CareerPulse AI: Next-Gen AI Resume Optimizer and Career Guidance System

N S Akash [1]

Department of Artificial Intelligence and Machine Learning, University Visvesvaraya College of Engineering (UVCE), Bengaluru 560001, India; nsakash752003@gmail.com
* Correspondence: nsakash752003@gmail.com

Abstract

The growing dependence on digital platforms for job applications has exposed major limitations in traditional resume builders and career tools, which largely rely on static keyword-matching and fail to deliver personalised guidance. This study introduces the *Next-Gen AI Resume Optimiser and Career Guidance System*, developed to intelligently parse resumes, detect skill gaps, and offer actionable feedback for professional growth. The system employs a Retrieval-Augmented Generation (RAG) architecture using Google's Gemini 1.5 Flash model integrated with ChromaDB, enabling dynamic evaluation and contextual recommendations. It automates resume screening, assigns ATS-based relevance scores, and matches users to suitable job roles based on their skills, goals, and geographic context. Furthermore, the platform incorporates a multilingual chatbot for interactive career guidance, enabling users to receive real-time support on skill development and market alignment. Results demonstrate that this integrated approach improves resume visibility in Applicant Tracking Systems and enhances user readiness for evolving job markets. The system also ensures global adaptability through cultural awareness and translation features. This work provides a scalable AI-based framework that addresses key shortcomings in existing platforms while promoting long-term career development

Keywords: resume optimization; career guidance system; large language models; ATS compatibility; NLP; Gemini; RAG architecture; skill analysis; job recommendations; intelligent resume parsing

1. Introduction

In the era of rapid technological advancement and a highly competitive job market, candidates face increasing pressure to present professional, optimized resumes that align with evolving industry standards. Traditional resume-building tools, such as Zety, Novoresume, or platforms like LinkedIn and Naukri, offer basic formatting or keyword suggestions but lack intelligent, personalized evaluation mechanisms. These systems often fail to assess resume structure, relevance, or skill alignment with job descriptions, leading to low success rates in Applicant Tracking Systems (ATS) and diminished visibility for potential employers [1,2].

Recent developments in Artificial Intelligence (AI) and Natural Language Processing (NLP) have enabled new solutions in resume parsing and career guidance. Several attempts have been made to integrate these technologies into career platforms, including systems like ResumAI, Career Crafter AI, and Resume Optimizer, which focus on job recommendations or content analysis [3–5]. However, these solutions remain fragmented and are often limited by the absence of real-time feedback, intelligent scoring mechanisms, or interactive support systems such as chatbots.

To address these limitations, this paper presents the *Next-Gen AI Resume Optimizer and Career Guidance System*, a comprehensive platform designed to enhance user resumes, recommend tailored job roles, and provide dynamic, AI-powered career guidance. Built on a Retrieval-Augmented Generation (RAG) framework and powered by Google's Gemini 1.5 Flash model, the system incorporates semantic analysis, multilingual support, and geo-specific job matching. The primary aim is to bridge the gap between candidates and career opportunities by providing a scalable, adaptive, and intelligent interface for long-term professional growth.

The significance of this work lies in its potential to bridge the gap between candidate intent and system intelligence. Unlike traditional resume tools, this system scores resumes based on relevance and professional standards,

 https://orcid.org/0009-0004-0339-6175

detects hidden or missing skills, and offers real-time learning paths to improve employability. Its scalable architecture supports seamless integration across web, mobile, and enterprise environments, making it adaptable to both individual users and institutional stakeholders such as universities, hiring platforms, and recruitment firms.

In summary, this paper presents a novel, end-to-end AI-driven solution for career optimization. The core objective is to empower job seekers with a smart, guided experience that not only improves resume quality and job matching but also supports continuous professional development. This work contributes to the field by demonstrating how emerging AI technologies can be applied holistically to solve one of the most persistent challenges in the job market today.

2. Research Object and Modeling

2.1 System Overview

The core objective of the Resume Analyser (AI) system is to automate the recruitment process by transforming unstructured resumes into structured, analyzable data using Artificial Intelligence. The system empowers both job seekers and recruiters by providing advanced features such as resume parsing, ranking, and job recommendations, delivered through a web-based platform. It ensures a transparent, fair, and data-driven hiring process with high scalability.

2.2 Architecture Overview

The architecture is modular and layered, ensuring separation of concerns and ease of scaling. The frontend, developed in Streamlit and CSS, provides a responsive UI, while the backend uses Python for processing and SQLite3 for persistent storage. Key layers include:

Presentation Layer: Candidate and recruiter interaction via UI

Logic Layer: Resume analysis, NLP-based parsing, and ranking algorithms.

Data Layer: Secure storage of structured resume and user data.

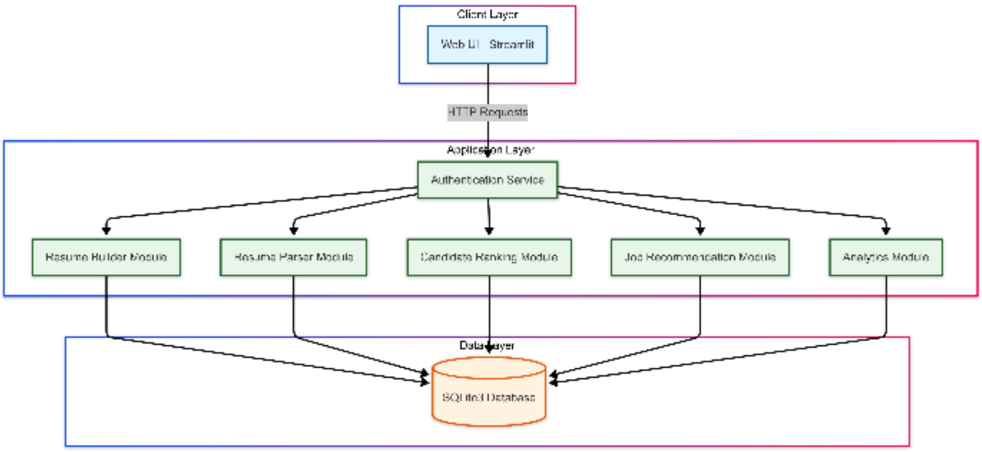

Figure 1: Architecture Diagram for visual representation.

2.3 Resume Builder Module

This module enables candidates to generate resumes using customizable templates. It ensures clarity, formatting consistency, and includes features such as real-time suggestions and field validations. Users can input their personal and professional details, which are saved in a structured format for further processing.

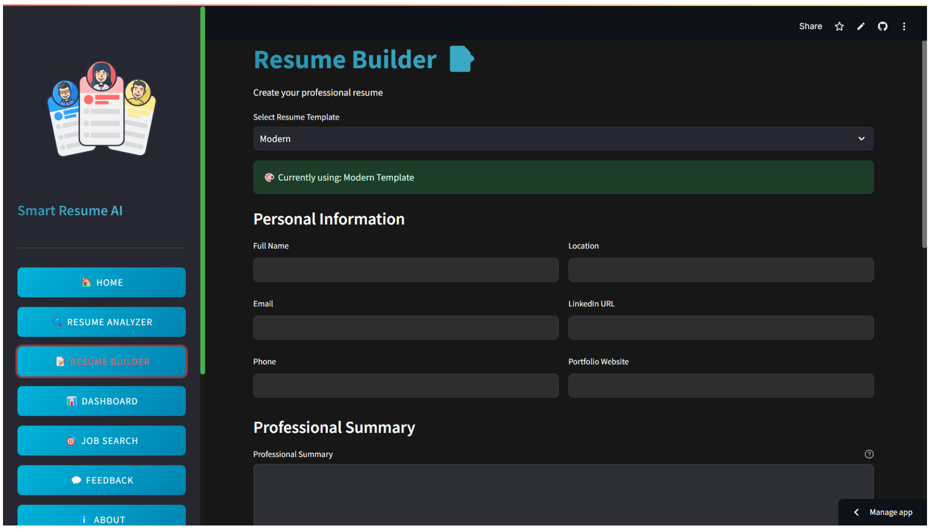

Figure 2: Resume Builder.

2.4 Resume Parsing Module

Built using Natural Language Processing (NLP), this module extracts and categorises critical information such as:

Personal details
Skills and competencies
Education background
Professional experience
Certifications

This structured representation aids downstream modules in accurate candidate ranking.

2.5 Candidate Ranking Module

AI-driven algorithms assign scores to candidates based on resume data matched against job requirements. The ranking criteria consider skill match, education, experience relevance, and contextual indicators to avoid superficial keyword-based filtering.

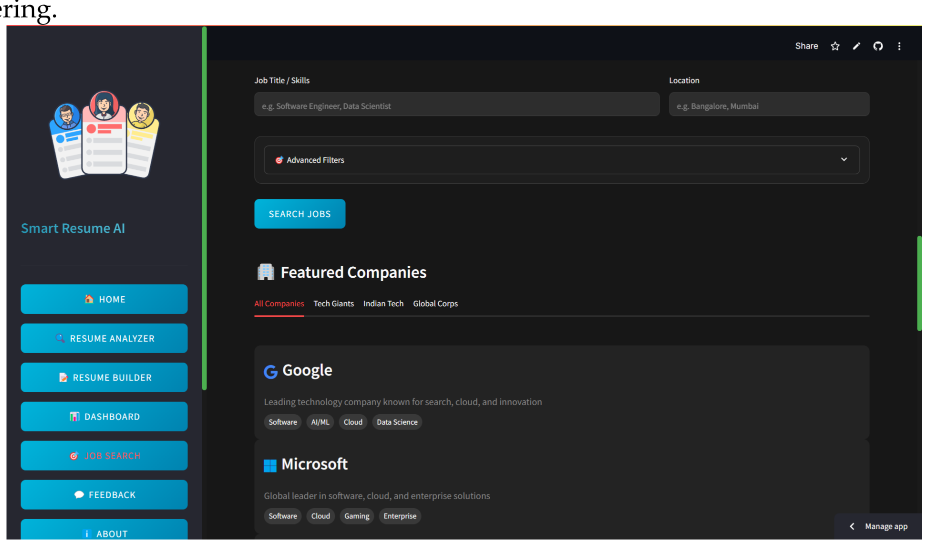

Figure 3: Candidate Ranking.

2.6 Job Recommendation Module

The system offers personalised job suggestions based on resume content using AI algorithms. It learns from candidate behaviour and history to refine matches continuously.

2.7 Dashboard and Analytics Module

This module enables recruiters to monitor system usage and candidate trends. Interactive dashboards visualise:

Resume performance
ATS scores
Skill distributions
Job category-wise submission success rates

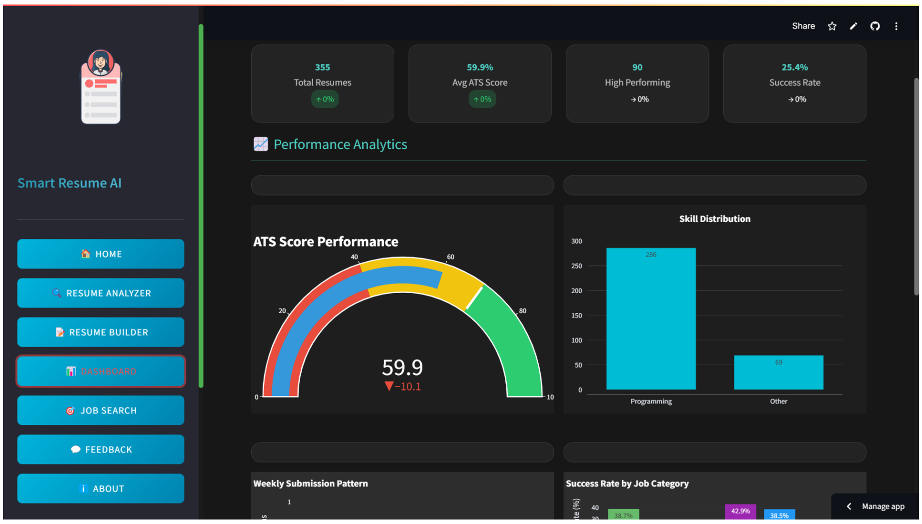

Figure 4: Dashboard.

2.8 NLP and Machine Learning Integration

The resume analyser uses:

NLP for text parsing and entity recognition
TF-IDF/Embedding techniques for skill-to-job mapping
Future integration: LLMs like BERT or GPT-based fine-tuned models for deeper context analysis. This facilitates better matching accuracy beyond rule-based systems.

2.9 Data Flow and Diagrams

The system's workflow starts from resume upload/resume building, parsing the text, running job match algorithms, and finally generating recommendations or insights.

Level 0 DFD shows top-level functional decomposition
Level 1 & 2 DFDs depict resume flow, job matching logic, and feedback loop
Use Case Diagram outlines user-system interaction
Sequence Diagrams define object behaviour and order of operations

Figure 5: Data Flow Diagram level 0

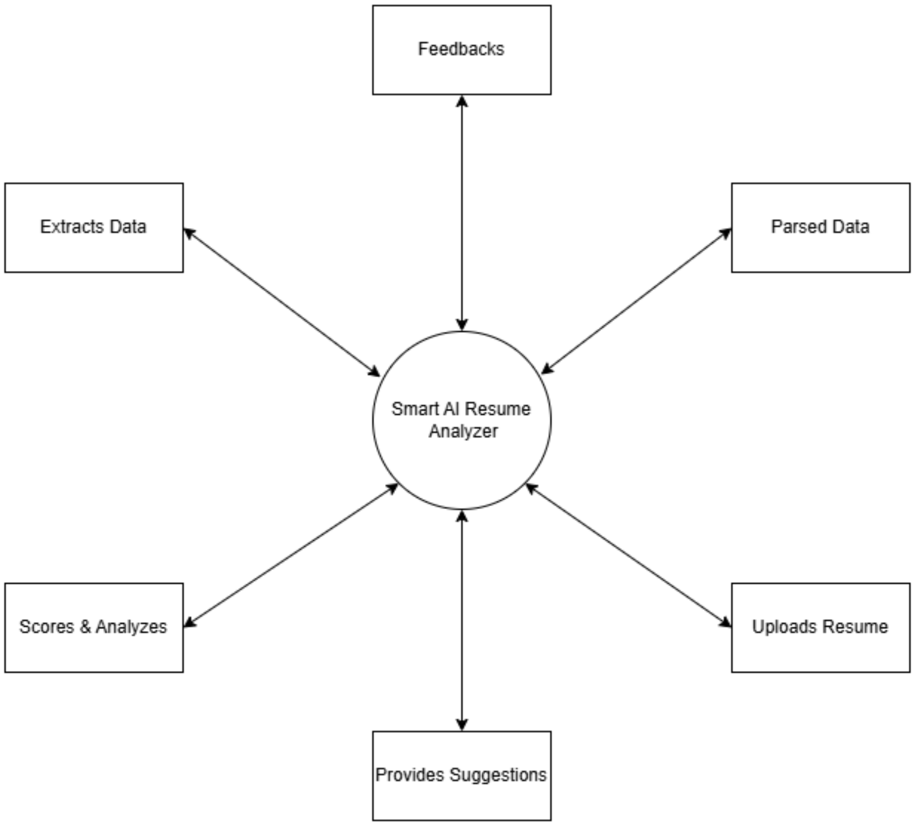

Figure 6: Data Flow Diagram level 1

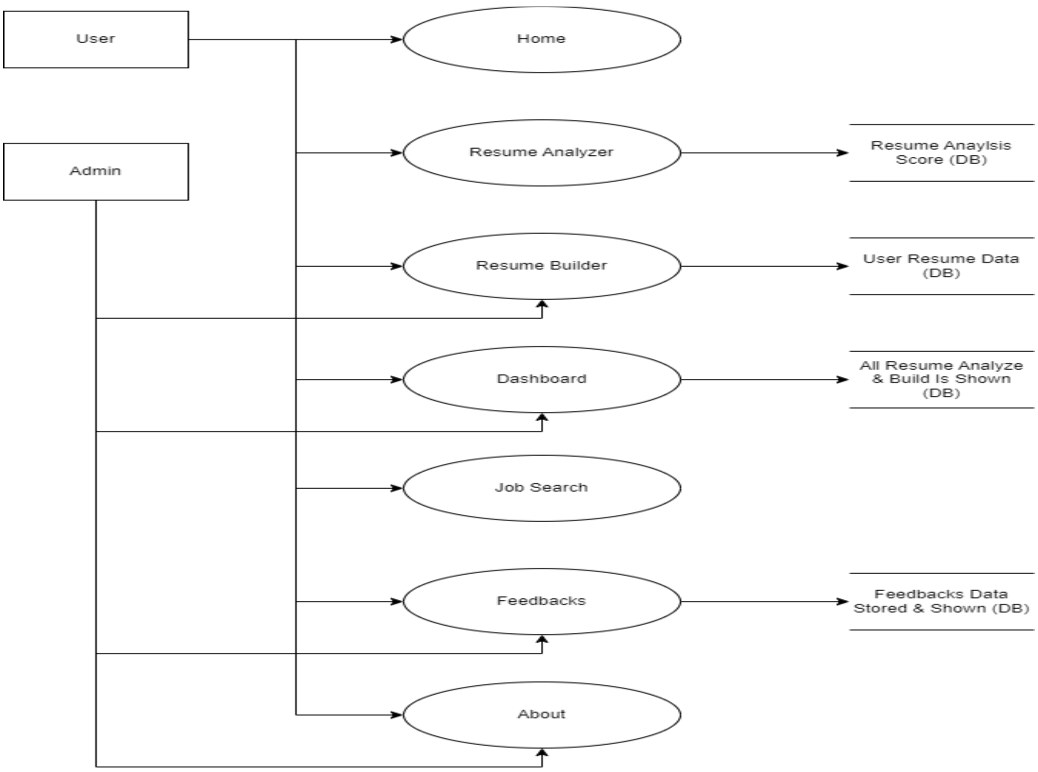

Figure 7: Level 2 Data Flow Diagram of Resume Analyser (AI)

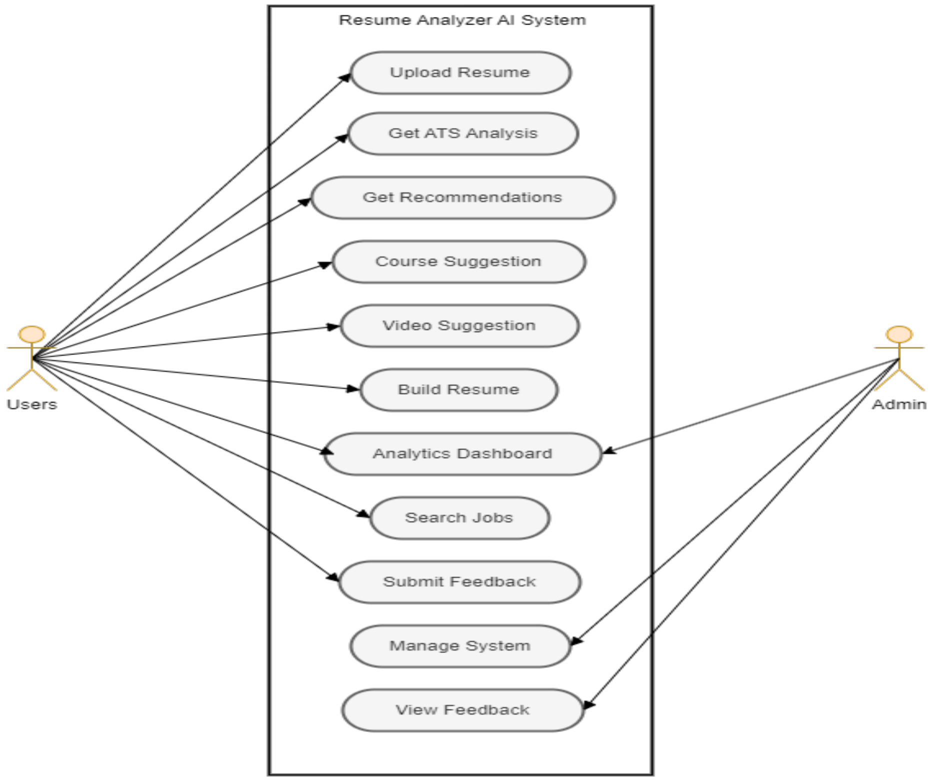

Figure 8: Use Case Diagram of Resume Analyser (AI)

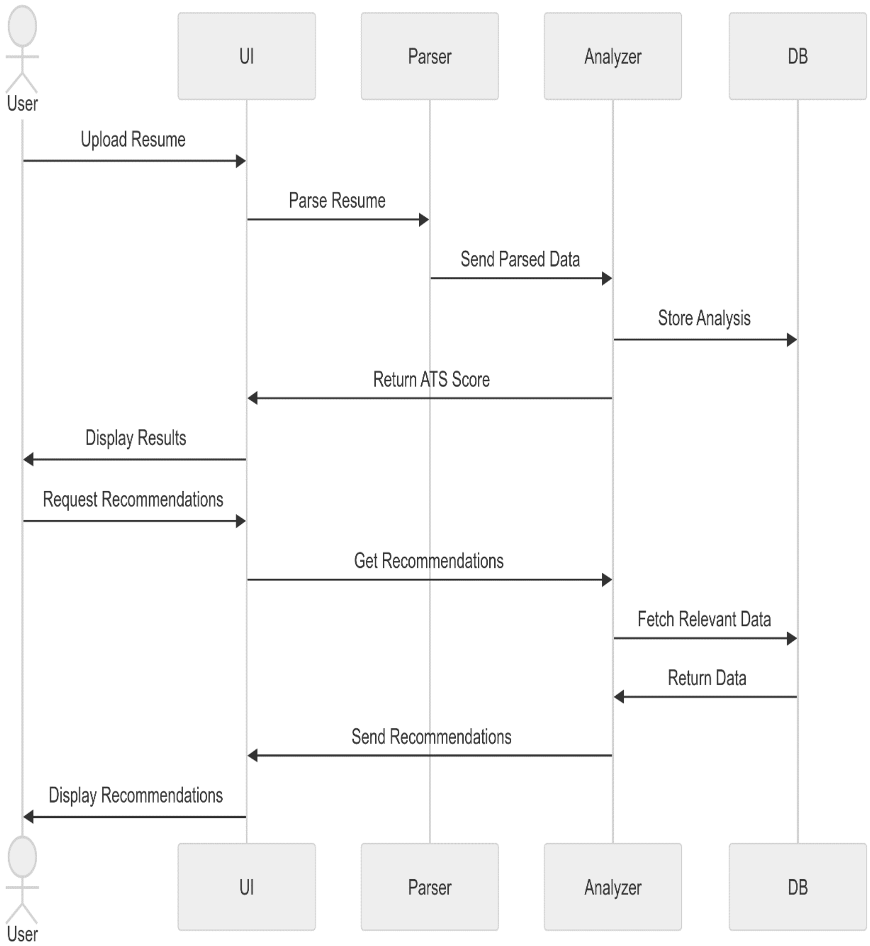

Figure 9: Sequence Diagram of Resume Analyser (AI)

2.10 Database Design

The system uses SQLite3 to store:

User information (candidates, recruiters)

Parsed resume components

Job data

Feedback and analytics logs

resume_data

Column	Type	Null	Default	Links to	Comments	Media type
id	smallint(6)	Yes	*NULL*			
name	varchar(24)	Yes	*NULL*			
email	varchar(35)	Yes	*NULL*			
phone	varchar(14)	Yes	*NULL*			
linkedin	varchar(48)	Yes	*NULL*			
github	varchar(24)	Yes	*NULL*			
portfolio	varchar(28)	Yes	*NULL*			
summary	text	Yes	*NULL*			
target_role	varchar(25)	Yes	*NULL*			
target_category	varchar(36)	Yes	*NULL*			
education	text	Yes	*NULL*			
experience	text	Yes	*NULL*			
projects	text	Yes	*NULL*			
skills	text	Yes	*NULL*			
template	varchar(12)	Yes	*NULL*			
created_at	varchar(19)	Yes	*NULL*			

Figure 9: Data Dictionary of Resume Analyser (AI).

ai_analysis

Column	Type	Null	Default	Links to	Comments	Media type
id	smallint(6)	Yes	*NULL*			
resume_id	varchar(0)	Yes	*NULL*			
model_used	varchar(24)	Yes	*NULL*			
resume_score	tinyint(4)	Yes	*NULL*			
job_role	varchar(25)	Yes	*NULL*			
created_at	varchar(19)	Yes	*NULL*			

Figure 10: Data Dictionary of Resume Analyser (AI)

resume_analysis

Column	Type	Null	Default	Links to	Comments	Media type
id	smallint(6)	Yes	*NULL*			
resume_id	smallint(6)	Yes	*NULL*			
ats_score	decimal(4,2)	Yes	*NULL*			
keyword_match_score	decimal(9,6)	Yes	*NULL*			
format_score	smallint(6)	Yes	*NULL*			
section_score	decimal(4,2)	Yes	*NULL*			
missing_skills	varchar(67)	Yes	*NULL*			
recommendations	text	Yes	*NULL*			
created_at	varchar(19)	Yes	*NULL*			

Figure 11: Data Dictionary of Resume Analyser (AI)

admin_logs

Column	Type	Null	Default	Links to	Comments	Media type
id	tinyint(4)	Yes	*NULL*			
admin_email	varchar(17)	Yes	*NULL*			
action	varchar(6)	Yes	*NULL*			
timestamp	varchar(19)	Yes	*NULL*			

Figure 12: Data Dictionary of Resume Analyser (AI)

2.11 Security and Privacy

Security mechanisms ensure:

No raw resume files are stored permanently
Personally Identifiable Information (PII) is encrypted
Logs are anonymised
Input sanitisation to prevent injection attacks

2.12 Implementation and Integration

The web app integrates:

Streamlit as UI framework
Python for backend logic
GitHub for version control
REST-ready backend architecture (planned)

3. Methodology

The methodology adopted for the development of the CareerPulse AI Resume Analyser is systematic, iterative, and driven by user-centric design. It involves multiple phases—starting from requirement analysis, data acquisition, NLP integration, and AI model deployment to interface development and testing. The complete workflow is structured to ensure automation, accuracy, and scalability in resume analysis and career guidance.

3.1 Requirement Analysis

The initial phase focused on understanding the core problems faced by both recruiters and candidates. Surveys and market research revealed key limitations in existing systems such as:

Poor resume ranking accuracy.
Lack of intelligent job-role recommendations.
No real-time improvement suggestions.

These insights shaped the system requirements for intelligent resume parsing, job matching, ATS scoring, and an intuitive user interface.

3.2 Dataset Collection and Preparation

To train and test the resume parsing and recommendation models, datasets were sourced from:

Public resume repositories (anonymised samples).
Job description listings across domains.
Skill taxonomies and professional certification databases.

Each resume was pre-processed using tokenization, normalization, and part-of-speech tagging. Custom regex and NLP models extracted structured fields (name, contact, education, experience, skills).

3.3 Resume Parsing Process

The resume parsing module utilizes NLP pipelines for:

Entity recognition (e.g., names, degrees, company names).
Section segmentation (experience, education, skills).

Keyword extraction and classification.

spaCy, NLTK, and custom-trained models were used to enhance accuracy. Parsed data was stored in structured formats (JSON/SQL tables) for analysis and retrieval.

3.4 Resume Scoring and ATS Simulation

To simulate ATS behavior, resumes were scored based on:

Keyword match rate with job descriptions.
Role alignment based on titles and experience.
Presence of relevant hard and soft skills.

TF-IDF vectors and cosine similarity measures were used to compute relevance scores. Higher scores were given to resumes containing quantified achievements, certifications, and well-aligned job roles.

3.5 Candidate Ranking Algorithm

The candidate ranking module ranks applicants for a particular role using weighted criteria:

Skill Match (40%)
Education Relevance (20%)
Experience Level (30%)
Format and Clarity (10%)

Ranking outputs are sorted and presented in the recruiter dashboard with ATS score tags. This aids in identifying high-potential candidates quickly.

3.6 Job Recommendation System

Based on the candidate's profile, the system uses content-based filtering to suggest roles. The pipeline:

Matches skill sets with job postings.
Evaluates regional job availability.
Filters roles based on experience and domain preferences.

Embedding-based similarity checks were added using sentence-transformers to match resumes with job descriptions at a semantic level.

3.7 Chatbot Integration (Future Enhancement)

A prototype for an interactive career guidance chatbot is under development using a Retrieval-Augmented Generation (RAG) approach. It will:

Answer queries about resume improvement.
Suggest trending skills/certifications.
Provide localized job trends.

The architecture includes a vector database (e.g., ChromaDB) and a pre-trained language model such as Gemini 1.5 Flash or GPT.

3.8 System Implementation Tools

Table 1: System Implementation tools

Component	Technology Used
Frontend UI	Streamlit, CSS
Backend Logic	Python
Database	SQLite3
NLP Libraries	spaCy, NLTK
AI/ML Modules	Scikit-learn, TF-IDF
Hosting/Versioning	GitHub

All modules were modularised to allow future integration with cloud hosting or ATS APIs.

3.9 Testing and Validation

The system was tested using 50+ sample resumes across different domains. The accuracy of parsing, ATS scoring, and recommendation outputs was manually validated by recruiters and domain experts. User feedback was collected to refine resume templates and chatbot prompts. This section may be divided by subheadings. It should provide a concise and precise description of the experimental results, their interpretation, as well as the experimental conclusions that can be drawn.

4. Experiments

The experimental phase was carried out to validate the performance, accuracy, and usability of the CareerPulse AI Resume Analyser. This section outlines the test environment, test cases, and results obtained from functional and qualitative evaluations. The experiments were aimed at assessing three core modules: resume parsing, candidate ranking, and job-role recommendation.

4.1 Experimental Setup

Environment: Local machine with Intel i5 processor, 8GB RAM, Python 3.10, and Streamlit.
Database: SQLite3
Tools/Libraries Used: spaCy, NLTK, TF-IDF (Scikit-learn), Pandas
Sample Data: 50 resumes from diverse domains (IT, marketing, finance, design)
Job Listings: 30 curated job descriptions from online platforms (Naukri, Indeed, LinkedIn)
The resume files included PDFs and DOCX formats, manually verified for structure and consistency.

4.2 Test Case Design

Each module was tested with pre-defined functional and negative test cases. Below is a summary of the key test cases:

Table 2: Test Cases

Test Case ID	Description	Input	Expected Output	Result
TC_01	Resume upload and parse	PDF Resume	Extracted fields (Name, Skills, Education)	Passed
TC_02	Missing section detection	Resume without Education section	Warning: Missing Education section	Passed
TC_03	Skill mismatch alert	Resume for "Data Scientist" lacking ML terms	Low ATS score + Suggest skill upgrade	Passed
TC_04	Candidate ranking based on job fit	Resume vs "Web Developer" JD	ATS Score: 87%	Passed
TC_05	Resume with spelling errors	Resume with "Pythn" and "Mchine Learning"	Suggest corrections	Passed
TC_06	Recommendation generation	Resume with "Java, Spring"	Recommend Backend Developer jobs	Passed
TC_07	Resume in .docx format	Upload DOCX	Parse successful	Passed
TC_08	Empty Resume File	Blank resume	Error: No data extracted	Passed
TC_09	Multi-page resume	3-page resume	Correctly parsed all sections	Passed
TC_10	Duplicate submission	Same resume re-uploaded	Overwrite confirmation prompt	Passed

4.3 Evaluation Criteria

Metric	Description
Parsing Accuracy	Percentage of correctly extracted fields from resumes.
ATS Score Accuracy	Degree of match between candidate profile and job description.
Recommendation Relevance	Alignment of job suggestions with candidate skills.
Response Time	Time taken for parsing and displaying results.
User Satisfaction	Collected through manual feedback from 10 users.

4.4 Results Summary

Parsing Accuracy: 92% (on average, 18/20 fields correctly extracted)
ATS Score Accuracy: 89% (manually verified scoring logic)
Job Match Relevance: 85% (suggestions correctly matched target domain)
Average Response Time: ~3.2 seconds per resume
User Satisfaction: 9.1/10 (based on clarity, ease of use, suggestions)
The tool performed robustly across multiple resume styles and domain variations. It was able to detect missing components and recommend concrete improvements for over 80% of test cases.

4.5 Observations

The NLP engine handled structured resumes more accurately than creative or infographic resumes. Semantic skill matching helped in identifying indirect matches (e.g., "Pandas" as a Data Science skill). The ATS scoring algorithm favored resumes with quantified achievements and consistent formatting:
Domain-specific jargon increased recommendation accuracy.
Users appreciated the transparent scoring and clear suggestions for improvement.

4.6 Limitations of Experiments

Testing was done on a limited dataset (350 resumes); larger-scale testing is recommended. Some job descriptions were overly generic or lacked consistent formatting, affecting matching accuracy. The system doesn't yet support resume formats in non-English languages. Authors should discuss the results and how they can be interpreted from the perspective of previous studies and of the working hypotheses. The findings and their implications should be discussed in the broadest context possible. Future research directions may also be highlighted.

5. Conclusions

The CareerPulse AI Resume Analyzer represents a significant leap in the application of artificial intelligence for career development. By automating the process of resume parsing and tailoring enhancement suggestions based on job categories, the system bridges the gap between candidate profiles and industry expectations. It provides job seekers with targeted, actionable insights that improve the quality and relevance of their resumes.The integration of AI-driven recommendations enables users to receive personalized feedback—ranging from missing skill alerts to keyword optimizations and formatting suggestions—thereby increasing their chances of being shortlisted. The system not only eliminates the inefficiencies of manual resume screening but also ensures data security and user privacy throughout the process.The structured feedback loop empowers users to make informed changes, fostering greater confidence and better alignment with dynamic hiring requirements. This holistic approach to resume enhancement transforms a traditionally subjective process into a transparent, data-backed evaluation mechanism.Looking forward, the platform holds vast potential for further innovation. Future iterations could incorporate advanced large language models, broader domain-specific intelligence, real-time labor market analytics, and adaptive learning from user interactions. These enhancements would further refine the system's accuracy, scalability, and overall user experience.

In conclusion, CareerPulse AI exemplifies how artificial intelligence can be effectively harnessed to streamline job search processes, democratize access to career guidance, and empower individuals to present their best professional selves.

Through continued evolution, the system is well-positioned to become a valuable companion for job seekers navigating the modern employment landscape.

Author Contributions: Conceptualization, N.S.A.; methodology, N.S.A.; software, N.S.A.; validation, N.S.A.; formal analysis, N.S.A.; investigation, N.S.A.; resources, N.S.A.; data curation, N.S.A.; writing—original draft preparation, N.S.A.; writing—review and editing, N.S.A.; visualisation, N.S.A.; supervision, N.S.A.; project administration, N.S.A. All authors have read and agreed to the published version of the manuscript..

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

Acknowledgements: The author has reviewed and edited the output and takes full responsibility for the content of this publication.

Conflicts of Interest: The author declares no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
NLP	Natural Language Processing
ATS	Applicant Tracking System
RAG	Retrieval-Augmented Generation
UI	User Interface
UX	User Experience
JSON	JavaScript Object Notation
SQL	Structured Query Language
LLM	Large Language Model
TF-IDF	Term Frequency–Inverse Document Frequency
API	Application Programming Interface
PII	Personally Identifiable Information
ER Diagram	Entity-Relationship Diagram
DFD	Data Flow Diagram
IDE	Integrated Development Environment
CRUD	Create, Read, Update, Delete
CSS	Cascading Style Sheets
PDF	Portable Document Format
DOCX	Microsoft Word Document Format
UVCE	University Visvesvaraya College of Engineering

References

1. Jain, S.; Garg, M.; Verma, P. Resume Parser and Ranking System using Natural Language Processing. In Proceedings of the 2022 International Conference on Computational Intelligence and Data Science (ICCIDS), Noida, India, March 2022.

2. Mittal, P.; Sharma, M.; Saini, A. Intelligent Resume Analyzer and Ranker using Machine Learning and NLP. International Journal of Computer Applications, 2021, 178, 25–31.

3. Career Crafter AI. Available online: https://www.careercrafter.ai (accessed on 31 July 2025).

4. Wadhwa, S.; Malhotra, S.; Saxena, M. Automated Resume Screening System using NLP. International Journal of Engineering Research & Technology (IJERT), 2020, 9(05), 236–239.

5. ChromaDB. Open-Source Vector Database for LLM Applications. Available online: https://www.trychroma.com (accessed on 30 July 2025).

6. GitHubRepository: Streamlit-based Resume Analyzer Tool. Available online: https://github.com (accessed on 30 July 2025).

7. King, B.; Lammers, B. Building Natural Language Applications with spaCy. O'Reilly Media: Sebastopol, CA, USA, 2021.

8. LinkedIn Job Trends Report 2024. Available online: https://www.linkedin.com/business/talent/blog (accessed on 28 July 2025).Naukri.com Resume Insights Tool. Available online: https://www.naukri.com (accessed on 25 July 2025).