www.ijarp.com ISSN 2456-9992 Page: 01-13

International Journal Advanced Research Publications

Research Article

Volume: 02
Issue: 04

AI-POWERED RESUME ANALYZER AND APPLICANT TRACKING SYSTEM(ATS)

[1*] Dr. S. Kuzhalvaimozhi, [2] Rohith. S., [2] Kaleem. M., [2] Nishanth. J. Gowda, [2] Nishanth. K. M.

[1]Professor, Department of Information Science and Engineering, The National Institute of Engineering, Mysuru, Karnataka, India.

[2]Undergraduate Student, Department of Information Science & Engineering, The National Institute of Engineering, Mysuru, Karnataka, India.

Article Received: 17 March 2026, Article Revised: 07 April 2026, Published on: 27 April 2026

***Corresponding Author: Dr. S. Kuzhalvaimozhi**

Professor, Department of Information Science and Engineering, The National Institute of Engineering, Mysuru, Karnataka, India.

DOI: https://doi-doi.org/101555/ijarp.4390

ABSTRACT

The rapid expansion of digital recruitment platforms has significantly increased the number of resumes received for each job opening, making manual screening inefficient and inconsistent. traditional applicant tracking systems (ats) rely heavily on keyword-based filtering, which often fails to capture the contextual relevance of skills, experience, and overall resume quality. this paper presents an ai-powered resume analyzer and applicant tracking system designed to automate resume evaluation using artificial intelligence and natural language processing techniques within a fully client-side environment.

KEYWORDS: Applicant Tracking System (ATS), Resume Analysis, Artificial Intelligence, Natural Language Processing, Keyword Filtering, Client-Side Architecture.

INTRODUCTION

The process of finding individuals for positions plays an important role in determining how well organizations function and the overall level of ability in the workforce. As online platforms for positions have developed, organizations receive large numbers of applications for each position that they present, and this makes the examination of documents presenting individual backgrounds a process that requires considerable time, shows inconsistency across cases, and reflects tendencies relating to individual judgment. To address this issue, many organizations use systems that follow applications through the process and provide means for

examining documents in an approach that does not require direct individual assessment. However, systems that follow this approach in the main use methods that identify particular words, and these methods do not allow the examination of meaning in context, the assessment of relevance within particular conditions, or the identification of abilities that apply across different types of work.

Individuals who show appropriate qualifications may be removed from consideration because documents use different words or present information using different structures. This limitation indicates that there is a need for approaches that allow the examination of documents in ways that go beyond the identification of particular words that appear in the text. This work presents a system that uses computational approaches for examining documents relating to applications and following applications through the process, and this system brings together the examination of meaning in context, responses that follow the structure that systems use for following applications, and assessment that considers issues relating to fairness in a form that functions entirely within the interface that individuals use and does not require support from separate computational resources.

RELATED WORK

Initial systems for automated recruitment used filtering based on rules and analysis of keyword frequency to match resumes against job descriptions. These approaches provide computational efficiency but show limitations in understanding meaning and often produce results with low accuracy in screening. Recent research examines models based on transformer structures that encode resumes and job descriptions into vector spaces with contextual information.

These models show performance that is superior in tasks matching individuals to jobs by capturing relationships between skills, experience, and role requirements that are more substantial. Fine-tuning that is specific to the domain provides further improvement in accuracy for recruitment scenarios. Parsing of resumes and preprocessing of documents also receive significant attention. Variations in resume formats, layouts, and documents that are scanned introduce challenges for extraction of text. Studies indicate that conversion from PDF to image and preprocessing based on OCR are important to improve analysis using AI in subsequent stages.

Large Language Models have been explored for generating evaluations of resumes that are explainable. Systems that provide feedback with structure alongside scores that are numerical improve transparency and trust from users.

Fairness in hiring that is automated has also become a concern that is critical, with research indicating risks of bias related to demographics and suggesting evaluation based only on attributes relevant to jobs. Despite these developments, systems that exist in many cases depend on architectures that are backend-heavy and databases that are centralized. The system that is proposed addresses this limitation by combining analysis that is semantic, feedback that is explainable, and design that is aware of fairness within a framework that is fully serverless and operates on the client side.

Skill gap analysis has emerged as a crucial area in AI-driven recruitment research. Patel et al. (20) proposed an AI-based resume screening and skill gap identification system that compared candidate resumes with job descriptions to identify missing competencies. Their system provided targeted recommendations for improvement, demonstrating the potential of NLP-based gap analysis. However, the study did not integrate job discovery or learning roadmaps into a unified workflow.

More recent studies have explored the use of AI for career guidance and upskilling. Dayal et al. (2025) reported large-scale deployment of AI- driven career preparation systems across educational institutions, showing measurable improvements in student preparedness and confidence. These findings validate the real-world impact of AI-assisted career development tools but do not explicitly focus on resume-driven job scraping and roadmap generation.

Although significant progress has been made in resume analysis, job recommendation, and skill gap identification, most existing systems operate in isolation and lack a holistic approach. Current platforms often stop at either recommending jobs or identifying gaps, leaving users without a clear path to skill acquisition. In contrast, the proposed system integrates resume analysis, job scraping, job recommendation, skill gap analysis, and AI-generated learning roadmaps into a single platform. By doing so, it extends existing research by providing actionable, personalized guidance that helps users progress from resume creation to job readiness in a structured and data-driven manner.

PROBLEM STATEMENT

Although approaches using automation show improvement in processes for finding candidates, systems that currently operate reveal multiple limitations. Filters that rely on matching specific terms lack understanding of context and often produce errors in identifying individuals with appropriate qualifications. Designs that focus operations on central servers increase costs for implementation and require more resources for ongoing maintenance, and this limits the extent to which smaller organizations can use these systems.

Most approaches also provide limited information to individuals applying for positions, and this reduces transparency and decreases opportunities for these individuals to improve their materials. Issues relating to fairness and the presence of bias also continue to occur, as certain systems inadvertently consider information about demographic features or formatting features rather than focusing only on skills and experience that candidates possess.

A need exists for a system that analyzes resumes using intelligent methods, that operates at scale, and that produces fair outcomes while providing explanations for its assessments and reducing requirements for infrastructure. The system that this work presents aims to address these issues through analysis using semantic approaches based on artificial intelligence and a design for the frontend that does not require servers.

Based on the extracted skills and role information, the system performs **job scraping** from online job portals. Real-time job postings relevant to the user's profile are collected by analyzing job titles, descriptions, and required competencies. These scraped job descriptions are filtered to ensure relevance to the user's existing skill set and presented to the user as recommended job opportunities. This approach ensures that job recommendations are grounded in actual market data rather than predefined or static datasets.

THE PROPOSED SYSTEM

The system that this work presents operates as a web application with a design that includes multiple components, and the application functions on the client side. The application integrates functions that allow individuals to provide resumes, and these documents undergo processing before analysis occurs. Analysis uses models that examine content, and the system provides responses that include measures and recommendations.

Individuals who use the system complete steps for identification, and this allows documents in PDF format to enter the system. Processing that occurs before analysis ensures that input maintains consistency for examination by models. Models that use approaches from language processing evaluate documents using criteria that systems for tracking applications employ, and evaluation produces responses that show structure. These responses include measures that indicate quality and suggestions that support improvement.

The system examines factors that relate to abilities, background, and content features, and examination excludes attributes relating to demographics. This approach ensures that analysis maintains fairness. The design eliminates components that operate on servers and systems that store data, and this supports privacy for individuals. The approach also reduces costs that deployment requires and simplifies processes that implementation involves.

METHODLOGY

The approach that the system uses follows steps in order. Individuals using the system provide information to confirm identity in a manner that ensures data protection, and this occurs prior to the stage when these individuals transfer documents that show work history and skills. Documents in the portable document format that individuals provide undergo checking to assess whether these meet requirements, and the system then changes the format of these documents to forms that allow processing through models that use artificial intelligence.

These forms include representations that show images or representations that show text content.

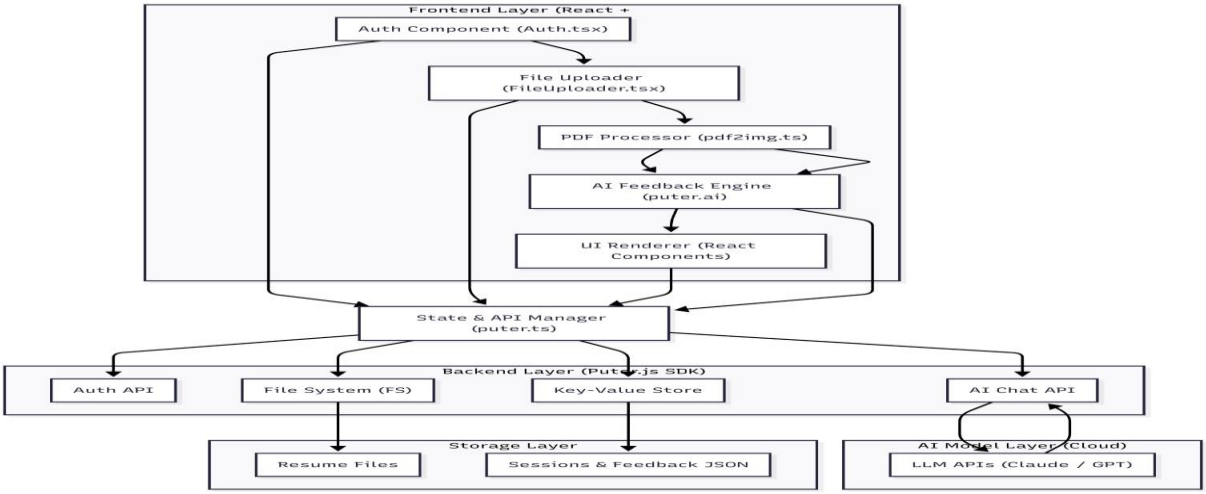

Figure 1: System architecture.

System Architecture

The system for analyzing resumes using AI follows a design that operates in the browser of the user. This design does not require servers that operate in the background or data storage in a central location. The system uses the **Puter.js SDK** to manage functions that include verifying users, handling files, interacting with AI, and storing data. This approach provides benefits that include easier scaling, simpler setup, and better protection for data belonging to users.

Resumes that users upload pass to a processor that converts PDF files. This processor changes the PDF into an image with high resolution or into text form. This conversion provides consistent handling of resumes that have different layouts and structures. The output from this process moves to the engine that generates AI feedback. This engine performs the evaluation of resumes.

The layer managing state and API functions serves as the central point for coordinating the application. This layer uses **puter.ts and Zustand** for managing state. It handles application state, sessions for users, files that users upload, and data from AI feedback.

The layer providing backend services operates entirely through the Puter.js SDK. This SDK functions as a virtual backend that is lightweight. The layer provides essential services without requiring servers that are custom-built for the backend. The layer managing storage handles data that persists during use of the system. Files containing resumes undergo temporary storage using the file system provided, while details about sessions and feedback from analysis appear as structured objects in the paired data store. The layer incorporating analysis models consists of large models for language that operate in cloud hosting, such as Claude or GPT, which the system reaches through the interface for chat-based analysis.This design removes requirements for database systems.

Authentication Module

The module providing authentication establishes secure access for users and manages sessions in the system. It allows individuals to enter and leave the system using services from the Puter.js SDK. This approach ensures that only individuals with authentication can provide resumes and obtain feedback that the system produces. When authentication occurs successfully, the system stores details of user sessions in the application state and in the store for key and value data.

This provides persistence of sessions across interactions that occur while maintaining privacy for users. The module handles authentication without using servers, and this eliminates the need for databases storing user data in the backend and reduces overhead relating to security.

Resume Upload Module

The module for uploading resumes allows individuals to provide resumes in PDF form using an interface that supports dragging and dropping files or selecting files. This module conducts validation on the client side to ensure files that individuals upload meet constraints for size and format before the system conducts further processing.

When validation occurs, the file containing the resume receives temporary storage using the API for the Puter File System and the system passes it to the module conducting preprocessing. This module functions as the point where analysis of resumes enters the system and ensures the system processes only documents that are valid.

PDF Processing Module

The module processing PDFs converts uploaded resume PDFs to a form suitable for analysis using AI. The module uses an engine for rendering PDFs in the browser and converts the resume to an image with high resolution or to text that allows extraction. This step in preprocessing ensures the system handles resumes consistently when they show different layouts and fonts and formatting styles.

AI Resume Analysis Module

The module for AI analysis of resumes represents the main component in the system. It uses Large Language Models accessed through the API for Puter AI to conduct semantic analysis of content in resumes. The module assesses resumes across multiple dimensions that include compatibility with ATS, relevance of skills, quality of content, structure, and tone.

The analysis occurs contextually rather than through matching of keywords in a simple manner, and this allows the system to identify transferable skills and experience with greater accuracy.

Feedback Generation and Visualization Module

Dashboard and Analytics

The module for generation and visualization of feedback presents results that AI produces to the user in a format that is interactive and allows easy understanding. It examines the

structured feedback in JSON form and shows scores for ATS, analysis by category, and suggestions for improvement using components that are visual.

This module provides greater transparency by explaining clearly how the system assessed the resume and what improvements individuals can make. The presentation that is intuitive helps individuals identify strengths and weaknesses in resumes quickly.

Storage Module

The module that provides storage functions manages data that the system produces and that requires persistence during use of the system. Files that contain resumes that users provide are stored in a manner that is temporary. This storage uses the File System that Puter provides. Information relating to sessions and feedback that AI provides are stored as objects that use JSON format. This storage occurs in the Store that provides key-value pairs for data.

TESTING

Resume Processing and Module Testing

Table 1. Resume Processing and Module Testing.

Test Case ID	Module	Test Description	Expected Result	Actual Result	Status
TC01	Authentication	User logs in using Puter.js OAuth.	User redirected to dashboard.	Successful redirection observed.	Pass
TC02	File Upload	Uploads valid .pdf file.	File preview displayed.	Preview shown correctly.	Pass
TC03	File Upload	Attempts to upload invalid .jpg file.	Error message displayed.	Message “Only PDF allowed” shown.	Pass
TC04	PDF Conversion	Converts uploaded PDF into .png.	Image preview generated.	Preview rendered successfully.	Pass
TC05	AI Analysis	Sends resume and job description to AI module.	Structured JSON feedback received.	JSON parsed and displayed correctly.	Pass
TC06	Feedback Display	Displays AI feedback with scores and recommendations.	Scores shown in categorized cards.	Rendered accurately with all data.	Pass
TC07	Error Handling	Simulates network failure or AI timeout.	Retry or error prompt shown.	Error handled gracefully.	Pass

Performance Metrics

Table 2. Performance Metrics.

Parameter	Metric Description	Observed Result
Average AI Response Time	Time taken by the system to send a resume to the AI model through Puter.js and receive structured feedback.	Approximately 3 seconds on an active internet connection.
Resume Upload Time	Time required to upload and store a 1 MB PDF resume using the Puter File System API.	Around 0.5 – 0.8 seconds, depending on network stability.
PDF Conversion Time	Time taken to convert a PDF into an image format using the pdfjs-dist library for further AI analysis.	Averaged 1.3 – 1.6 seconds for single-page documents.
Frontend Load Time	Time taken for the web application (React + Vite) to fully load and render the initial interface in the browser.	Approximately 1.1 seconds on Chrome and Edge browsers.
Error Recovery Rate	Success rate in detecting and handling invalid uploads, connection issues, or AI timeouts without system crashes.	100% of tested cases handled gracefully with error prompts.

RESULTS AND DISCUSSION

The system using approaches powered by methods that provide analysis for documents and tracking for applications was examined to establish that functions, measures of performance, and features relating to consistent operation in providing analysis that occurs without requiring input from individuals were present. Testing that was conducted focused on establishing that main components such as methods for verifying identity, procedures for receiving documents, processing that prepares materials from portable document format, analysis using approaches that provide intelligence, and presentation of results to individuals were functioning.

Testing that examined functions established that the component verifying identity handled periods when individuals use the system in ways that were appropriate using services from Puter point js. The component that receives documents from individuals established format and size in ways that were appropriate, which showed that only documents in portable document format were included in processing.

The component that processes portable document format materials changed documents to forms that were suitable for analysis using approaches providing intelligence in ways that were consistent, and this allowed handling that was the same across different arrangements that documents contain.

The component providing analysis using approaches with intelligence produced results that were organized following the style that systems tracking applications use, and these results included measures showing overall quality, assessment examining how abilities relate to requirements, and recommendations that indicate ways to make documents better. The component that presents results to individuals displayed findings in ways that were clear using parts that were organized by type, and this allowed individuals to understand outcomes from the process examining documents without difficulty.

Assessment examining performance showed that the complete set of procedures analyzing documents was finished within periods of a few seconds when measures across cases were examined. The design using an approach without servers for the part that individuals interact with contributed to delays that were low by removing requirements for communication with systems that process requests, and the system maintained performance that was stable across multiple occasions when testing was conducted without cases where functions failed.

The results that the approach providing intelligence generated were the same across repeated occasions when analysis examined the same document. Differences that were small in results describing features were present, but measures showing overall quality following the approach that systems tracking applications use and abilities identified as strong or weak remained the same. The approach using analysis that examines meaning proved to be effective in providing assessment for documents in ways that go beyond simple matching that compares terms.

Testing that involved individuals using the system indicated that the system was not difficult to use and provided results that contained meaning that was useful. The findings show that the system that this work presents meets objectives that the design established in effective ways by providing assessment for documents that is fast, that shows accuracy, and that can be understood within a setting that is not resource intensive and that operates on the side where individuals interact with the system.

CONCLUSION

The system for analyzing resumes using methods from work in artificial intelligence presented in this study shows how approaches combining language processing and current web design allow improvements in evaluating applications. The main focus of this work was to address limitations that appear in systems using only matching of words by providing

analysis that examines meaning, generates clear responses for individuals, and operates without requiring large computing structures.

The approach uses models that process language to examine resumes in ways that consider context and relationships between elements. This allows assessment not only of particular abilities that appear in documents but also of how these abilities relate to requirements, how information is organized, and how content is presented. The system produces measures similar to those used in tracking applications and provides organized responses that show individuals their strong features and aspects requiring development. This increases clarity in processes for selecting candidates. The approach differs from other systems by avoiding dependence on large server structures and centralized storage of data, which results in lower costs for operation, better capacity for expansion, and improved protection of information.

The system provides an approach that is practical, efficient, and fair compared to other methods for screening resumes. This work shows how design using web-based intelligence combined with large models for processing language can change workflows in recruitment, improve experience for candidates, and support decisions in hiring that are informed by clear analysis. The study meets objectives that were stated and establishes a foundation for future development toward solutions for recruitment at larger scales.

FUTURE ENHANCEMENTS

Multi-page resume parsing support

The system shows improvement through support for examining resumes across multiple pages. The current approach focuses on the first page to maintain performance in the analysis. Extending the analysis to include all pages allows the system to provide more comprehensive evaluation of experience that candidates present and projects that individuals include in documents. This produces more accurate measures for scoring used by systems that track applicants.

Automatic job description fetching and matching

Another development involves automatic retrieval of descriptions from job postings and matching these to resumes. The integration of external portals and interfaces allows the system to retrieve descriptions in a dynamic way and conduct matching between resumes and jobs without requiring manual input from users. This transforms the system from a tool that analyzes resumes into a platform that provides complete assistance for recruitment.

Recruiter-oriented analytics dashboard

A dashboard designed for recruiters represents another possible extension. Such a dashboard allows recruiters to compare multiple candidates and view summaries that rank individuals. It also allows analysis of strengths and limitations that candidates show and supports decisions about hiring that use data in an efficient way.

Multi-language resume analysis

Additional developments include integration with storage systems based in the cloud for archiving resumes over extended periods. The system can also support analysis of resumes in multiple languages to allow recruitment across different regions. Performance improvements through methods that store data temporarily and adjustments to models that reduce computational requirements can further decrease the time required for responses.

REFERENCES

1. Pham, T., et al., "Resume2Vec: Transformer-based Matching for Recruitment," Electronics (MDPI), 2025. https://www.mdpi.com/2079-9292/14/4/794
2. Zhang, Y., et al., "BERT-based Person–Job Fit with Attention Mechanism," Journal of Big Data (Springer), 2023. https://journalofbigdata.springeropen.com/articles/10.1186/s40537-023-00767-4
3. Kaur, A., & Kumar, R., "Comparative Study of Pre-trained Language Models for Job Recommendation," IEEE Access, 2023. https://ieeexplore.ieee.org/document/10051319
4. Zhao, L., & Xu, J., "LLM Agents for Automated Resume Screening," arXiv preprint arXiv:2409.06789, 2024. https://arxiv.org/abs/2409.06789
5. Kaushik, P., & Ghosh, S., "Enhanced Resume Screening using Sentence-BERT," IJACSA, 2024. https://thesai.org/Publications/ViewPaper?Code=IJACSA&Issue=8&SerialNo=25&Volume=15
6. Wang, H., & Li, D., "Automated Resume Parsing using NLP and Machine Learning," Expert Systems with Applications (Elsevier), 2022. https://doi.org/10.1016/j.eswa.2022.118234
7. Goyal, A., et al., "NLP-based Resume Classification for ATS Systems," Int. J. Engineering Research, 2023.
8. Nikiforos, M., & Foussias, G., "Job Recommender Systems: A Comprehensive Review," ACM Computing Surveys, 2021. https://dl.acm.org/doi/10.1145/3466824

9. Mehrabi, N., et al., “A Survey on Bias and Fairness in Machine Learning,” ACM Computing Surveys, 2021. https://dl.acm.org/doi/10.1145/3457607
10. Barocas, S., & Raghavan, M., “The Ethics of Algorithmic Hiring Systems,” Communications of the ACM, 2021.
11. Sun, Q., & Williams, D., “Ethics and Discrimination Risks in AI-enabled Recruitment,” Nature Human Behaviour, 2023. https://www.nature.com/articles/s41562-023-01601-3 .
12. European Union, “Artificial Intelligence Act: Regulation of AI Systems,” COM/2021/206 final, Brussels, 2024. https://artificialintelligenceact.eu
13. U.S. Equal Employment Opportunity Commission, “Technical Assistance on AI and Title VII,” 2023. https://www.eeoc.gov/ai
14. NYC Commission on Human Rights, “Local Law 144: Automated Employment Decision Tools Guidance,” 2023.
https://www.nyc.gov/site/cchr/law/automated-employment-decision-tools.page