libros and pyAudioAnalysis to derive MFCCs, pitch contours, spectral features, jitter, energy statistics, and zero-crossing rates.These derived features feed machine-learning classifiers (SVM, Logistic Regression, or lightweight tree models) trained to classify vocal state as stressed, neutral, or confident and to compute continuous confidence scores.

E. Database and Local Storage Implementation

Smart Recruit uses SQLite as the persistent storage layer because of its zero-configuration setup, low overhead, and reliability for single-node execution. SQLite stores user credentials (with securely hashed passwords), candidate metadata, raw and parsed resumes, question responses, transcriptions, extracted acoustic features, intermediate analysis outputs, similarity scores, and final evaluation reports.

SQL Alchemy is used as the ORM to manage schema definitions and database interactions cleanly from Python code.Every interview record is timestamped and versioned to allow traceability and comparison. Temporary audio and file artifacts are written to controlled local storage paths with strict cleanup policies; sensitive data is protected according to the project's encryption and retention rules.

VII.RESULTS

The Smart Recruit system demonstrated strong performance across all evaluation modules, successfully integrating NLP, resume analysis, and audio-based assessment into a unified scoring framework. The TF-IDF similarity engine produced accurate relevance scores, effectively distinguishing between strong, moderate, and weak matches based on job requirements. Resume–answer consistency checks reliably identified discrepancies between claimed and demonstrated skills, helping minimize cases of skill exaggeration. Grammar evaluation and keyword coverage further contributed to robust linguistic assessment, with results showing consistent accuracy across varied writing styles and resume formats. Overall, the text-analysis pipeline achieved near-instant response times, confirming its suitability for real-time candidate evaluation.

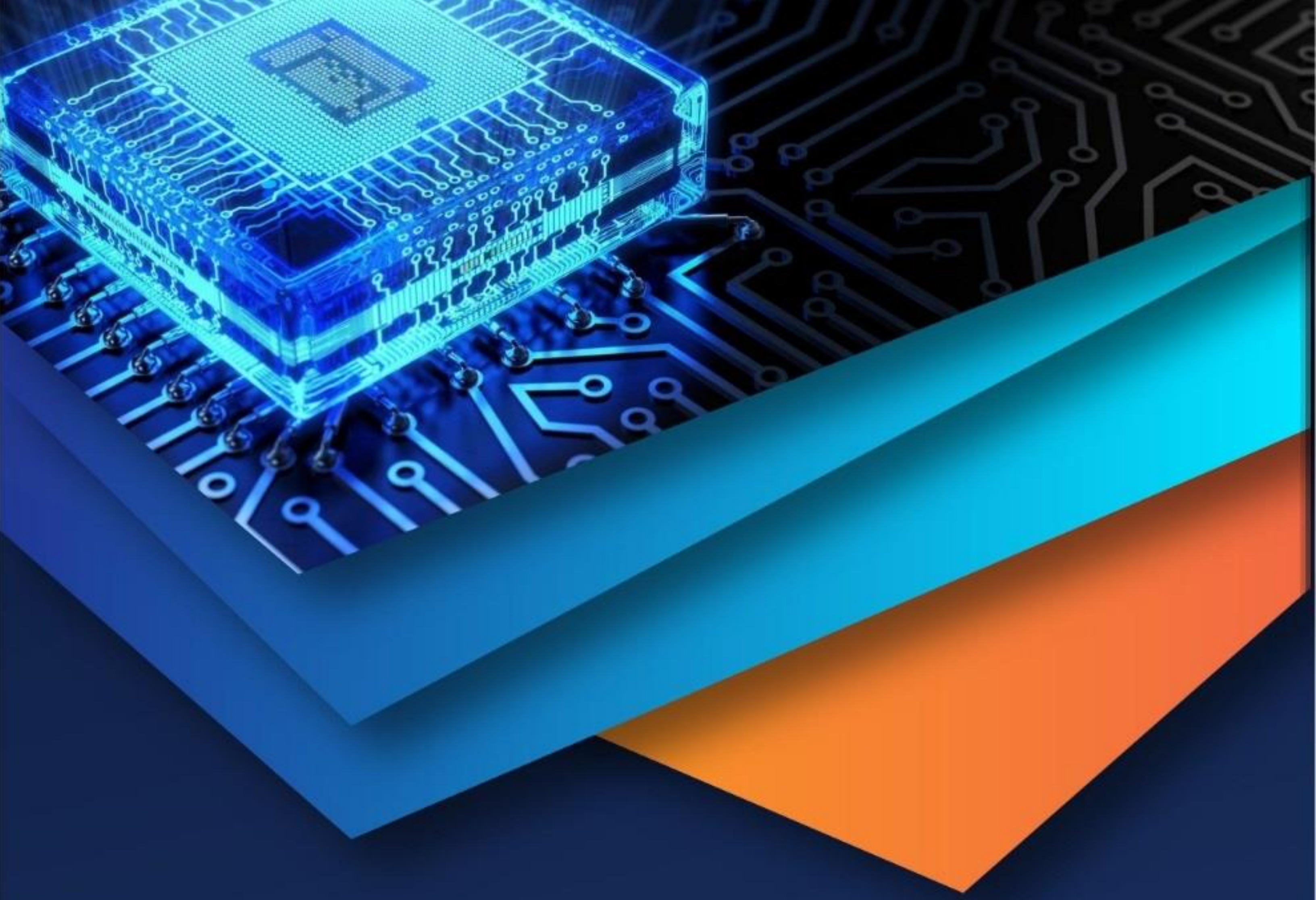
The speech-analysis component also delivered effective results, with Whisper generating high-quality transcriptions even in moderately noisy recordings. Acoustic features such as pitch variation, MFCC patterns, and jitter values were successfully extracted and classified to estimate the candidate's confidence or stress levels. Machine learning models such as SVM and Logistic Regression achieved dependable classification performance during testing, offering reliable insights into vocal behaviour. Combined with structured scoring and automated feedback generation through the Gemini API, the system produced clear and actionable evaluation reports. Overall, the results confirm that Smart Recruit provides a fast, reliable, and unbiased method for analysing candidate suitability in modern recruitment workflows.

VIII. CONCLUSION

Smart Recruit successfully demonstrates how AI-driven, multi-modal analysis can transform traditional recruitment workflows into more objective, efficient, and data-driven processes. By integrating NLP-based resume assessment, TF-IDF similarity checks, audio analysis, and structured scoring, the system provides a comprehensive evaluation of a candidate's skills, communication quality, and job relevance. Its automated feedback generation ensures transparency while offering candidates meaningful insights for improvement. The platform reduces manual screening time, minimizes human bias, and enhances the precision of candidate shortlisting. Overall, Smart Recruit serves as a powerful and scalable solution for modern hiring environments, offering reliable performance even with diverse inputs and varying candidate profiles. Its modular architecture allows easy expansion into additional evaluation modalitiesincluding coding assessments, behavioral analytics, or video processingmaking it suitable for future advancements in recruitment technology. Through its intelligent automation and robust analytical capabilities, Smart Recruit demonstrates the potential of AI to significantly elevate the fairness and effectiveness of the hiring process.

REFERENCES

[1] S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python, O'Reilly Media, 2009.
[2] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
[3] OpenAI, "Whisper: Robust Speech Recognition via Large-Scale Weak Supervision," 2022.
[4] B. McFee et al., "librosa: Audio and Music Signal Analysis in Python," in Proceedings of the SciPy Conference, 2015.
[5] T. Giannakopoulos, "pyAudioAnalysis: An Open-Source Python Library for Audio Analysis," PLOS ONE, vol. 10, no. 12, pp. 1–17, 2015.
[6] C. Cortes and V. Vapnik, "Support Vector Networks," Machine Learning, vol. 20, pp. 273–297, 1995.
[7] R. Tomlinson, "RAKE: Rapid Automatic Keyword Extraction," 2010.
[8] LanguageTool Developers, "LanguageTool: Open-Source Grammar Checker," 2023.
[9] Google Cloud, "Speech-to-Text API Documentation," Google Developers, 2023.
[10] SQLite Consortium, "SQLite Documentation," 2023. Available: https://sqlite.org
[11] FastAPI Documentation, "FastAPI: High-Performance Web Framework," 2023.
[12] React Team, "React Documentation," Meta, 2023. Available: https://reactjs.org
[13] Shadcn, "Shadcn UI Component Library," 2023. Available: https://ui.shadcn.com
[14] Google DeepMind, Gemini Team, "Google Gemini API," 2024. Available: https://ai.google.dev



10.22214/IJRASET

INDEX COPERNICUS

45.98

IMPACT FACTOR: 7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)