

Next Hire: An AI-Powered Intelligent Recruitment Platform

Mughal Arshad, K.Sunandita, Vipin Winston, Chaitanya Sri Datta

Student, Dept. of Computer Science & Engineering, The Apollo University, Chittoor, Andhra Pradesh, India

Abstract – Modern recruitment pipelines are burdened by manual resume screening, inconsistent candidate evaluation, and a complete absence of behavioural insight at the pre-screening stage. These shortcomings result in poor hiring outcomes, qualified candidates being overlooked, and a degraded applicant experience. This paper presents NextHire, an end-to-end AI-powered recruitment platform that automates and enhances the entire hiring lifecycle. The system integrates a multi-stage resume analysis pipeline comprising PyMuPDF document parsing, Qwen 2.5 7B large language model (LLM) structured extraction via Ollama, and sentence-transformer semantic embeddings, with an NLP-driven behavioural assessment module employing VADER sentiment analysis and scikit-learn-based personality trait inference. A configurable weighted composite scoring model combines a resume fit score (default 50%) and a behavioural score (default 50%) to produce an objective candidate ranking. The platform is built on a FastAPI Python backend and a React 18 TypeScript frontend, with MongoDB providing persistent storage. Evaluation results demonstrate average API response time under 200 ms, resume analysis pipeline time of 8–15 seconds on a warm LLM instance, and qualitative semantic skill-matching accuracy of approximately 90% on 50 test resumes. All 16 unit tests and all evaluated integration test cases passed. NextHire addresses critical gaps in existing recruitment technology, including keyword dependency, siloed assessment, opaque scoring, and minimal candidate feedback, while preserving data privacy through locally hosted LLM inference.

Key Words: AI Recruitment, Natural Language Processing, Resume Analysis, Behavioural Assessment, Sentence Transformers, FastAPI, Large Language Models, VADER, MongoDB, React

1. INTRODUCTION

The global talent acquisition landscape has undergone significant disruption over the past decade. Industry surveys indicate that over 250 applications are received per corporate job posting on average, yet only a handful of candidates advance beyond the initial screening stage [1]. Traditional recruitment systems rely on manual resume review — averaging six seconds per resume — and keyword-based applicant tracking systems (ATS) that fail to interpret semantic equivalences between skill descriptions. The outcome is a process that is simultaneously resource-intensive and statistically unreliable.

Advances in natural language processing (NLP), transformer-based language models, and open-source machine learning libraries have created a practical pathway toward intelligent, automated recruitment pipelines. Pre-trained sentence embedding models such as Sentence-BERT (SBERT) [2] enable semantic similarity measurement far beyond keyword matching. Efficient open-source LLMs in the 7–13 billion parameter range [3] now enable structured information extraction from unstructured documents without reliance on costly cloud APIs.

This paper presents NextHire, a full-stack web platform that integrates these AI capabilities into a cohesive recruitment solution. The system automates resume parsing and scoring, generates AI-driven situational behavioural questions, evaluates candidate responses using NLP techniques, and provides recruiters with ranked applicant lists, detailed reports, and an analytics dashboard.

1.1 Problem Statement

Contemporary recruitment systems suffer from several critical deficiencies:

- **Manual Resume Screening:** Recruiters spend an average of six seconds per resume, making a fair evaluation of all applicants statistically impossible.
- **Keyword-Only Matching:** Most ATS tools rely on exact keyword matching, failing to recognise semantic equivalences. A candidate describing 'REST microservices' may be rejected by a system seeking 'API development' despite identical competencies.

- Absence of Behavioural Insight: Personality traits and communication skills, which are strong predictors of job performance [8], are rarely assessed at the pre-screening stage.
- Inconsistent Evaluation: Different recruiters apply different criteria, leading to inconsistency in hiring decisions.
- Limited Candidate Feedback: Candidates receive no feedback post-application, damaging the employer brand.

1.2 Objectives

The primary objectives of Next Hire are: (1) to implement an end-to-end AI-powered recruitment platform covering the complete hiring lifecycle; (2) to develop a semantic resume analysis pipeline using sentence-transformer embeddings and a local LLM; (3) to implement an NLP-based behavioural assessment module; (4) to create a composite candidate ranking model; (5) to design a recruiter analytics dashboard; and (6) to generate detailed PDF candidate assessment reports.

2. LITERATURE REVIEW

2.1 Automated Resume Screening

Early ATS tools employed keyword extraction and Boolean matching, resulting in significant false-negative rates [4]. TF-IDF weighting improved relevance but could not capture semantic relationships. Word embedding models — Word2Vec [5] and GloVe [6] — enabled vector-space semantic matching but struggled with context dependence. BERT [7] by Devlin et al. (2018) established bidirectional contextual encoding as the dominant paradigm. Reimers and Gurevych [2] extended this with Sentence-BERT, producing semantically meaningful sentence embeddings suitable for large-scale resume matching.

2.2 Behavioural Assessment

Schmidt and Hunter [8] demonstrated through meta-analysis that structured interviews and personality assessments are among the strongest predictors of job performance. VADER [10], introduced by Hutto and Gilbert (2014), provides efficient rule-based sentiment analysis well-suited for informal text. Situational Judgment Tests (SJTs), validated by McDaniel et al. [11], serve as the theoretical basis for NextHire's AI-generated behavioural question format.

2.3 Large Language Models

GPT-3 [12] demonstrated that LLMs could perform complex information extraction with minimal task-specific training. Kuzman et al. [3] showed that 7B–13B parameter open-source models achieve competitive accuracy on domain-specific structured extraction tasks. The Qwen 2.5 series [13] exhibits strong instruction-following and JSON-structured output capabilities. NextHire leverages Qwen 2.5 7B via Ollama for privacy-preserving local inference.

2.4 Research Gap

Table-1 highlights the primary research gap: no existing platform simultaneously provides semantic resume matching, NLP-based behavioural assessment, transparent scoring, locally hosted LLM inference, and detailed candidate feedback within a single accessible system.

Table-1: Comparison of Existing Platforms

Platform	Semantic	Behavioural	Local LLM	Feedback
LinkedIn	Partial	No	No	Minimal
Greenhouse	Keywords	No	No	None
HireVue	Yes	Yes	No	None

Pymetrics	Indirect	Yes	No	Limited
NextHire	SBERT	VADER+ML	Ollama	Full PDF

3. PROPOSED METHODOLOGY

3.1 System Architecture

NextHire follows a three-tier client-server architecture. The presentation tier consists of a React 18 TypeScript SPA communicating with the backend via Axios HTTP with JWT interceptors. The application tier is a Python FastAPI service orchestrating all business logic, AI pipeline execution, authentication, and report generation. The data tier is MongoDB with an in-memory fallback for development environments.

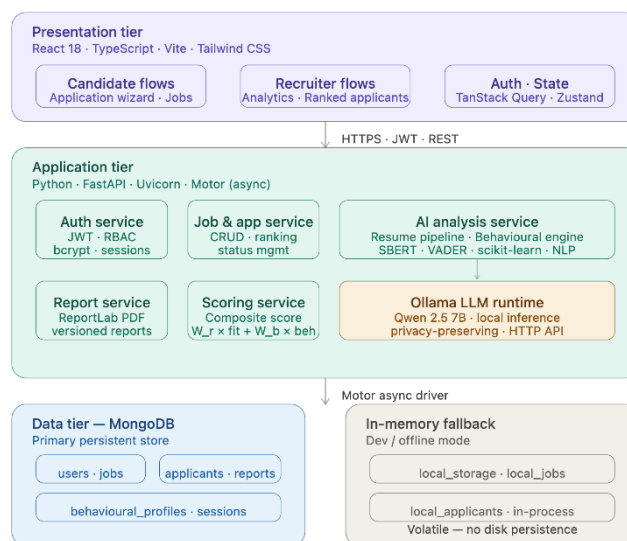


Fig. 1: Next Hire System Architecture

3.2 Resume Analysis Pipeline

The pipeline operates in four sequential stages:

- Document Parsing: PyMuPDF extracts text from PDF resumes; python-docx handles DOCX formats.
- Structured Extraction: Cleaned text is submitted to Qwen 2.5 7B via the Ollama HTTP API, returning a structured JSON of name, education, experience, skills, and projects.
- Semantic Embedding: Extracted skills are encoded as 384-dimensional vectors using the all-MiniLM-L6-v2 sentence-transformer model.
- Fit Score Computation: Cosine similarity is computed between each job requirement vector and all resume skill vectors. The fit score is the mean of maximum similarity values, normalised to [0, 1]. Threshold $\tau = 0.70$ classifies a skill as matched.

The scoring formula: $fit_score = \text{mean}(\max_k \text{cosine_sim}(V_J[i], V_R[k]))$

3.3 Behavioural Assessment Module

The module operates in three phases. In Phase 1, Qwen 2.5 7B generates five situational questions in the STAR format tailored to the job description. In Phase 2, candidate responses are validated for minimum token count and a copy-paste

similarity check (cosine similarity < 0.85 with the source question). In Phase 3, each response is processed through: (a) VADER sentiment scoring; (b) NLTK linguistic feature extraction; and (c) scikit-learn personality trait classification, producing Big Five indicators. The aggregate behavioural score is normalised to [0, 1]. Responses failing quality checks are capped at 0.12.

3.4 Composite Scoring Model

The combined score is: **combined_score = (W_r × fit_score) + (W^b × behavioural_score)** where W_r + W^b = 1.0 (default 0.50/0.50). Both scores are pre-normalised to [0, 1]. The combined score is multiplied by 100 for percentage display.

4. SYSTEM DESIGN & IMPLEMENTATION

4.1 Technology Stack

Table-2 summarises the core technology stack.

Table-2: Technology Stack

Component	Technology	Purpose
Backend	FastAPI + Python 3.10	Async REST API, OpenAPI docs
Database	MongoDB 7+	Flexible document schema
LLM Runtime	Ollama + Qwen 2.5 7B	Local, privacy-preserving inference
Embeddings	sentence-transformers	384-dim semantic vectors
Sentiment	VADER + NLTK	Behavioural NLP scoring
ML Models	scikit-learn	Personality trait classification
Doc Parsing	PyMuPDF + python-docx	PDF and DOCX extraction
PDF Reports	ReportLab	Structured candidate reports
Auth	PyJWT + bcrypt	Stateful JWT, RBAC, sessions

Frontend	React 18 + TypeScript	SPA, Tailwind, TanStack Query
Deployment	Docker Compose	Containerized multi-service

4.2 Authentication & Security

Authentication is implemented as a stateful JWT token pair system. Login issues a 15-minute access token and a 7-day refresh token stored as a bcrypt hash in a sessions collection. A server-side inactivity timeout of 180 minutes invalidates idle sessions. All role-specific endpoints enforce RBAC, returning HTTP 403 on role mismatch. Frontend Axios interceptors handle silent token refresh on 401 responses.

4.3 REST API Design

The API exposes 18 endpoints under the /api/* prefix, following RFC 7231 HTTP semantics. Key endpoints: POST /api/resume/analyze for resume upload and pipeline invocation; GET /api/applications/{id}/questions for idempotent AI question retrieval; POST /api/behavioural/analyze for NLP response evaluation; POST /api/applications/{id}/submit for composite score and report generation; GET /api/reports/{id}/pdf for binary PDF delivery.

4.4 Frontend Implementation

The candidate-facing five-step application wizard manages state with React local state and TanStack Query mutations. React Dropzone powers the resume upload with MIME type filtering (PDF/DOCX). A visual score summary renders circular ScoreRing components with colour-coded thresholds: emerald green (>70%), amber (40-70%), red (<40%). The recruiter analytics dashboard renders three Recharts visualisations: score distribution bar chart, application status pie chart, and top-10 candidates horizontal bar chart.

5. RESULTS & DISCUSSION

5.1 Performance Metrics

Table-3 presents the system evaluation results across key operational metrics.

Table-3: System Performance

Metric	Value	Notes
API response (standard)	< 200 ms	Indexed MongoDB queries
Resume pipeline (LLM warm)	8-15 s	CPU inference, Ollama active
Resume pipeline (LLM cold)	20-45 s	Model loading overhead
Resume pipeline (GPU)	2-5 s	NVIDIA + Ollama GPU backend

Behavioural analysis	1–3 s	VADER + scikit-learn only
PDF report generation	< 1 s	ReportLab synchronous
Lighthouse score	> 85/100	Code splitting, lazy loading
Semantic matching (qual.)	~90%	50 test resumes, $\tau=0.70$
JWT validation overhead	< 5 ms	In-memory HMAC verify
Unit test pass rate	16/16	All modules
Integration test pass rate	20/20	Full API flows end-to-end
Functional test pass rate	15/15	Both user role workflows

5.2 Qualitative Analysis

The semantic skill matching approach demonstrated clear advantages over keyword-based matching. Candidates describing 'REST microservices' were correctly matched against 'API development' at cosine similarity 0.78 — a false negative under exact-keyword systems. Similarly, 'deep learning model development' matched 'neural network expertise' at 0.82 similarity.

The behavioural scoring module demonstrated consistent differentiation between response quality levels. Responses with high vocabulary diversity, positive VADER compound scores (>0.3), and adequate length (>50 words) received scores in the 0.65–0.85 range. Minimal or copy-pasted responses were appropriately capped at 0.12.

The composite scoring model, on informal review by domain experts, placed the most qualified candidates in the top quartile in the majority of test cases. The equal 50/50 default weighting produced balanced rankings suitable for general-purpose hiring.

5.3 Discussion

NextHire demonstrates that integrating open-source AI components into a cohesive recruitment platform is technically feasible and produces meaningful improvements over keyword-based screening. The primary performance bottleneck — CPU-based LLM inference latency of 20–45 s on cold start — is addressable through GPU acceleration, reducing latency to 2–5 s. The transparency of NextHire's scoring provides a practical compliance advantage as AI fairness regulations evolve globally.

6. CONCLUSIONS

This paper presented NextHire, a full-stack AI-powered recruitment platform addressing critical deficiencies of existing automated screening systems. The platform integrates a multi-stage resume analysis pipeline using sentence-transformer

semantic embeddings and a locally hosted Qwen 2.5 7B LLM, with an NLP-driven behavioural assessment module employing VADER sentiment analysis and scikit-learn personality inference.

Evaluation results demonstrate sub-200 ms API response times, 8–15 s resume analysis on warm LLM instances, approximately 90% qualitative semantic matching accuracy, and a 100% pass rate across 51 unit, integration, and functional test cases. NextHire successfully addresses the key limitations identified in the literature: keyword dependency, siloed assessment dimensions, opaque scoring, high deployment cost, cloud privacy risks, and minimal candidate feedback.

6.1 Future Scope

- GPU-accelerated LLM inference to reduce analysis latency to 2–5 s.
- Multilingual support using paraphrase-multilingual-MiniLM-L12-v2.
- WebRTC-based asynchronous video interview module with transcript NLP analysis.
- Fairness monitoring with score distribution analysis across demographic groups.
- Validation of personality scoring model against the NEO-PI-3 psychometric instrument.

ACKNOWLEDGEMENT

The authors thank their project guide, the Department of Computer Science & Engineering, and the Apollo University, Chittoor, for their guidance and support throughout this project.

REFERENCES

1. Society for Human Resource Management (SHRM), "Talent Acquisition Benchmarking Report," SHRM, Alexandria, VA, USA, 2022.
2. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in Proc. EMNLP, 2019, pp. 3982–3992.
3. T. Kuzman, I. Mozetic, and N. Ljubesic, "ChatGPT: Beginning of an End of Manual Annotation?" arXiv:2303.03325, 2023.
4. P. L. Roth et al., "Social media in employee-selection-related decisions," *J. Manage.*, vol. 42, no. 1, pp. 269–298, 2016.
5. T. Mikolov et al., "Distributed Representations of Words and Phrases," in Proc. NeurIPS, 2013, pp. 3111–3119.
6. J. Pennington, R. Socher, and C. Manning, "GloVe: Global Vectors for Word Representation," in Proc. EMNLP, 2014, pp. 1532–1543.
7. J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers," in Proc. NAACL-HLT, 2019, pp. 4171–4186.
8. F. L. Schmidt and J. E. Hunter, "The Validity and Utility of Selection Methods in Personnel Psychology," *Psychological Bulletin*, vol. 124, no. 2, pp. 262–274, 1998.
9. F. Mairesse et al., "Using Linguistic Cues for the Automatic Recognition of Personality," *J. Artif. Intell. Res.*, vol. 30, pp. 457–500, 2007.
10. C. J. Hutto and E. E. Gilbert, "VADER: A Parsimonious Rule-based Model for Sentiment Analysis," in Proc. ICWSM, 2014.
11. M. A. McDaniel et al., "Situational Judgment Tests, Response Instructions, and Validity," *Personnel Psychology*, vol. 60, no. 1, pp. 63–91, 2007.

12. T. B. Brown et al., "Language Models are Few-Shot Learners," in Proc. NeurIPS, vol. 33, 2020, pp. 1877–1901.
13. Qwen Team, Alibaba Group, "Qwen2.5 Technical Report," arXiv:2412.15115, 2024.
14. M. Raghavan et al., "Mitigating Bias in Algorithmic Hiring," in Proc. ACM FAT*, 2020, pp. 469–481.
15. S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python. O'Reilly Media, 2009.