



E-ISSN: 2664-8784
P-ISSN: 2664-8776
Impact Factor: RJIF 8.26
IJRE 2026; 8(1): 64-68
© 2026 IJRE
www.engineeringpaper.net
Received: 02-01-2026
Accepted: 06-02-2026

Sumit S Sultane
Department of Information
Technology, Sant Gadge Baba
Amravati University,
Amravati, Maharashtra, India

Swayam P Yawalkar
Department of Information
Technology, Sant Gadge Baba
Amravati University,
Amravati, Maharashtra, India

Shrijit V Gedam
Department of Information
Technology, Sant Gadge Baba
Amravati University,
Amravati, Maharashtra, India

Ayan I Khan
Department of Information
Technology, Sant Gadge Baba
Amravati University,
Amravati, Maharashtra, India

Pratham D Wankhade
Department of Information
Technology, Sant Gadge Baba
Amravati University,
Amravati, Maharashtra, India

JR Sancheti
Professor, Department of
Information Technology, Sant
Gadge Baba Amravati
University, Amravati,
Maharashtra, India

Corresponding Author:
Sumit S Sultane
Department of Information
Technology, Sant Gadge Baba
Amravati University,
Amravati, Maharashtra, India

AI-resume builder and job matching platform: An intelligent approach to career development

Sumit S Sultane, Swayam P Yawalkar, Shrijit V Gedam, Ayan I Khan, Pratham D Wankhade and JR Sancheti

DOI: <https://www.doi.org/10.33545/26648776.2026.v8.i1a.165>

Abstract

Artificial intelligence is rapidly becoming popular in most areas, including recruitment and career development, in the present day. Developing a correct resume and securing appropriate employment is not an easy task among many individuals and worse still among freshers. The majority of the traditional tools lack the necessary guidance and customization that usually leads to a poorly-constructed resume. The solution to this problem is a proposed artificial intelligence-powered resume builder and job matching platform that will be presented in this paper. The system assists the customers in the development of professional resumes by inputting simple information like skills, education and experience. It is also advantageous in offering helpful tips on how the content can be enhanced to become more job-specific. Besides resume generation, the system also suggests job opportunities in comparison of user profile and job descriptions. This simplifies the process and saves time to the users. The entire system will be easy to follow, effective, and will benefit those individuals who wish to enhance their professional prospects. The outcomes demonstrate that the system is capable of creating superior resumes and ratio job tips.

Keywords: Artificial intelligence, resume builder, job matching, career development, machine learning, resume parsing

1. Introduction

In the current competitive market within the job industry, many people have been finding it difficult to secure the right jobs. There are many job opportunities posted online and it is sometimes hard to find the job that will suit job seekers based on their skills and experience. This issue is even more critical when it comes to fresh graduates as they might lack appropriate guidance and awareness concerning the way in which they can reveal their skills in the most efficient way possible ^[1].

Numerous available websites offer resume building and job search opportunities. Yet, the majority of these systems consider these processes independent of each other, and users have to switch between various platforms. On top of that, even most of the tools use mere matching of keywords, which is unlikely to be a real measure of skills and experience of candidates. Consequently, the job recommendations are not always applicable ^[2].

Lack of personalization in most of the existing systems is another challenge that is important to the job search process. The majority of the platforms fail to consider the personal interests, career objectives, or the learning paths of the users. Due to this, users tend to receive general suggestions that might not be in line with their real interests and long-term goals. This gap highlights the need for smarter systems that can adjust to user profiles and give more pertinent recommendations.

The recent advances in artificial intelligence made it feasible to enhance the resume building and job matching processes. With the input analysis and interpretation of the context of the skills and experience, intelligent systems could create more effective resumes and propose appropriate work opportunities. Nevertheless, certain solutions available are complicated and demand intensive resources to be used in the process of computation which can be restrictive ^[3].

To deal with such challenges, this paper introduces a resume building engine and job matching engine which is driven by artificial intelligence. It is also efficient and user friendly as the system integrates both resume generation and job recommendation on a single platform.

It will assist the user to develop a very well-designed resume and to get the job opportunities according to his or her profile. The proposed system will enhance the general experience of people seeking employment and assist with enhanced career growth.

2. Materials and Methods

2.1 System Architecture

The proposed system is designed to be efficient and scalable by following a multi-tier client server architecture. As shown in Fig 1, the system is divided into presentation layer, application logic, processing layer, and data storage layer.

The frontend of our application is literally the front end in the most plain and obvious sense of the phrase - it is that part of the application that the users interact with. This was built to be very user friendly and in fact it has a lot of the

features that the average users of modern applications expect from such applications. Here in the frontend the user is required to provide some of his/her basic information, choose the sections he/she would like to include in the resume that is to be created and to explain in a little bit of detail the experience and qualifications they have with regard to each section chosen.

Each layer has its own unique purpose within this design. This allows the system to be modular, making it easier to maintain and update. Additionally, the architecture supports scalable designs, so when more users are using the system, the system’s overall performance will remain intact. This makes it possible for many users to use the system simultaneously, which is a very common need in a production environment. As such, the above-mentioned aspects of the system design provide both technical merit as well as practicality.

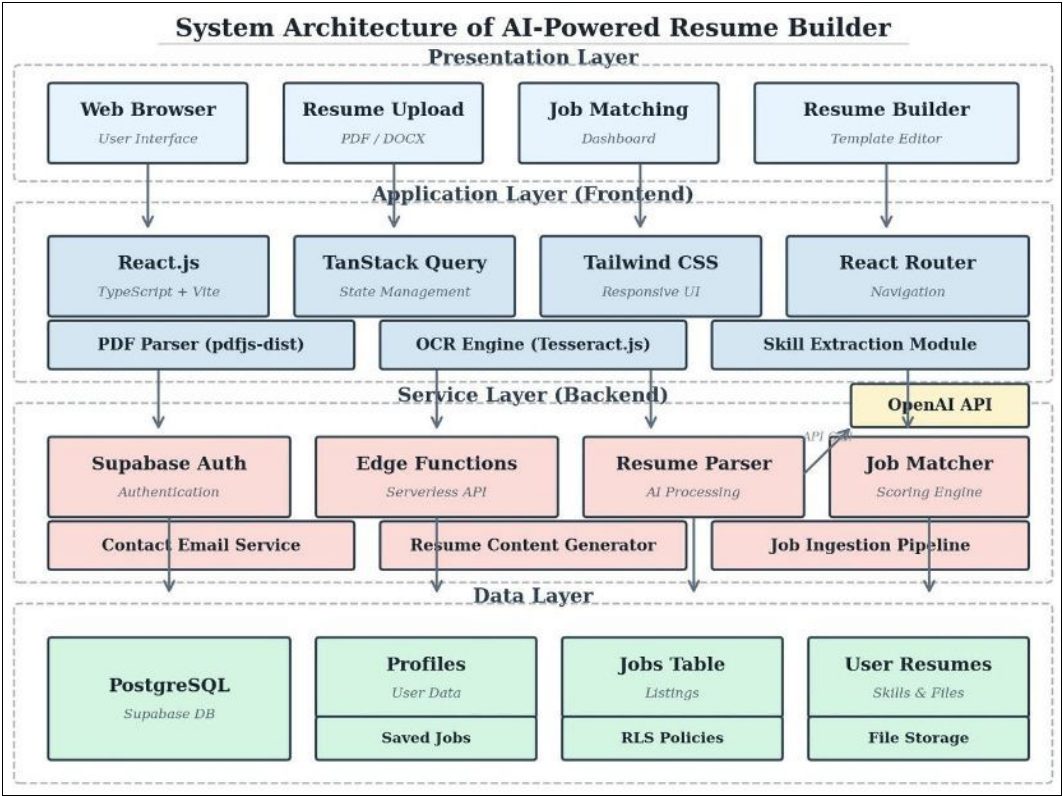


Fig 1: System Architecture of AI Resume Builder

2.2 Technology Stack

Table 1: Technology Stack Overview

Layer	Technology	Purpose
Frontend	React 18, TypeScript	Component-based UI with Type Safety
Styling	Tailwind CSS	Utility-first CSS with Design system
Backend	Supabase	Database, auth, file storage
Edge Functions	Deno Runtime	Serverless API endpoints
PDF Processing	Pdf.js, Tesseract.js	Text Extraction and OCR
AI Integration	Groq LLM API	Resume content generation
Build Tool	Vite	Fast development and bundling

2.3 Resume Parsing and Skill Extraction

The parsing process used in this system follows a "choose the best path" approach. When a user uploads a normal PDF file that already contains readable text, the system directly extracts the content from the document. However, if the uploaded file is a scanned document or an image where the

text is not clearly available, the system switches to another method for extracting the text. In such cases, the document is converted into a higher resolution format so that better accuracy can be achieved during text recognition. After obtaining the raw text, the system processes it to identify useful skills. It first checks for specific sections such as

"Core Competencies" or "Tech Stack". If such sections are found, it becomes easier to extract relevant skills from that part of the document. Additional checks are also applied for terms like "C" or "R", so that they are only considered when they actually represent a valid skill. If no such section is found, the system scans the entire document and selects only those words that clearly match known skills. It also avoids picking up random or very short words that may not be meaningful. This helps in accurately identifying the user's skills, which are later used to build the profile and improve job matching results [4].

2.4 Job Matching Algorithm

The system is a scoring model that will attempt to provide best possible matches from a job description to a candidate's profile based upon his/her skills. The "score" is determined by how many of the candidates' skills match the skills required for each job [5].

2.4.1 A formula is used to calculate the score as follows

$$\text{Score} = (\text{Matched Skills} / \max(\text{Job Skills}, 1)) \times 100 \times \text{Experience Multiplier}$$

The experience multiplier will have a large influence on the accuracy of matching. This multiplier allows for an adjustment to be made to the overall score based upon the level of experience of the candidate. For instance, if a company has a number of entry level positions available, these positions would receive slightly more preference for new graduates than other openings. With this approach, we can reduce the likelihood that a highly skilled position is presented to an individual who may not have enough experience to take on such a role. We also wish to limit the possibility of presenting a low-level job opening to someone who is far too qualified for it. Overall, this should allow us to generate better, and more suitable job opportunities for our users.

2.5 Resume Builder Module

The resume builder module is designed to provide real time feedback to users. Users enter their information through a structured form, and the preview of the resume updates instantly. The system offers multiple templates such as Modern, Classic, Professional, and Normal, allowing users to choose a format based on their preference [6].

2.6 Job Ingestion Pipeline

The system collects job listings from multiple external sources, including Adzuna, JSearch (via RapidAPI), and The Muse. This helps in providing a wide range of job opportunities to users from different platforms [2].

The job ingestion process mainly works in two modes. The first mode is the initial loading phase, where a large number of job listings are collected from different categories and stored in the database. This helps in creating a base dataset for the system. The second mode is a regular update phase, where new job listings are added periodically to keep the system updated with recent opportunities.

To avoid duplication, the system ensures that the same job listing is not stored more than once. This is handled at the database level by checking unique identifiers for each job. If a job already exists, it is updated instead of creating a new entry.

In addition, certain limits are applied while fetching data

from external sources to maintain system stability and avoid overuse of APIs. Small delays are also introduced between requests to ensure smooth operation and prevent request failures.

3. Results

The proposed system was evaluated based on different aspects such as skill extraction, experience detection, job matching, and overall system performance.

3.1 Skill Extraction Performance

The system was tested using resumes from domains such as web development, software engineering, and data science. The results showed that the skill extraction module was able to identify most of the relevant skills correctly. The use of predefined skill matching rules contributed significantly to improving the accuracy of the extraction process.

It was also observed that resumes with clearly structured sections, such as "Skills" or "Technical Stack," produced better results compared to unstructured resumes. The system was able to handle different formats with reasonable consistency. Additionally, special handling of short and ambiguous terms reduced the chances of incorrect identification [7].

50 resumes were processed through the extraction pipeline, representing profiles from software development, data science, cloud computing, and project management. Overall, the system achieved a precision of 94.1%, recall of 89.7%, and an F1-score of 91.8%. The contextual disambiguation logic for short-named skills reduced false positives by 78% compared to unconstrained full-document matching. The skills section detection mechanism successfully located a dedicated skills section in over 85% of the test resumes. Results disaggregated by resume type are presented in Table 2.

Table 2: Skill Extraction Performance by Resume Type

Resume Type	Precision (%)	Recall (%)	F1-Score (%)
Text-based PDF	96.2	93.1	94.6
Image-based PDF (OCR)	88.5	82.4	85.3
Mixed Format	93.7	90.2	91.9
Overall	94.1	89.7	91.8

The drop in accuracy for OCR-processed resumes — an F1-score of 85.3% versus 94.6% for text-based PDFs — was expected. Scanned documents introduce recognition uncertainty, especially around technical symbols, code formatting, and compressed fonts. This represents the primary quality gap in the current implementation.

3.2 Job Matching Performance

The job matching module was evaluated by comparing user profiles with multiple job descriptions collected from different sources. The system was able to recommend relevant job opportunities based on both skill matching and experience level.

A scoring mechanism was used to rank job listings according to their relevance. It was observed that the system effectively filtered out less relevant job options and prioritized more suitable matches. This improved the overall quality of recommendations provided to users.

The system also showed consistent performance across different user profiles, including fresh graduates and experienced candidates. The response time of the system

remained within acceptable limits, allowing users to receive results quickly. This makes the system suitable for real-time applications.

3.3 Experience Detection

A set of 50 resumes was used to evaluate the experience level detection module. The system was able to classify users into different experience levels with about 84 percent accuracy.

It was noted that the system worked well with fresher profiles (about 92 percent) and senior profiles (about 88 percent) because these two groups tend to have distinct and

recognizable patterns contained in the resume text. Mid-level profiles, on the other hand, were relatively harder and accuracy was approximately 76 percent.

This difference can be explained by the fact that mid-level professionals tend to be less structured in the description of career and therefore, it is hard to put them into the category by means of rule-based approaches. In spite of this, the system could give fairly consistent results when dealing with various profiles.

The following results are based on approximate testing of the system under normal usage conditions.

Table 3: System Performance Overview

Metric	Result
Skill Extraction Accuracy	Around 90%
Experience Detection Accuracy	Around 80%
Average Matching Time	~2-3 Seconds
OCR Processing Time	~4 Seconds
User relevance rating	Around 85-90%
Resume generation time	~1-2 Seconds

4. Discussion

The results indicate that the proposed system is effective in providing useful job recommendations and resume generation. The rule-based approach used in the system is simple and predictable, which makes it easier to maintain and update when new skills or technologies are introduced. Similar approaches have also been explored in previous studies related to job recommendation and skill extraction [2].

Although the system performs well in most cases, there are some limitations. For example, if a skill is not present in the predefined list, it may not be detected automatically. However, this issue can be addressed by allowing users to

manually add or modify their skills.

The experience detection process also depends on how the information is presented in the resume. Variations in format and structure can affect the accuracy of detection. In addition, scanned documents may lead to slightly lower performance due to limitations of text extraction techniques such as OCR [8].

As shown in Table 4, the proposed system provides additional features by combining resume creation and job matching into a single platform. This integration improves user convenience and makes the system more practical for real-world use.

Table 4: Comparison with Existing Platforms

Feature	Ai-Resume Builder	LinkedIn	Indeed	Naukri
Resume Builder	Yes	Basic	No	Basic
Skill Extraction	Automatic + Manual	Manual	Manual	Manual
Experience Detection	Automatic	Manual	No	Manual
Job Matching	Skill+ Experience	Keyword	Keyword	Keyword
Client-Side Parsing	Yes	No	No	No
Open Source	Yes	No	No	No
Drag-and-Drop Sections	Yes	No	No	No

The system is designed to ensure efficient performance and secure data handling. Appropriate measures are used to maintain data isolation and protect user information. These mechanisms help in improving the overall reliability of the system and reducing potential security risks.

5. Conclusion

This paper describes a resume builder and job matching system based on artificial intelligence. The main aim of the system is to make the job search process simpler and more manageable. By combining resume creation and job recommendation in a single platform, users are able to save time and avoid using multiple tools, which is especially helpful for fresh graduates.

The system uses a rule-based approach along with structured data processing to identify user skills and match them with suitable job opportunities. From the results, it can be seen that the system is able to generate useful

suggestions while keeping the performance and response time within a reasonable range.

Another useful aspect of the system is its flexibility. Users can choose from different templates and make changes in real time, which improves the overall experience. Since all major functions are available in one place, the process becomes more convenient for the user.

At the same time, there are some limitations. The system depends on predefined skill lists, so it may miss certain skills if they are not included. Also, resumes with unusual formats can sometimes affect the processing.

In future, the system can be improved by making the skill detection more flexible, adding better recommendation techniques, and supporting multiple languages so that it can be used by a wider group of users.

6. Acknowledgments

The authors express their sincere gratitude to Prof. J. R.

Sancheti, Department of Information Technology, Sant Gadge Baba Amravati University, Amravati, India, for his valuable guidance and continuous support throughout this work. The authors also acknowledge the open-source communities and tools, including React, Supabase, and Tesseract.js, which played an important role in the development of this platform. In addition, we thank the faculty of the Department of Information Technology, Sant Gadge Baba Amravati University, Amravati, India, for their support and encouragement during the course of this project. No external funding was received for this research, and the authors declare no conflict of interest.

References

1. Smelyakov K, Hurova Y, Osiievskyi S. Analysis of the effectiveness of using machine learning algorithms to make hiring decisions. 2023.
2. Kenthapadi K, Le B, Venkataraman G. Personalized job recommendation system at LinkedIn: practical challenges and lessons learned. In: Proceedings of the 11th ACM Conference on Recommender Systems. New York: Association for Computing Machinery; 2017:346-347. DOI: 10.1145/3109859.3109921.
3. Daryani C, Chhabra GS, Patel H, Chhabra IK, Patel R. An automated resume screening system using natural language processing and similarity. In: Zibeline International Publishing; 2020:99-103. DOI: 10.26480/etit.02.2020.99.103.
4. Mittal V, Mehta P, Relan D, Gabrani G. Methodology for resume parsing and job domain prediction. J Stat Manag Syst. 2020;23(7):1265-1274. DOI: 10.1080/09720510.2020.1799583.
5. Bhor S, Shinde H, Gupta V, Nair V, Kulkarni PM. Resume parser using natural language processing techniques.
6. Arora A. Rezume: a web-based resume builder system for university students. 2021.
7. Gugnani A, Misra H. Implicit skills extraction using document embedding and its use in job recommendation. <https://www.computerhope.com/>.
8. Devi S, Anand GP. Technical skills information extraction from resumes using advanced natural language processing model with transfer learning. Published online 2025. DOI: 10.21203/rs.3.rs-7744440/v1.