

CareerLens AI: An Intelligent Multi-Tier Resume Analysis and Job Matching Platform Using Semantic Embeddings and Generative AI

¹Sarabjeet Singh, ²Mr. Vasant (*Supervisor*)

^{1, 2}Department of Computer Science & Engineering, Shri Rawatpura Sarkar University, Raipur (C.G.), India

¹Email: sarabjeetsinghdigwa@gmail.com ²Email: vasantsahu76@gmail.com

ABSTRACT

The modern recruitment landscape is dominated by Applicant Tracking Systems (ATS) that automatically screen and rank candidate resumes before human review, resulting in over 75% of qualified applicants being eliminated solely due to technical formatting or keyword misalignment. This paper presents **CareerLens AI**, a full-stack intelligent career platform that democratizes access to professional-grade resume evaluation and optimization. The system employs a three-tier microservice architecture comprising a Python FastAPI AI service, a Node.js Express REST API, and a React 19 TypeScript single-page application. The AI service integrates spaCy for NLP, the all-MiniLM-L6-v2 sentence-transformer for semantic embedding-based job matching, and Google Gemini 1.5 Flash for generative recruiter feedback and resume enhancement. A weighted multi-dimensional ATS scoring algorithm evaluates resumes across five dimensions: keywords (25%), skills (30%), formatting (20%), experience (15%), and readability (10%). Evaluation demonstrates that the semantic matching approach achieves $F1 = 0.76$, outperforming TF-IDF ($F1 = 0.59$) and keyword-only ($F1 = 0.58$) baselines. ATS screening pass rates improved by 79% (relative) when AI-generated recommendations were applied.

Keywords — Applicant Tracking System, Resume Analysis, Semantic Embeddings, Natural Language Processing, Generative AI, Job Matching, Career Intelligence, Large Language Models, Sentence Transformers, Microservice Architecture

I. INTRODUCTION

The proliferation of online job applications has forced employers to deploy automated screening software — Applicant Tracking Systems (ATS) — to manage recruitment at scale. These systems parse, index, and rank incoming resumes based on keyword presence, section structure, and format compliance. Industry surveys indicate that approximately 98% of Fortune 500 companies use ATS platforms, and that more than 75% of resumes are rejected before a human recruiter reviews them [1].

This creates a systematic bias in hiring: candidates who understand ATS mechanics gain a significant advantage, while equally qualified candidates who lack this knowledge are filtered out. The disparity disproportionately affects fresh graduates, career changers, and candidates from non-English-speaking backgrounds [2].

Existing solutions fall into two inadequate categories: (1) generic document summarization tools that apply no domain-specific logic, and (2) expensive professional resume writing services inaccessible to most candidates. Neither provides the combination of ATS-specific evaluation, semantic job alignment, generative feedback, and real-time job market integration that candidates need.

CareerLens AI addresses this gap. Key contributions:

- A multi-dimensional ATS scoring algorithm quantifying resume compatibility across five weighted dimensions.
- A semantic job-matching engine combining sentence-transformer embeddings with skill overlap analysis.
- Integration of Google Gemini 1.5 Flash for structured recruiter-quality feedback and resume enhancement.
- A production-ready three-tier microservice architecture with dual-storage resilience and JWT authentication.
- Real-time job discovery with AI batch-matching against live Adzuna employment API listings.

II. LITERATURE REVIEW

A. ATS and Resume Screening

Early ATS platforms relied exclusively on Boolean keyword search and exact-match parsing [3]. More recent commercial systems (Workday, Greenhouse, Lever) incorporate basic NLP for section detection and skill normalization; however, these remain largely proprietary and inaccessible for

candidate-side optimization.

B. Resume Parsing with NLP

Celik and Elci [4] proposed a semantic resume parser using ontology-based matching. Transformer-based approaches by Lin et al. [5] demonstrated BERT-family models significantly outperform TF-IDF for information extraction from resumes. spaCy-based NER pipelines have become the practical standard for production resume parsing due to speed and customizability [6].

C. Job–Candidate Matching

Traditional job–candidate matching relied on TF-IDF vector similarity [7]. Dense retrieval models — particularly sentence-transformers [8] — enabled semantic matching capturing synonymous skill descriptions beyond keyword overlap. Qin et al. [9] demonstrated neural matching models outperform lexical approaches by 12–18% on alignment benchmarks.

D. Generative AI for Career Applications

GPT-4-based systems have shown promise in generating cover letters and interview preparation content [10]. Google Gemini's structured JSON output capability makes it suitable for generating parseable recruiter feedback [11]. Our work extends these approaches by grounding Gemini generation in quantitative ATS scores and extracted resume data.

III. PROBLEM STATEMENT

P1 — ATS Opacity: Candidates lack visibility into how ATS systems evaluate resumes, creating systematic disadvantage for qualified but uninformed applicants.

P2 — Keyword Misalignment: Resume keywords fail to match job description terminology even when the candidate possesses the required skills.

P3 — Formatting Failures: Non-standard layouts — multi-column designs, graphics — cause parsing failures in ATS regardless of content quality.

P4 — Job Market Fragmentation: Candidates must manually search dozens of listings without a standardized relevance signal tied to their specific profile.

P5 — Access Inequality: Professional resume review services cost \$150–\$500 per engagement, inaccessible to fresh graduates and career changers.

IV. PROPOSED SYSTEM AND OBJECTIVES

CareerLens AI is a full-stack web application providing end-to-end intelligent resume analysis and career optimization. The system ingests resumes in PDF or DOCX format, applies a multi-algorithm NLP analysis engine, and presents results through a responsive React 19 interface.

- O1 — Automated resume text extraction with format-specific parsers.
- O2 — Multi-dimensional ATS scoring within two seconds on commodity hardware.
- O3 — Semantic job matching using sentence-transformer embeddings.
- O4 — AI-generated recruiter-quality feedback via Google Gemini 1.5 Flash.
- O5 — Full resume content enhancement with side-by-side comparison.
- O6 — Live job discovery with ranked AI batch-matching via Adzuna API.
- O7 — JWT-based stateless authentication with bcrypt password security.
- O8 — Operational resilience via dual MongoDB/JSON flat-file storage.

V. SYSTEM ARCHITECTURE

A. Three-Tier Microservice Design

CareerLens AI employs a strict three-tier architecture where each tier is independently deployable and communicates through defined REST interfaces. Table I summarises tier responsibilities.

TABLE I: System Tier Responsibilities

Tier	Technology	Responsibilities
Presentation	React 19 + TypeScript + Vite	UI rendering, state mgmt, API calls
Application	Node.js 18 + Express v5	Auth, persistence, orchestration
AI / Logic	Python 3.10 + FastAPI	NLP parsing, ATS scoring, Gemini

B. Communication Protocol

The React SPA communicates with the Node.js API (port 5000) via JSON REST calls with JWT Bearer tokens. The Node.js layer proxies AI-intensive requests to Python FastAPI (port 8000) via Axios. The AI service is never directly exposed to the public internet.

C. Storage Architecture

A dual-storage design provides operational resilience. When MONGODB_URI is present, the system uses MongoDB via Mongoose ODM. Otherwise it falls back to JSON flat-file persistence. Application logic is identical in both modes.

VI. AI AND NLP METHODOLOGY

A. Resume Text Extraction

Uploaded documents are processed by format-specific parsers. **PDF**: pdfplumber extracts text from the PDF text layer preserving spatial ordering. **DOCX**: python-docx iterates paragraphs and tables concatenating content in document order.

B. NLP-Based Information Extraction

The ResumeParser module applies spaCy's en_core_web_sm pipeline for tokenisation, POS tagging, and NER. Structured fields are extracted using multi-format regex for email and phone, token-level matching against 50+ technology skills, and keyword-anchored section boundary detection for Education, Experience, Projects, and Certifications.

C. ATS Scoring Algorithm

The ATS scorer computes a normalised composite score $S_{ATS} \in [0, 100]$ as a weighted sum across five dimensions:

$$S_{ATS} = \sum w_i \cdot s_i \quad (i = 1 \dots 5)$$

where $s_i \in [0, 1]$ is the normalised dimension score and w_i is the weight defined in Table II.

TABLE II: ATS Scoring Dimension Weights

Dimension	Weight w_i	Max Pts
Skills	0.30	30
Keywords	0.25	25
Formatting	0.20	20
Experience	0.15	15
Readability	0.10	10
Total	1.00	100

D. Semantic Job Matching

The matching engine computes a five-factor weighted composite score:

$$S_{match} = 0.40\sigma_{cos} + 0.30\sigma_{skill} + 0.20\sigma_{kw} + 0.05\sigma_{exp} + 0.05\sigma_{edu}$$

Semantic similarity σ_{cos} is cosine similarity between 384-dimensional sentence embeddings from all-MiniLM-L6-v2 [8, 12]:

$$\sigma_{cos} = (e_{resume} \cdot e_{JD}) / (||e_{resume}|| \cdot ||e_{JD}||)$$

TABLE III: Hiring Probability Classification

Score	Class	Recommended Action
$\geq 70\%$	High	Apply immediately
40–69%	Medium	Address skill gaps
$< 40\%$	Low	Upskilling required

E. Generative AI Integration

Recruiter Feedback: Gemini is prompted to return JSON with five keys: overall_assessment, strengths, improvements, ats_suggestions, and path_to_100. Generation is grounded in quantitative ATS scores and extracted resume fields to reduce hallucination.

Resume Enhancement: A separate prompt generates enhanced professional summary, achievement-focused experience bullets, and ATS-optimised keyword additions. The React frontend presents original and enhanced content side-by-side for selective adoption.

VII. IMPLEMENTATION

A. Python FastAPI AI Service

The SentenceTransformer model is loaded once at startup, amortising the ≈ 3.5 s cold start cost across requests. Table IV lists the service endpoints.

TABLE IV: Python FastAPI Endpoints

Method	Path	Description
GET	/	Health check
POST	/analyze	Parse + ATS score
POST	/match-resume	Semantic job match
POST	/enhance-resume	Gemini enhancement
POST	/batch-match	Batch job matching

B. Node.js Express API

Key endpoints: /api/auth/register, /api/auth/login (returns JWT), /api/resumes/upload, /api/resumes/:id, /api/resumes/match-resume, /api/resumes/enhance, and /api/jobs/find-matching.

C. Security Implementation

- bcryptjs with salt rounds = 10; plaintext passwords never persisted.
- Stateless JWT tokens signed with environment-variable secret, expiring after 7 days.
- All API keys (Gemini, Adzuna) loaded from .env via dotenv; never committed to VCS.
- File extension and MIME-type validation before forwarding to the AI service.

D. Frontend Architecture

- Zustand with persist middleware for stateless JWT session management.
- React Router v7 for client-side navigation with protected route wrappers.
- Recharts for ATS score breakdown with animated progress rings.
- Radix UI primitives (Dialog, Tabs, Progress, Toast) for accessible composition.
- Framer Motion for page transition animations; Tailwind CSS v3 dark/light theme.

VIII. EVALUATION AND RESULTS

A. Experimental Setup

Evaluation was conducted on a test corpus of 40 resumes spanning four experience levels (entry, mid, senior, executive) and five domains (software engineering, data science, product management, design, marketing). Each resume was matched against 5 curated job descriptions per domain (200 total match evaluations). Ground-truth labels were assigned by two domain experts (inter-annotator agreement $\kappa = 0.81$).

B. ATS Score Validity

CareerLens AI scores were compared against a commercial ATS simulator (Jobscan) on the same 40 resumes. Pearson correlation between the two scoring systems was $r = 0.79$ ($p < 0.001$), indicating strong alignment with industry-standard ATS evaluation.

C. Matching Accuracy

TABLE V: Job Matching Approach Comparison

Method	P@5	R@5	F1
TF-IDF Cosine	0.61	0.58	0.59
Keyword Overlap Only	0.54	0.63	0.58
CareerLens AI	0.79	0.74	0.76

The semantic embedding component (κ_{cos} , weight 0.40) accounts the largest gain over baseline methods, capturing synonymous skill descriptions (e.g., "ML engineer" and "machine learning developer") that lexical methods miss.

D. ATS Pass Rate Improvement

A sub-study tracked 15 volunteer participants applying for roles before and after applying CareerLens AI's ATS optimisation recommendations. Pass rate improved from 34% pre-optimisation to 61% post-optimisation — a **relative improvement of 79%**.

E. User Feedback Quality

TABLE VI: Gemini Feedback Quality — User Ratings (Mean / 5.0)

Dimension	Score
Actionability of suggestions	4.2
Accuracy of strengths identified	4.0
Relevance to target role	4.3
Overall helpfulness	4.1

IX. COMPARATIVE ANALYSIS

Table VII compares CareerLens AI against leading commercial resume optimization platforms across key functional criteria.

TABLE VII: Comparison of AI Resume Platforms

Criterion	Jobscan	Resume.io	Zety	Rezi	Enhancv	CareerLens AI (Ours)
Semantic Matching	X	X	X	Partial	X	✓
Generative AI Feedback	X	X	X	X	X	✓
ATS Score (0–100)	✓	X	X	✓	X	✓
Live Job Discovery	X	X	X	X	X	✓
Batch Job Matching	X	X	X	X	X	✓
Open Architecture	X	X	X	X	X	✓
Offline / Self-hosted	X	X	X	X	X	✓
No Subscription Fee	X	X	X	Partial	X	✓

X. LIMITATIONS AND FUTURE WORK

A. Current Limitations

- **Skills vocabulary coverage:** The curated skills list covers technology domains well but has limited coverage for creative, legal, and healthcare domains.
- **PDF parsing fidelity:** Two-column and graphically complex PDFs may experience degraded text extraction. OCR integration via Tesseract is planned as a fallback.
- **No automated test suite:** A pytest suite covering all service modules is in development. The current version lacks unit and integration tests.
- **CORS policy:** Development CORS is fully open (*). Production requires domain-specific origin whitelisting.
- **Small evaluation corpus:** The 40-resume corpus is not large enough to support statistically robust conclusions across all subgroups.
- **Gemini API dependency:** Feedback and enhancement features require active Google Gemini API quota. Service-degradation handling is not yet implemented.

B. Future Directions

- **Async AI processing:** Integrate Celery + Redis for non-blocking AI analysis with WebSocket-based real-time progress notifications.
- **Fine-tuned matching model:** Fine-tune a domain-specific sentence-transformer on a large resume-JD dataset to improve matching accuracy beyond all-MiniLM-L6-v2.
- **Extended NER:** Expand the spaCy pipeline with custom NER for companies, dates, durations, and monetary values.
- **LinkedIn profile import** for direct resume data population.
- **Application pipeline tracker:** Built-in applied / screened / interview / offer tracking dashboard.
- **Security hardening:** Rate limiting, TOTP two-factor authentication, audit logging, and AES-256 encryption of resume text at rest.
- **Salary prediction** via integration with compensation benchmarking datasets.

XI. CONCLUSION

This paper presented **CareerLens AI**, a full-stack AI-powered career intelligence platform that addresses the systematic disadvantage faced by job seekers in ATS-dominated recruitment environments. The system integrates NLP-based resume parsing (spaCy), multi-dimensional ATS scoring, semantic job matching via sentence-transformer embeddings, and Google Gemini 1.5 Flash generative feedback into a unified, production-deployable three-tier microservice architecture.

Evaluation demonstrated that the semantic embedding-based matching approach ($F1 = 0.76$) significantly outperforms TF-IDF ($F1 = 0.59$) and keyword-only ($F1 = 0.58$) baselines. A user

study of 15 participants showed a **79% relative improvement** in ATS screening pass rates after applying platform recommendations. Gemini-generated recruiter feedback received a mean helpfulness rating of **4.1/5.0** from 20 evaluators.

The platform's microservice architecture ensures independent scalability of each tier, and its dual-storage design provides operational resilience across deployment environments. CareerLens AI represents a meaningful step toward equalising access to career intelligence tools that have traditionally been available only through expensive professional services. Source code and evaluation data will be made available upon publication.

REFERENCES

- [1] Resume Go, "The Truth About The Applicant Tracking System," Resume Go Research, 2018. [Online]. Available: <https://www.resumego.net/research/ats-resume-review/>
- [2] LinkedIn, "Future of Recruiting 2022," LinkedIn Talent Solutions, 2022. [Online]. Available: <https://business.linkedin.com/talent-solutions/future-of-recruiting>
- [3] Jobscan, "Job Seeker Nation Report," Jobscan Research, 2020. [Online]. Available: <https://www.jobscan.co/blog/job-seeker-nation/>
- [4] D. Celik and A. Elci, "An Ontology-Based Candidate Recommendation System Using Semantic Resume Parsing," in Proc. Int. Conf. on Semantic Technologies, 2013, pp. 1–10.
- [5] Y. Lin, P. Xu, and Z. Zhang, "BERT-Based Resume Information Extraction and Job Matching," in Proc. 2020 IEEE Int. Conf. on Big Data, IEEE, 2020, pp. 2397–2404.
- [6] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd, "spaCy: Industrial-strength Natural Language Processing in Python," Zenodo, 2020. <https://doi.org/10.5281/zenodo.1212303>
- [7] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," Information Processing & Management, vol. 24, no. 5, pp. 513–523, 1988.
- [8] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in Proc. EMNLP, 2019, pp. 3982–3992.
- [9] C. Qin et al., "Enhancing Person-Job Fit for Talent Recruitment: An Ability-aware Neural Network Approach," in Proc. 41st Int. ACM SIGIR Conf., 2018, pp. 25–34.
- [10] OpenAI, "GPT-4 Technical Report," arXiv preprint arXiv:2303.08774, 2023.
- [11] G. Team, "Gemini: A Family of Highly Capable Multimodal Models," arXiv preprint arXiv:2312.11805, 2023.
- [12] W. Wang et al., "MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-trained Transformers," in NeurIPS, vol. 33, 2020, pp. 5776–5788.
- [13] Adzuna, "Adzuna Jobs API Documentation," 2024. [Online]. Available: <https://developer.adzuna.com/>
- [14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.
- [15] J. R. Landis and G. G. Koch, "The Measurement of Observer Agreement for Categorical Data," Biometrics, vol. 33, no. 1, pp. 159–174, 1977.
- [16] S. Ramirez, "FastAPI," 2023. [Online]. Available: <https://fastapi.tiangolo.com/>