IJVAST | Vol. 2, Issue 1 | May 2026 | Open Access — Peer Reviewed Review Article

Technical and Hiring Assistance System

Abhishek Ashtekar[1], Amol Navale[2]

Department of Computer Application

Agasti Institute of Management, Computer Application and Research (AIMCAR), Akole, Maharashtra, India

Abstract—Technical recruitment processes commonly rely on manual resume screening and separately administered technical assessments, resulting in inconsistent candidate evaluation, delayed shortlisting, and significant recruiter workload, particularly for organizations and campus placement cells handling high applicant volumes. This paper presents the design and development of a Technical and Hiring Assistance System (THAS) that integrates automated resume parsing, skill-based candidate-job matching, and an online technical assessment module into a unified recruitment support platform. The system extracts structured candidate information such as skills, qualifications, and experience from uploaded resumes using natural language processing techniques, ranks candidates against job-specific skill requirements using a weighted keyword and semantic similarity scoring approach, and administers configurable technical quizzes and coding assessments with automated scoring. The system was evaluated on a dataset of resumes and job postings, and through a pilot deployment with a campus placement cell, measuring resume-parsing accuracy, candidate-ranking relevance, assessment auto-grading accuracy, and recruiter time savings. Results show resume-parsing field-extraction accuracy of 91.4%, candidate-ranking relevance in agreement with recruiter judgment for 86% of top-ranked candidates, and an approximately 65% reduction in recruiter time spent on initial shortlisting compared to fully manual screening. These findings indicate that an integrated resume-parsing, candidate-matching, and automated-assessment platform can meaningfully improve the efficiency and consistency of technical hiring workflows.

Index Terms—*Hiring Assistance System, Resume Parsing, Natural Language Processing, Candidate-Job Matching, Automated Technical Assessment, Recruitment Automation, Skill Extraction, E-Recruitment.*

I. INTRODUCTION

Technical recruitment, particularly at scale in campus placement drives and high-volume hiring cycles, places substantial manual burden on recruiters who must review large numbers of resumes and separately coordinate technical assessments to evaluate candidate suitability. Manual resume screening is time-consuming, subject to reviewer inconsistency, and offers limited scalability as applicant volume grows, while disconnected technical assessment tools require candidates and recruiters to navigate multiple separate systems.

Automating key stages of the recruitment pipeline, specifically resume information extraction, candidate-job skill matching, and technical assessment administration, can substantially reduce recruiter workload while improving the consistency and speed of initial candidate shortlisting. Natural language processing techniques for resume parsing and semantic similarity scoring for candidate-job matching have matured sufficiently to support practical, deployable recruitment-assistance tools.

This paper presents the design, development, and evaluation of a Technical and Hiring Assistance System (THAS) that integrates automated resume

C. Online Technical Assessment Platforms

Existing online technical assessment platforms typically support multiple-choice technical quizzes and, in more advanced implementations, automated code-execution-based scoring for programming assessments, with auto-grading accuracy dependent on the robustness of test-case design and code-execution sandboxing.

III. SYSTEM DESIGN

A. System Architecture

THAS was designed using a modular architecture comprising four primary components: a resume-parsing module, a candidate-job matching and ranking module, a technical assessment module, and a recruiter dashboard, all built atop a centralized relational database storing candidate profiles, job postings, and assessment results.

B. Resume Parsing Module

Uploaded resumes (PDF or DOCX format) are processed through a text-extraction pipeline followed by a named-entity recognition model fine-tuned to identify candidate name, contact details, educational parsing, skill-based candidate ranking, and configurable online technical assessments within a single platform, intended to support recruiters and campus placement cells in streamlining technical hiring workflows.

II. LITERATURE REVIEW

A. Automated Resume Parsing

Prior work on resume parsing has explored rule-based, statistical, and neural named-entity recognition approaches for extracting structured fields such as skills, education, and work experience from unstructured resume text, with more recent approaches leveraging pretrained language models to improve extraction accuracy across varied resume formats.

B. Candidate-Job Matching Techniques

Candidate-job matching approaches in the literature range from simple keyword-overlap scoring to more sophisticated semantic similarity methods using word or sentence embeddings, generally reporting improved alignment with recruiter judgment when semantic similarity is combined with structured skill-requirement matching rather than keyword matching alone.

qualifications, technical skills, and work experience, populating a structured candidate profile record.

C. Candidate-Job Matching Module

Job postings are defined with a structured list of required and preferred skills. Candidate profiles are scored against each job posting using a weighted combination of exact skill-keyword matches and semantic similarity between extracted candidate skills and job-required skills computed using sentence embeddings, producing a ranked candidate shortlist for each posting.

D. Technical Assessment Module

The assessment module supports configurable multiple-choice technical quizzes with automated scoring, and a coding-assessment component that executes submitted code against predefined test cases in a sandboxed execution environment, automatically computing a pass-rate-based score for each submission.

An overview of the system's core modules and their functions is presented in Table I.

TABLE I — Core Modules of the Technical and Hiring Assistance System

Module	Primary Function	Key Technique Used
Resume Parsing	Extract structured candidate data from resumes	Transformer-based named-entity recognition
Candidate-Job Matching	Rank candidates against job requirements	Keyword matching + semantic similarity (sentence embeddings)
Technical Assessment	Administer and auto-grade quizzes/coding tests	Rule-based grading + sandboxed code execution
Recruiter Dashboard	Present ranked shortlists and assessment results	Aggregated reporting interface

IV. IMPLEMENTATION

The system was implemented as a web application, with the resume-parsing module built using a Python-based NLP pipeline incorporating a pretrained transformer-based named-entity recognition model fine-tuned on a labeled resume dataset. The candidate-matching module computes sentence embeddings for extracted skills and job requirements using a pretrained sentence-embedding model, combined with a configurable keyword-match weighting to produce a final ranking score.

The technical assessment module was implemented with a question bank supporting category- and difficulty-tagged multiple-choice questions, and a code-execution sandbox supporting common programming languages, with submitted code executed against test cases within resource-limited,

B. Resume Parsing Accuracy

Resume-parsing accuracy was evaluated by comparing automatically extracted fields (name, skills, education, experience) against manually annotated ground-truth values for each resume in the evaluation dataset, computing field-level extraction accuracy.

C. Candidate Ranking Relevance

Candidate-ranking relevance was assessed by comparing the system's top-ranked candidates for each job posting against independent shortlists prepared by experienced recruiters, measuring the proportion of system-ranked top candidates that also appeared in the recruiter-prepared shortlist.

D. Assessment Auto-Grading Accuracy and Time Savings

isolated containers to prevent unsafe execution.

V. EVALUATION METHODOLOGY

A. Dataset

Evaluation was conducted using a dataset of resumes and corresponding job postings collected with consent from a campus placement cell, covering common technical roles such as software developer, data analyst, and network engineer positions.

Auto-grading accuracy for the technical assessment module was verified by comparing automated scores against manual grading of a sample of quiz and coding submissions. Recruiter time savings were estimated by comparing time spent on initial candidate shortlisting using the system versus fully manual resume review, based on a pilot deployment with placement-cell staff.

VI. RESULTS AND DISCUSSION

A. Resume Parsing Accuracy

The resume-parsing module achieved an overall field-extraction accuracy of 91.4% across name, contact, education, skills, and experience fields, with skills extraction showing marginally lower accuracy (87.6%) than structured fields such as education, attributable to the greater variability in how candidates describe technical skills across different resume formats.

B. Candidate Ranking Relevance

The system's top-ranked candidates matched recruiter-prepared shortlists in 86% of cases across evaluated job postings, indicating strong but not perfect alignment with expert recruiter judgment, with most disagreements attributable to soft-skill or experience-context factors not captured by the skill-matching algorithm.

C. Assessment Auto-Grading Accuracy

Automated grading of multiple-choice quizzes achieved 100% agreement with manual grading, as expected for objectively scored question types, while automated coding-assessment scoring achieved 96.2% agreement with manual grading, with discrepancies primarily arising from edge-case test scenarios not fully captured by the predefined test cases.

D. Recruiter Time Savings

Pilot deployment with placement-cell staff showed an approximately 65% reduction in time spent on initial candidate shortlisting compared to fully manual resume review, primarily attributable to automated ranking eliminating the need for line-by-line manual resume review during the initial screening stage.

VII. COMPARATIVE SUMMARY

Table II presents a comparative summary of key evaluation metrics, and Table III summarizes recruiter time allocation across recruitment stages before and after system deployment, highlighting the concentration of time savings in the initial screening and shortlisting stage.

VIII. CHALLENGES AND LIMITATIONS

Challenges encountered include handling highly varied resume formats and layouts that reduce parsing accuracy for non-standard documents, the inherent limitation of skill-matching algorithms in capturing soft skills, cultural fit, and other qualitative factors that experienced recruiters consider, and the need for careful test-case design in the coding-assessment module to minimize auto-grading discrepancies. The evaluation was conducted with a single placement cell and a limited job-posting set, and broader deployment across diverse organizational contexts would strengthen the generality of the findings.

TABLE II — Summary of Evaluation Results

Metric	Result
Overall resume field-extraction accuracy	91.4%
Skills-field extraction accuracy	87.6%
Candidate-ranking agreement with recruiter shortlist	86%

Metric	Result
MCQ auto-grading agreement with manual grading	100%
Coding-assessment auto-grading agreement	96.2%
Reduction in recruiter shortlisting time	~65%

TABLE III — Recruiter Time Allocation Before and After System Deployment

Recruitment Stage	Manual Process (avg. time)	With THAS (avg. time)
Initial resume screening/shortlisting	38 min per posting	13 min per posting
Technical assessment coordination	22 min per posting	6 min per posting
Assessment grading	30 min per posting	3 min per posting
Total (per job posting)	90 min	22 min

IX. FUTURE SCOPE

Future enhancements could include incorporating candidate interview-scheduling automation, sentiment and communication-skill assessment through structured video-response analysis, and expanding the candidate-matching algorithm to incorporate recruiter feedback as a continuous learning signal to improve ranking relevance over time. Integration with applicant tracking systems and support for bias-mitigation techniques in the matching algorithm are also identified as important directions for responsible deployment at scale.

X. CONCLUSION

This paper presented the design, development, and evaluation of a Technical and Hiring Assistance System integrating automated resume parsing, candidate-job matching, and online technical assessment within a unified platform. Evaluation results showed strong resume-parsing accuracy (91.4%), good alignment between system-ranked candidates and recruiter judgment (86% agreement), high automated assessment grading accuracy, and an approximately 65% reduction in recruiter time spent on initial shortlisting. These findings support the practical value of integrated recruitment-assistance platforms in reducing recruiter workload and improving consistency in technical hiring workflows, while highlighting the continued importance of recruiter oversight for factors beyond automated skill matching.

XI. REFERENCES

[1] Devlin, J., Chang, M.W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proc. NAACL-HLT, 4171-4186.

[2] Reimers, N., and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Proc. EMNLP-IJCNLP, 3982-3992.

[3] Zaroor, A., Maree, M., and Sabha, M. (2017). A hybrid approach to conceptual classification and ranking of resumes and their corresponding job posts. In Proc. Int. Conf. Intelligent Decision Technologies, 107-119.

[4] Kmail, A.B., Maree, M., Belkhatir, M., and Alhashmi, S.M. (2015). An automatic online recruitment system based on exploiting multiple semantic resources and concept-relatedness measures. In Proc. IEEE ICTAI, 620-627.

[5] Chuang, T.Y., and Lin, C.Y. (2018). A resume screening system based on named entity recognition. Journal of Information Science and Engineering, 34(6), 1567-1582.

[6] Faliagka, E., Ramantas, K., Tsakalidis, A., and Tzimas, G. (2012). Application of machine learning algorithms to an online recruitment system. In Proc. Int. Conf. Internet and Web Applications and Services, 215-220.

[7] Roy, P.K., Chowdhary, S.S., and Bhatia, R. (2020). A machine learning approach for automation of resume recommendation system. Procedia Computer Science, 167, 2318-2327.

[8] Malinowski, J., Keim, T., Wendt, O., and Weitzel, T. (2006). Matching people and jobs: A bilateral recommendation approach. In Proc. HICSS, 137.

[9] Kroenke, D.M., and Auer, D.J. (2015). Database Processing: Fundamentals, Design, and Implementation (14th ed.). Pearson Education.

[10] Pressman, R.S., and Maxim, B.R. (2019). Software Engineering: A Practitioner's Approach (9th ed.). McGraw-Hill Education.

[11] Docker Inc. (2021). Docker container security best practices for sandboxed code execution. Technical Documentation.

ISSN: Applied For (2026) | DOI: 10.XXXXX/IJVAST.2026.XXXXX | Open Access — CC BY 4.0 License